



DEPARTAMENTO DE LENGUAJES Y SISTEMAS
INFORMÁTICOS E INGENIERÍA DEL SOFTWARE

FACULTAD DE INFORMÁTICA

UNIVERSIDAD POLITÉCNICA DE MADRID

TESIS DOCTORAL

MARCO DE INTEGRACIÓN DE LA
USABILIDAD EN EL PROCESO DE
DESARROLLO SOFTWARE

AUTOR: Xavier Ferré Grau

DIRECTORAS: Natalia Juristo

Ana María Moreno

2005

*A la meva mare i als meus germans,
i a qui al meu costat està, omplint la meva vida*

Agradecimientos

La realización del presente trabajo de tesis ha supuesto un esfuerzo a lo largo de los años, en el cual he podido comprobar el cariño y la amistad de muchas personas. En estas líneas únicamente podré incluir una parte de todos los agradecimientos que me gustaría dar.

En primer lugar, querría agradecer a mis directoras de tesis, Natalia y Ana. A Natalia por las oportunidades que me ha dado, las cuales me han permitido sacar a la luz lo mejor de mí mismo en el trabajo, por su paciencia y por la fenomenal capacidad de análisis y visión que me ha demostrado en todo el tiempo que llevo trabajando con ella, y por todo lo que he aprendido de ella en lo referente a las relaciones interpersonales y la inteligencia emocional. A Ana por su capacidad de trabajo y por su disposición a ayudar en cualquier momento, por su especial capacidad de analizar las cosas desde el punto de vista más ventajoso, y por su apoyo incondicional, haciendo que me sintiera uno más en el grupo desde el primer momento.

También me gustaría agradecer en general a todo el grupo de Ingeniería del Software, por todo lo que me han aportado y por su brillantez, que hace que me sienta muy orgulloso de estar a su lado. En especial querría agradecer a Angélica, Ricardo y Nelson por los buenos momentos que me han hecho pasar a lo largo de todos estos años, y por creer en mi trabajo que ahora da sus frutos en forma de tesis. A Nelson también por su espíritu inquieto, por haberme transmitido el entusiasmo por el estudio del desarrollo de software, y por haberme ayudado a ver con otros ojos el mundo que nos rodea. A Nora por su capacidad de emocionarse y emocionarme, y a Tere por su apoyo en todo momento y por hacerme partícipe de su visión del mundo.

En este último año también he tenido la suerte de formar parte de un espléndido equipo en Decanato. Quiero agradecer a todo el equipo lo que me han aportado como persona, que ha sido mucho, y su compañía en todos los momentos buenos y malos que hemos pasado juntos, que ha servido para acrecentar aún más el aprecio que ya les tenía.

Quiero agradecer también a mis amigos de la Coral Polifónica Sagrada Familia, que han seguido la evolución de mi trabajo de tesis con incansables preguntas por su finalización, siempre con alegría y haciéndome sentir su cariño. Su preocupación por mi tesis me ha ayudado mucho a conseguir un grado más de perseverancia, y ha sido un gran apoyo. Entre ellos, María, Gemma, Pepema e Ivonne, Carmen, Nieves, Asun, Pablo, Cristina y Marisa ocupan un lugar único por los momentos tan especiales que hemos pasado juntos y por la gran amistad y cariño que nos une.

Entre mis amigos más veteranos, también Fede, Ramos, Marta y Pilar han estado a mi lado todos estos años, y me han demostrado el gran corazón que tienen y la capacidad de hacerme sentir fenomenal con ellos a pesar de que pasáramos largas temporadas sin vernos. Querría también agradecer a Alejandro por todo lo que me ha aportado con su especial visión del mundo y su siempre estimulante compañía. A Fermín por su gran generosidad y por haberme ofrecido la capacidad de poder encontrar mi lugar en el mundo. A mis amigos del CERN por acogerme entre ellos y hacer que ese año quedara grabado en mi memoria como ninguno. Todos mis amigos han contribuido de una u otra manera apoyándome durante la realización de esta tesis. A pesar de que no pudiera dedicarles todo el tiempo que me hubiera gustado, siempre han estado a mi lado cuando les necesitaba.

Finalmente, a mi familia, tengo tantas cosas que agradecerles. A mi madre, a mis hermanos y a mi cuñada Marta quiero, sobre todo, agradecerles por el apoyo y el amor que me han demostrado toda la vida, que me ha permitido desarrollarme como persona, y me ha hecho sentir en todo momento como una persona especial. A mis sobrinos Albert, Marc, Neu y Jaume Genís, por haberme hecho sentir que soy una parte importante de sus vidas.

Resumen

En el desarrollo de software está incrementándose el reconocimiento de la usabilidad como atributo de calidad clave para el éxito de un producto. Sin embargo, las técnicas pertenecientes al campo de la IPO (Interacción Persona-Ordenador), las cuales nos permiten alcanzar un nivel adecuado de usabilidad en el producto software, se aplican en el marco de métodos IPO, y no en el proceso de desarrollo habitual. Por otra parte, la construcción de software según los métodos y prácticas de la IS (Ingeniería del Software) no trata de forma adecuada la usabilidad a lo largo del proceso de desarrollo. El problema que se aborda en este trabajo es, precisamente, cómo ayudar a los ingenieros software a seleccionar las técnicas y actividades IPO más apropiadas para integrar en su proceso de desarrollo, de modo que se trate adecuadamente la usabilidad del producto software. El estudio realizado sobre el estado de la cuestión revela que dicho problema está abierto.

La solución original que se propone consiste en un marco de integración de la usabilidad en el proceso de desarrollo. El marco permite al ingeniero software acceder a una selección de las 35 técnicas IPO mejor dotadas para su integración en el proceso de una organización con escasa o nula experiencia previa en el tratamiento de la usabilidad. El único requisito para el proceso de desarrollo en el que se quiera integrar la usabilidad es que tenga un carácter iterativo. Para cada técnica del marco se detalla su caracterización según un conjunto de criterios relevantes para el objetivo de integración, desde el punto de vista de la IS.

Se ha comprobado la viabilidad de la solución propuesta mediante su aplicación por un socio industrial del proyecto europeo STATUS en la construcción de un producto comercial. Por otra parte, la solución propuesta se ha aplicado en dos casos de estudio realizados por alumnos de la Facultad de Informática de la Universidad Politécnica de Madrid.

En definitiva, este trabajo aporta, por primera vez, una herramienta flexible que facilita la selección de técnicas y actividades IPO a integrar en el proceso de desarrollo, dando respuesta a las posibles necesidades de un amplio abanico de organizaciones y procesos software.

Abstract

Software development organizations are paying more and more attention to usability as a critical quality attribute for the success of a product. Nevertheless, techniques belonging to the HCI (Human-Computer Interaction) field, which allow to reach an adequate usability level in a software product, are applied as part of HCI methods, but they are decoupled from the overall software development process. On the other side, software construction according to SE (Software Engineering) methods and practices does not adequately address usability all over the development process. The problem addressed is, therefore, how to help software engineers to select the best HCI techniques and activities to integrate into their particular software development process, so that the usability of the software product is properly managed. The study performed in the state of the art reveals that such a problem is not solved yet.

The original solution proposed is a framework for integrating usability into the software process. The framework offers software engineers a set of 35 selected HCI techniques, which are the best fit for their integration into the software process of an organization with little or no previous experience in usability. The only condition to be met by the software process into which usability is to be integrated is to be based on an iterative approach. The framework details for each HCI technique its characterization according to a set of criteria which are relevant for the integration aim, from a SE viewpoint.

The feasibility of the proposed solution has been verified through its application by an industrial partner of the European-funded project STATUS, in the development of a commercial product. Additionally, the proposed solution has been applied in two case studies by students of the Computer Science School at Universidad Politecnica de Madrid.

In short, this work offers, for the first time, a flexible tool that eases the selection of HCI techniques and activities to integrate into the development process, providing an answer to the needs of a wide range of organizations and software processes.

Índice de Contenidos

CAPÍTULO 1 INTRODUCCIÓN	1
1.1 ÁREA DE INVESTIGACIÓN: EL PROCESO DE DESARROLLO SOFTWARE Y LA USABILIDAD.....	1
1.2 PROBLEMA DE INVESTIGACIÓN ABORDADO	6
1.2.1 Identificación del Problema	6
1.2.2 Relevancia del Problema y Dificultades que Plantea	7
1.3 APROXIMACIÓN A LA SOLUCIÓN.....	9
1.4 PUBLICACIÓN DE RESULTADOS.....	11
CAPÍTULO 2 ESTADO DE LA CUESTIÓN	15
2.1 INTRODUCCIÓN	15
2.2 ESTÁNDARES DE PROCESO DE DESARROLLO SOFTWARE	16
2.2.1 Estándar ISO/IEC 12207:1995.....	17
2.2.1.1 Descripción	17
2.2.1.2 Análisis.....	18
2.2.2 Estándar IEEE 1074-1997.....	19
2.2.2.1 Descripción	19
2.2.2.2 Análisis.....	20
2.2.3 Enmienda 1 al Estándar ISO/IEC 12207.....	21
2.2.3.1 Descripción	21
2.2.3.2 Análisis.....	22
2.3 MÉTODOS IPO	23
2.3.1 Ciclo de Vida en Estrella	23
2.3.1.1 Descripción	23
2.3.1.2 Análisis.....	24
2.3.2 Estándar ISO 13407	26
2.3.2.1 Descripción	26
2.3.2.2 Análisis.....	30
2.3.3 Ciclo de Vida de la Ingeniería de Usabilidad.....	30
2.3.3.1 Descripción	30
2.3.3.2 Análisis.....	33
2.3.4 Diseño Centrado en el Uso.....	34
2.3.4.1 Descripción	34
2.3.4.2 Análisis.....	37
2.3.5 Modelo ISO/TR 18529.....	39
2.3.5.1 Descripción	39
2.3.5.2 Análisis.....	40
2.4 EXTENSIÓN DE MÉTODOS IS CON ASPECTOS DE USABILIDAD	41
2.4.1 Propuesta de Coutaz.....	41
2.4.1.1 Descripción	41
2.4.1.2 Análisis.....	43
2.4.2 MUSE (Method for USability Engineering).....	44
2.4.2.1 Descripción	44
2.4.2.2 Análisis.....	46
2.4.3 Propuesta de Hakiel	49
2.4.3.1 Descripción	49
2.4.3.2 Análisis.....	51
2.4.4 LUCID	52
2.4.4.1 Descripción	52

2.4.4.2 Análisis.....	54
2.4.5 Propuesta de Costabile	54
2.4.5.1 Descripción	54
2.4.5.2 Análisis.....	56
2.4.6 Propuesta ad-hoc de Anderson et al.	57
2.4.6.1 Descripción	57
2.4.6.2 Análisis.....	60
2.4.7 Wisdom.....	61
2.4.7.1 Descripción	61
2.4.7.2 Análisis.....	62
2.4.8 Plug-in de Experiencia del Usuario para RUP	64
2.4.8.1 Descripción	64
2.4.8.2 Análisis.....	65
2.4.9 RUPi.....	66
2.4.9.1 Descripción	66
2.4.9.2 Análisis.....	67
2.4.10 Caracterización de Técnicas de Holzinger	68
2.4.10.1 Descripción.....	68
2.4.10.2 Análisis.....	69
2.5 RESUMEN DE LA REVISIÓN DEL ESTADO DE LA CUESTIÓN.....	70

CAPÍTULO 3 PLANTEAMIENTO DEL PROBLEMA 75

3.1 INTRODUCCIÓN	75
3.2 DEFINICIÓN DEL PROBLEMA.....	75
3.2.1 Diferencias entre Proceso IPO y Proceso IS	76
3.2.2 Diferencias de Terminología entre IPO e IS	77
3.2.3 Cuándo Tratar la Usabilidad en el Desarrollo	78
3.3 HIPÓTESIS DE TRABAJO	78
3.4 CONDICIONES DE APLICACIÓN DE LA SOLUCIÓN	79
3.5 ESBOZO DE LA SOLUCIÓN	80
3.6 APLICACIÓN Y EVALUACIÓN DE LA SOLUCIÓN PROPUESTA.....	82

CAPÍTULO 4 ACTIVIDADES Y TÉCNICAS EN UN PROCESO CENTRADO EN EL USUARIO..... 85

4.1 INTRODUCCIÓN	85
4.2 ESTUDIO DE ACTIVIDADES DE USABILIDAD EN LA LITERATURA IPO	86
4.2.1 Fuentes Consideradas.....	86
4.2.2 Resultados del Estudio	87
4.2.3 Especificación del Contexto de Uso.....	91
4.2.3.1 Análisis de Usuarios	91
4.2.3.2 Análisis de Tareas	92
4.2.4 Especificaciones de Usabilidad.....	93
4.2.5 Desarrollo del Concepto del Producto	94
4.2.6 Prototipado	95
4.2.7 Diseño de la Interacción.....	96
4.2.8 Evaluación de Usabilidad.....	97
4.3 ESTUDIO DE TÉCNICAS DE USABILIDAD EN LA LITERATURA IPO	97
4.3.1 Fuentes Consideradas.....	98
4.3.2 Proceso de Recopilación y Clasificación	99
4.3.3 Clasificación de las Técnicas	101
4.3.3.1 Técnicas Relacionadas con Actividades de Análisis	101
4.3.3.2 Técnicas Relacionadas con Actividades de Diseño.....	103
4.3.3.3 Técnicas Relacionadas con Actividades de Evaluación	104
4.4 CARACTERIZACIÓN DE TÉCNICAS	108
4.4.1 Criterios de Caracterización.....	108
4.4.2 Caracterización de Técnicas Relacionadas con Análisis.....	111
4.4.3 Caracterización de Técnicas Relacionadas con Diseño.....	114
4.4.4 Caracterización de Técnicas Relacionadas con Evaluación	118

4.5 SELECCIÓN DE TÉCNICAS.....	122
--------------------------------	-----

CAPÍTULO 5 ASIGNACIÓN DE TÉCNICAS IPO..... 129

5.1 INTRODUCCIÓN	129
5.2 RELACIÓN ENTRE ACTIVIDADES IPO Y ACTIVIDADES IS.....	129
5.2.1 Actividades IPO Relacionadas con Análisis	130
5.2.2 Actividades IPO Relacionadas con Diseño.....	131
5.2.3 Actividades IPO Relacionadas con Evaluación	132
5.2.4 Descripción de la Relación entre Actividades IPO y Actividades IS.....	134
5.3 ASIGNACIÓN DE TÉCNICAS IPO A TIPOS DE ACTIVIDADES IS.....	135
5.3.1 Técnicas Asignadas a Actividades de Análisis	135
5.3.1.1 Educción y Análisis de Requisitos	135
5.3.1.2 Especificación de Requisitos.....	140
5.3.1.3 Validación de Requisitos.....	140
5.3.2 Técnicas Asignadas a Actividades de Diseño	141
5.3.2.1 Diseño de la Interacción	141
5.3.2.2 Otras Técnicas Relacionadas con el Diseño	142
5.3.3 Técnicas Asignadas a Actividades de Evaluación	143
5.3.3.1 Evaluación por Expertos	143
5.3.3.2 Test de Usabilidad.....	144
5.3.3.3 Estudios de Seguimiento de Sistemas Instalados	145
5.3.4 Resumen.....	145

CAPÍTULO 6 MOMENTOS DE APLICACIÓN DE TÉCNICAS IPO EN EL DESARROLLO 147

6.1 INTRODUCCIÓN	147
6.2 PRERREQUISITOS SOBRE EL PROCESO DE DESARROLLO.....	148
6.2.1 Características de un Proceso Centrado en el Usuario	148
6.2.2 Condiciones de un Proceso para Integrar la Usabilidad.....	150
6.3 TIPOS DE CICLOS EN UN PROCESO ITERATIVO	151
6.4 ASIGNACIÓN DE TÉCNICAS IPO A ETAPAS DEL DESARROLLO	152
6.4.1 Técnicas para los Ciclos Iniciales	152
6.4.1.1 Técnicas Especialmente Apropriadas	153
6.4.1.2 Técnicas Neutras	155
6.4.1.3 Técnicas No Habituales.....	156
6.4.2 Técnicas para los Ciclos Centrales.....	157
6.4.2.1 Técnicas Especialmente Apropriadas	157
6.4.2.2 Técnicas Neutras	157
6.4.2.3 Técnicas No Habituales.....	159
6.4.3 Técnicas para los Ciclos de Evolución.....	160
6.4.3.1 Técnicas Especialmente Apropriadas	160
6.4.3.2 Técnicas Neutras	160
6.4.3.3 Técnicas No Habituales.....	162
6.5 ADECUACIÓN DE LAS TÉCNICAS IPO A LAS ETAPAS DE UN DESARROLLO ITERATIVO.....	162

CAPÍTULO 7 INTEGRACIÓN DE LAS TÉCNICAS IPO EN EL PROCESO.... 165

7.1 INTRODUCCIÓN	165
7.2 VISTA POR TÉCNICAS.....	166
7.2.1 Criterios de Caracterización.....	166
7.3 VISTA POR ACTIVIDADES.....	170
7.3.1 Actividades Afectadas por la Usabilidad	170
7.3.2 Tabla Resumen por Tipo de Actividad	172
7.4 VISTA POR MOMENTOS DEL DESARROLLO	175
7.4.1 Etapas en un desarrollo iterativo	175
7.4.2 Tabla Resumen por Momento de Aplicación.....	176
7.5 USO DEL MARCO	176
7.6 HERRAMIENTA WEB PARA EL USO DEL MARCO DE INTEGRACIÓN.....	180

CAPÍTULO 8 APLICACIÓN Y EVALUACIÓN DE RESULTADOS	181
8.1 INTRODUCCIÓN	181
8.2 ESTUDIO DE VIABILIDAD	181
8.2.1 Aplicación e–Suite	182
8.2.2 Prácticas de Desarrollo en LogicDIS	182
8.2.3 Estrategia para la Integración de la Usabilidad	184
8.2.4 Resultados de la Aplicación del Marco de Integración	185
8.2.4.1 Impacto en el Proceso de Desarrollo	185
8.2.4.2 Impacto en la Usabilidad Percibida por los Usuarios	187
8.3 CASOS DE ESTUDIO.....	188
8.3.1 Caso de Desarrollador con Experiencia Media	188
8.3.1.1 Aplicación de Gestión de Calendarios	188
8.3.1.2 Integración de la Usabilidad en el Proceso de Desarrollo	189
8.3.1.3 Resultados Preliminares de la Aplicación del Marco de Integración	190
8.3.2 Caso de Desarrolladores Noveles.....	192
8.3.2.1 Proyecto Práctico de Construcción de un Sistema Software	192
8.3.2.2 Integración de la Usabilidad en el Proceso de Desarrollo	192
8.3.2.3 Resultados de la Aplicación del Marco de Integración	193
8.4 DISCUSIÓN	195
 CAPÍTULO 9 CONCLUSIONES Y LÍNEAS FUTURAS.....	199
9.1 INTRODUCCIÓN	199
9.2 CONCLUSIONES.....	199
9.3 LÍNEAS FUTURAS DE INVESTIGACIÓN	202
 BIBLIOGRAFÍA	205
 ANEXO A TÉCNICAS DE USABILIDAD.....	213
 ANEXO B CUESTIONARIO LOGICDIS.....	233
 ANEXO C CUESTIONARIO ALUMNO DE MASTER.....	241
 ANEXO D CUESTIONARIO DE SATISFACCIÓN.....	247
 ANEXO E CUESTIONARIO ALUMNO DE GRADO	253
 ANEXO F ÍNDICE DE ACRÓNIMOS	259

Capítulo 1

Introducción

El trabajo de investigación que se presenta en este documento se enmarca en el área de proceso de desarrollo software de la IS (Ingeniería del Software), y desde dicha perspectiva planteamos el problema de la integración de las técnicas y actividades de usabilidad en el desarrollo de software. En primer lugar, se introduce el área de investigación con los condicionantes que plantea, para a continuación tratar la existencia y relevancia del problema que el presente trabajo aborda. Finalmente, se presenta la aproximación a la solución que se propone y se detallan las publicaciones que se han realizado de este trabajo.

1.1 Área de Investigación: El Proceso de Desarrollo Software y la Usabilidad

El proceso de desarrollo software es el mapa que sigue una organización para producir sistemas software, esto es, el conjunto de actividades que se llevan a cabo para el desarrollo.

Los procesos de desarrollo en los que se enmarcan las técnicas y actividades de usabilidad objeto del presente trabajo (denominados procesos centrados en el usuario) tienen un enfoque distinto al de la IS. Se va a tratar, en primer lugar, la perspectiva de proceso de la IS, mientras que, a continuación, se aborda el enfoque de los procesos centrados en el usuario.

La IS se caracteriza por aplicar un enfoque sistemático, disciplinado y cuantificable al desarrollo de software [IEEEGlossary, 90]. En el contexto de esta preocupación por la adopción de prácticas sistemáticas y disciplinadas, la definición del proceso de desarrollo software tiene un carácter especialmente relevante en la disciplina. Debido al interés que la comunidad de la IS ha puesto en el proceso software, se ha invertido un esfuerzo considerable en el estudio de la definición, evaluación y mejora del proceso software [Kawalek, 96] [Dermiane, 99] [Fuggetta, 00].

Las organizaciones de desarrollo de software ponen especial énfasis en la definición de un proceso que permita controlar la construcción de software de forma similar al control de procesos de producción en otras ingenierías. Razones típicas por las que una organización de desarrollo de software puede considerar valioso un proceso definido son: Incrementar la calidad del producto, facilitar la comprensión y comunicación entre miembros del equipo, dar soporte a la mejora del proceso, dar soporte a la gestión del proceso, proporcionar guiado automático en la aplicación del proceso y proporcionar soporte a la ejecución automática de ciertas tareas [SWEBOK, 04].

El proceso de desarrollo se puede definir a distintos niveles, existiendo una gran diversidad de procesos de desarrollo definidos en la literatura para la construcción de sistemas software

[Acuña, 01]. Dentro de esta variedad podemos identificar elementos comunes: Un proceso definido típicamente incluye al menos la descripción de las actividades que hay que llevar a cabo, junto con las técnicas a aplicar en cada actividad. Por otra parte, cada proceso tiene una terminología propia para las actividades que lo forman, la cual depende del nivel al que esté definido y de los condicionantes del desarrollo en los que se centra. A pesar de la diferencia de terminología, sin embargo, las actividades de un proceso pueden típicamente enmarcarse en tipos genéricos de actividades como son: Actividades de análisis, de diseño, y de evaluación. Un ingeniero software puede acomodar las actividades que formen parte del proceso de desarrollo definido en su organización a tipos genéricos de actividades de este tipo, puesto que forman parte del cuerpo de conocimientos de la IS [SWEBOK, 04].

Cada organización cuenta con su propio proceso software, que puede estar explícitamente definido o no, y que los desarrolladores siguen con mayor o menor grado de fidelidad. El área de proceso software defiende que un proceso definido explícitamente y mejorado de forma continua logrará un mejor producto. Es decir, defiende la hipótesis de que el empleo de un buen proceso de desarrollo conlleva la obtención de un producto software de buena calidad.

Si nos planteamos qué significa obtener un producto de buena calidad, podemos tomar la definición de Humphrey de calidad software: "alcanzar niveles excelentes de adecuación para el uso" [Humphrey, 89] (según se cita en [SWEBOK, 04]). Así, para evaluar la calidad del software es preciso comprender el propósito para el cual el sistema se va a usar. Por tanto, la calidad no es una medida del software aislado, sino que es una medida que tiene en cuenta la relación entre el producto y su dominio de aplicación. Haciendo una analogía con el diseño industrial, un chip destinado a la industria del juguete se considera de buena calidad si cumple ciertos criterios, mientras que para un chip similar destinado a la industria aeronáutica se considerarán unos requisitos mucho más exigentes para ser juzgado como poseedor de un nivel de calidad suficiente. En el caso del software, muchas veces no está definido con precisión el dominio en el que va a ser utilizado, construyéndose de esta forma sistemas software con un nivel de calidad insuficiente para el dominio de aplicación. El usuario es una parte esencial de tal dominio de aplicación, por lo que la usabilidad es un componente básico de la calidad del software. Esta visión de la usabilidad como atributo de calidad no es novedosa, puesto que, a pesar de que no existe un conjunto aceptado de atributos críticos de calidad software, la usabilidad ha estado presente como atributo en descomposiciones de calidad del software desde los años 70 [Boehm, 78] [McCall, 77]. Por tanto, la usabilidad se reconoce como atributo de calidad del software.

Que se reconozca explícitamente la usabilidad como atributo de calidad, no significa que automáticamente las prácticas de la IS traten adecuadamente todo lo relacionado con la usabilidad del producto software que está desarrollando. Por el contrario, observando las técnicas de la IS que forman parte de los distintos procesos de desarrollo existentes, podemos identificar diferencias en el modo en que son tratados distintos atributos de calidad del software. Una parte de los atributos, como por ejemplo la fiabilidad o la corrección, han recibido una gran atención, y su presencia en el desarrollo está implícita por la preocupación por dichos atributos que permea las técnicas de la IS. A pesar de la gran diversidad de procesos existentes, la incorporación de técnicas que tratan estos atributos a lo largo del desarrollo es habitual entre los distintos procesos. Dicha situación es consecuencia de la premisa aceptada en la IS de que los atributos de calidad deben tratarse en distintas actividades del proceso de desarrollo. El cuidado por la usabilidad, por el contrario, no está presente en las técnicas de la IS más que tangencialmente, y, así, su presencia en los procesos de desarrollo es marginal. En la mayor parte de los casos no existe o está relegada a una única actividad de diseño de la IU (Interfaz de Usuario), sin existir una preocupación por la usabilidad a lo largo de todo el proceso de desarrollo, como sí existe para otros atributos de calidad. Quizás esta diferencia de interés entre unos atributos de calidad y otros se deba a que la IS se ha centrado principalmente en establecer mecanismos sistemáticos para la construcción de software para problemas bien definidos. Incluso en este caso, los desafíos que

se plantean son muy importantes, y las técnicas de la IS se han enfocado a resolver los problemas que pueden surgir para este tipo de problemas. La preocupación por la usabilidad impone la incorporación de usuarios en el proceso de desarrollo. Sin embargo, el trato con usuarios, el acomodo a sus necesidades y limitaciones, introduce un grado importante de incertidumbre en el desarrollo, debido a la complejidad de la psique humana y a la variabilidad interpersonal. Frente a una fórmula o especificación escrita, las apreciaciones subjetivas de los usuarios resultan mucho más complejas de tratar. Dada la carencia de las técnicas de la IS a este respecto, es preciso complementarlas en ciertas actividades con técnicas de usabilidad que pueden aportar la adecuada perspectiva de usabilidad a dichas actividades. En ocasiones será necesario, incluso, modificar la forma de aplicación de ciertas técnicas de la IS, mientras que en otras bastará con incluir cierto tipo de actividades de usabilidad en el desarrollo que no suelen tenerse en cuenta habitualmente.

Por tanto, podemos concluir que la IS concede una especial importancia al empleo de un proceso de desarrollo software definido. Dicho proceso típicamente incluye al menos la definición de actividades que hay que llevar a cabo, junto con las técnicas a aplicar en cada actividad. La descripción de las actividades de un proceso sigue la terminología propia de dicho proceso, pero habitualmente pueden referirse a tipos genéricos de actividades que forman parte del corpus de conocimientos de la IS. En cuanto a la usabilidad, está considerada como uno de los atributos de calidad del software, pero las técnicas que se aplican a lo largo de los procesos de la IS no tienen en cuenta la usabilidad como tienen en cuenta otros atributos de calidad. Para que la tengan en cuenta será preciso modificar la aplicación de técnicas en ciertas actividades, complementado su aplicación con técnicas de usabilidad, mientras que en otras será necesario mejorar el proceso con la inclusión de cierto tipo de actividades de usabilidad que aporten el punto de vista preciso para un adecuado tratamiento de la usabilidad.

La disciplina denominada HCI (*Human-Computer Interaction*) en inglés, se denomina IPO (Interacción Persona-Ordenador) en castellano [Lorés, 01], y se ocupa según Preece et al. de "diseñar sistemas informáticos que den soporte a personas de tal forma que éstas puedan llevar a cabo sus actividades productivamente y con seguridad personal" [Preece, 94]. Por tanto, la usabilidad es una de las máximas preocupaciones de la IPO como objetivo del desarrollo de sistemas informáticos, siendo la disciplina que mejor trata la usabilidad en el desarrollo de software. A la IPO contribuyen distintos campos del saber como, por ejemplo, la Informática, la Psicología, la Sociología o el Diseño Industrial y la Ergonomía. Así, la IPO está dotada de un carácter multidisciplinar per se, que proporciona todos los puntos de vista que se requieren para el desarrollo de sistemas informáticos que den un soporte adecuado a personas con todos los condicionantes que dicho objetivo implica. La Figura 1.1 muestra la diversidad de temas que trata la IPO.

La visión que tiene la IPO del desarrollo de software es, en cierto sentido, más amplia que el enfoque habitual de la IS, en el que la atención se centra en gran medida en el sistema software en ejecución. La IPO considera también las implicaciones sociales y organizacionales de la realización de una tarea con un sistema informático, así como las limitaciones del ser humano en el procesamiento de información, y los temas de comunicación implicados. Por el contrario, al tener esta visión interdisciplinar del desarrollo de software, ocupándose de temas en los que se intersectan diversas disciplinas, no trata el desarrollo de software con la misma profundidad en temas específicos. Así, la IS tiene un nivel de detalle mayor que la IPO en los temas que le son propios, como puede ser el proceso de desarrollo software. La Figura 1.2 muestra gráficamente como la IPO tiene más amplitud en cuanto a temas abarcados, pero menor profundidad en el modo en el que trata la construcción del software en sí.



Figura 1.1 - Temas que Trata la IPO

Interacción
Persona-Ordenador
(IPO)



Ingeniería
del Software
(IS)

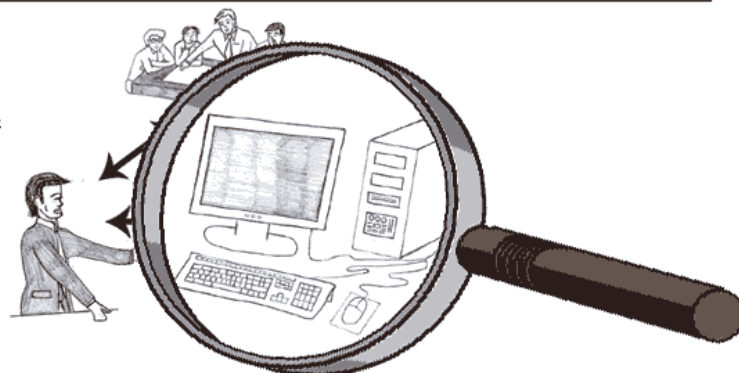


Figura 1.2 - Comparación entre los Objetivos de la IPO y de la IS

La IPO es un campo maduro, cuyos métodos son utilizados en numerosas organizaciones de desarrollo de software para asegurar que el nivel de usabilidad del producto software alcance

un mínimo establecido. Los métodos de la IPO tienen un enfoque de desarrollo que se denomina centrado en el usuario (o centrado en el humano, según la fuente). El estándar ISO 13407 sobre procesos de desarrollo centrados en el humano para sistemas interactivos [ISO13407, 99] indica que la incorporación de un enfoque centrado en el humano se caracteriza por lo siguiente:

- La implicación activa de usuarios y una clara comprensión de los requisitos de usuarios y sus tareas,
- una adecuada asignación de funciones entre los usuarios y la tecnología,
- la iteración de soluciones de diseño, y
- el diseño multidisciplinar.

Debido a la visión más amplia a la que aspira la IPO, los métodos que le son propios y que, por tanto, siguen el enfoque de desarrollo centrado en el usuario, no se ocupan adecuadamente de otros atributos de calidad que no sean la usabilidad. Los objetivos en los que se centra la IS no son tratados con el mismo nivel de profundidad por estos métodos, resultando así poco adecuados para el desarrollo de software según la perspectiva de la IS. Esta perspectiva es la adoptada por la mayor parte de organizaciones de construcción de software, debido a que los aspectos más relevantes para la IS son de gran importancia para dichas organizaciones. Entre ellos se encuentra el objetivo de contar con un proceso de desarrollo definido, y la IPO no considera los procesos de desarrollo con el mismo grado de formalidad y sistematización que la IS. A modo de ejemplo, algunos autores de la IPO, como [Shneiderman, 98] y [Nielsen, 93], no estructuran las labores destinadas a tratar la usabilidad en actividades (según se entienden en la IS), de tal forma que hay técnicas de usabilidad que no están explícitamente asignadas a actividades en la literatura. A pesar de la menor formalización del proceso en la IPO, la caracterización de un proceso de desarrollo centrado en el humano que define el estándar ISO 13407, define las normas básicas que deben cumplirse para tratar adecuadamente la usabilidad y, por tanto, puede tomarse como base para la mejora de los procesos de desarrollo de la IS en lo que respecta a la usabilidad.

Los métodos de la IPO no pueden tomarse como base para la integración de la usabilidad en el proceso de desarrollo porque obligan al acomodo de la construcción de software a su perspectiva del desarrollo. Esto es así porque estos métodos se presentan como un proceso paralelo al de la IS, el cual tiene puntos de conexión específicos con el proceso de desarrollo general. Este planteamiento no sería problemático si no fuera porque temas clave del desarrollo como son los requisitos, son incluidos en el proceso de usabilidad por autores IPO como [Mayhew, 99], considerando de esta forma que el proceso de desarrollo general no debe ocuparse de dichos temas, sino tomar los resultados ofrecidos por las actividades de usabilidad. Este enfoque está enfrentado al de la IS, por lo que la integración resulta extremadamente compleja, puesto que acomodar todo el proceso de la IS a los condicionantes que imponen dichos métodos puede producir que otros atributos de calidad de los que se ocupa la IS queden desatendidos. Para la IS resulta mucho más adecuado acomodar sus procesos para tratar un atributo más de calidad (la usabilidad), que adoptar el enfoque de métodos que tratan correctamente un atributo concreto y tener que ampliarlo para que traten adecuadamente el resto de atributos de calidad.

Podemos concluir que la IPO y la IS se intersectan profundamente, por lo que es preciso estudiar cómo acomodar el enfoque centrado en el usuario en el proceso de desarrollo general de forma detallada, no sólo como enlaces puntuales entre dos procesos que transcurren de forma paralela.

1.2 Problema de Investigación Abordado

1.2.1 Identificación del Problema

La usabilidad forma parte de las descomposiciones de la calidad en atributos desde los años 70, como ya se ha indicado, pero es recientemente cuando se está destacando como atributo de especial relevancia en el desarrollo de software. Glass indica que "la gran importancia de las interfaces gráficas de usuario actualmente dice mucho de la importancia de la usabilidad" [Glass, 03]. Según Hakiel "la mayoría de organizaciones de desarrollo de software reconocen la relevancia de la usabilidad en el éxito de los productos software" [Hakiel, 97]. En contraste con la relevancia otorgada a la usabilidad, nos encontramos con una falta de atención por la usabilidad en la educación y práctica de la IS. Así, Larman sostiene que "no hay probablemente ninguna otra técnica con mayor disparidad entre su importancia para el éxito del desarrollo de software y la falta de una atención y educación formal que la ingeniería de usabilidad y el diseño de la IU" [Larman, 02].

Diez años atrás, la usabilidad era raramente considerada en otras organizaciones de desarrollo de software aparte de aquellas para las que la usabilidad era un atributo del producto software con importancia capital, y en ocasiones incluso el único en el que se centraban. En estas organizaciones el desarrollo se realiza según el enfoque centrado en el usuario y los desarrolladores tienen una formación mayoritaria en IPO. Sin embargo, este panorama está cambiando debido al creciente reconocimiento de la importancia de la usabilidad y debido a la identificación de las carencias en los métodos de la IS al respecto. Cada vez en más organizaciones se incorporan expertos en IPO como parte de los equipos de desarrollo o de departamentos de usabilidad creados al efecto. Según los resultados obtenidos de una encuesta realizada entre asistentes al congreso CHI-2000 (el congreso de la IPO con mayor importancia a nivel mundial), el uso de métodos de desarrollo centrado en el usuario tendrá un mayor impacto en los próximos años [Vredenburg, 02]. En la misma línea, Dumas y Redish predicen para la década actual "un crecimiento continuado en el interés en la usabilidad por parte tanto de los usuarios como de los directivos. Dicho crecimiento se producirá tanto por la expansión de los grupos de usabilidad existentes, como por la introducción de la usabilidad en más compañías" [Dumas, 99]. En el futuro, esta tendencia se verá incluso incrementada por la inclusión de un nuevo proceso de usabilidad en la primera enmienda al estándar ISO 12207 de procesos de desarrollo software [ISO12207_1, 02]. En el año 2002, se publicó esta primera enmienda, que incluye dicho proceso de usabilidad entre los procesos de soporte al ciclo de vida. Así, la usabilidad pasa a ser algo de lo que preocuparse a lo largo de todo el desarrollo, de modo similar al resto de atributos de calidad. Aquellas organizaciones que aleguen conformidad con el estándar deberán gestionar la usabilidad en sus procesos de desarrollo dando así contenido a este proceso. Será preciso incluir expertos de usabilidad en la plantilla de las organizaciones de desarrollo de software, o bien formar a los equipos de desarrollo en la aplicación de prácticas de usabilidad.

El interés creciente que despierta la usabilidad y las técnicas de la IPO entre las organizaciones de desarrollo de software, junto con la creciente inclusión de expertos en usabilidad en la plantilla de dichas organizaciones, lleva a la disyuntiva de cómo integrar las técnicas de usabilidad con el resto de técnicas que se aplican en el desarrollo. Según Costabile "debemos integrar las nociones del campo de la IPO en la formación de los profesionales del software [...] Y de forma más importante, debemos integrar los métodos y técnicas de diseño de la IU en las metodologías de desarrollo de software estándar" [Costabile, 01]. En esta misma línea, Shneiderman sostiene que "cualquier metodología de diseño centrada en el usuario debe encajar con la metodología de IS usada" [Shneiderman, 98]. El proceso de usabilidad nombrado en la primera enmienda al estándar ISO 12207 [ISO12207_1, 02] incluye como actividad la implementación del proceso, la cual consiste básicamente en

especificar y gestionar cómo las actividades de usabilidad se van a desarrollar y cómo van a encajar con el resto de actividades del proyecto. Así, podemos afirmar que esta enmienda al estándar genera una necesidad de guías acerca de cómo realizar tal labor de integración.

El presente trabajo de investigación aborda el problema de la integración de las técnicas y actividades de usabilidad en el proceso de desarrollo software, en una organización de desarrollo que tiene un proceso definido y que sigue mayoritariamente el enfoque de la IS.

1.2.2 Relevancia del Problema y Dificultades que Plantea

La integración de las técnicas y actividades de usabilidad en el proceso de desarrollo general es un objetivo perseguido desde ambos campos (IPO e IS). Desde la IS debido a la importancia que se le reconoce a la usabilidad como atributo de calidad, y desde la IPO por el impulso que supone a los temas clave de la disciplina la inclusión de sus prácticas en los procesos de desarrollo generales.

La identificación de cuál es el mejor modo de abordar el objetivo de la integración es un tema abierto, señalado en la literatura como elemento clave que es preciso investigar. Así, Mayhew identifica entre las futuras contribuciones al campo de la ingeniería de la usabilidad la integración del ciclo de vida de la ingeniería de usabilidad con el ciclo de vida de desarrollo software existente [Mayhew, 99]. En concreto, indica que "trabajos específicamente dirigidos a integrar las tareas del ciclo de vida de la ingeniería de usabilidad en las metodologías existentes de la IS [...] serían extremadamente productivos" [Mayhew, 99]. En referencia a esta cuestión añade, sin embargo, que "no tenemos todavía ideas específicas sobre *exactamente* cómo integrar más eficiente y suavemente las tareas del ciclo de vida de la ingeniería de usabilidad con las tareas del desarrollo de software estándar" [Mayhew, 99] (el resaltado en *italica* pertenece al original). Sobre este tema, Hix y Hartson destacan que "los canales de interconexión entre aquellas actividades de desarrollo de la interfaz, y aquellas que no corresponden a la interfaz, son esenciales para un equipo de desarrollo integrado" [Hix, 93]. En referencia a dichos canales de comunicación, Hix y Hartson aseguran que no están bien comprendidos, y que "se necesita investigación para comprender mejor y dar soporte a las necesidades reales de comunicación de dicho proceso complejo" [Hix, 93].

Se ha identificado que los métodos IPO, cuando se aplican, están desacoplados del proceso de desarrollo general. Así lo aseguran Metzker y Reiterer cuando indican que "los modelos de proceso centrados en el humano están desacoplados del proceso de desarrollo software general" [Metzker, 02]. Acoplar ambos procesos resulta una tarea ardua, pues según Jokela indica, "el desafío de los especialistas en usabilidad consiste en integrar su trabajo en el proceso central llevado a cabo por los diseñadores. Como está reconocido ampliamente, se trata de un verdadero reto en muchos casos" [Jokela, 02].

La IS ha sido hasta ahora ajena a las prácticas de usabilidad del campo de la IPO, según apunta Coutaz cuando asegura que "las herramientas, métodos y notaciones IPO están disponibles pero no incorporadas en la práctica de la IS" [Coutaz, 94]. Esta carencia en los equipos de desarrollo de software ha sido detectada también por Seffah y Andreevskaia, quienes indican que "una barrera para una práctica más amplia del enfoque de diseño centrado en el humano es que su estructura y técnicas son todavía relativamente desconocidas, infrutilizadas, difíciles de dominar, y esencialmente no están bien integradas en los equipos de desarrollo" [Seffah, 03]. Este desconocimiento por parte de los desarrolladores tiene causas diversas, pero una de las principales reside en que la presentación de las actividades y técnicas de usabilidad a los desarrolladores no ha sido la adecuada hasta el momento. A este respecto, resulta relevante el proyecto realizado por Bevan y Earthy para evaluar la madurez en usabilidad de dos compañías (*Inland Revenues/EDS*, e *Israel Aircraft Industries*) e introducir métodos simples centrados en el usuario en su desarrollo. En las conclusiones de este

proyecto, recogidas en [Bevan, 01a], se indica que las dos compañías tuvieron dificultades para comprender los beneficios potenciales de los métodos centrados en el usuario, los cuales difieren en naturaleza de las otras actividades de la IS; y en concreto Bevan y Earthy aseguran que "todavía no está claro cómo presentar mejor las actividades de diseño centrado en el usuario propuestas de forma que puedan ser comprendidas y apreciadas por los diseñadores y desarrolladores" [Bevan, 01a]. Este tipo de problema ha sido identificado también por Hakiel, quien asegura que "las descripciones de elementos ofrecidos por la IPO en términos de actividades y métodos no son directamente comprensibles para los jefes de proyecto software. De forma similar, las descripciones de actividades y métodos IPO dan escasas indicaciones de cómo tienen que gestionarse en referencia a las etapas y entregables en el ciclo de vida de desarrollo de un producto software" [Hakiel, 97].

El desconocimiento y las carencias en comprensión del enfoque centrado en el usuario por parte de desarrolladores formados en la IS, se deben también a la brecha terminológica existente entre la IPO y la IS. Nielsen asegura que "a los desarrolladores y los jefes de proyecto software a menudo les intimida la terminología y las extrañas instalaciones de laboratorio empleadas por algunos especialistas en usabilidad" [Nielsen, 93]. Según Seffah y Metzker, "las habilidades de diseño de la interfaz de usuario son a menudo improvisadas y apreciadas como poco importantes por desarrolladores y jefes de proyecto [...] Ayudar a los estudiantes de informática y de IS a comprender y dominar las técnicas de desarrollo centrado en el usuario en su propio lenguaje y contexto cultural es una exitosa estrategia pedagógica" [Seffah, 04].

La diferencia terminológica IPO-IS se observa en aquello que la IPO considera como su especialidad: El diseño de la IU. En la IS la actividad de Diseño de la IU se refiere únicamente al diseño de los elementos concretos que van a formar parte de la IU y su comportamiento asociado. No suele incluir ningún tipo de actividad relacionada con la ingeniería de requisitos. Por otra parte, un principio ampliamente aceptado en la IS es la necesidad de separar la parte del sistema que se encarga de los elementos de la IU de la parte correspondiente a la lógica de negocio. La aplicación estricta de este principio hace que el diseño de la IU no esté relacionado directamente con el diseño de los procesos internos del sistema. En su parte gráfica, el diseño de la IU está asociado a profesionales del diseño gráfico, los cuales siguen unos principios estéticos. Este concepto del diseño de la IU es el que lleva a considerarlo en la IS no como parte de la disciplina sino como una disciplina relacionada [SWEBOK, 04]. Por el contrario, desde el campo de la IPO, el diseño de la IU tiene un significado más amplio. El diseño de la interacción según Hix y Hartson implica "las acciones de los usuarios, retroalimentación, apariencia de la ventana, y tareas de los usuarios, y tiene que ver también con la funcionalidad, con la secuencia de acciones, contenido y acceso a la información, así como con los detalles del diseño de los objetos de la interfaz, la disposición de la ventana y los estilos de interacción" [Hix, 93]. En concreto, la secuencia de acciones, el contenido y acceso a la información pueden afectar profundamente a los procesos internos del sistema. Nótese que a pesar de referirse al diseño de la interacción (un término que usaremos en este trabajo de investigación para evitar la confusión con la visión clásica del diseño de la IU en la IS), el título del trabajo de Hix y Hartson es "Desarrollando Interfaces de Usuario", lo cual puede llevar a un desarrollador a minusvalorar lo que dicho trabajo puede aportar a su labor de construcción de software. De forma general, podemos afirmar que la visión de qué implica diseñar la IU para la IPO abarca un campo mayor de actuación que la versión restringida que tiene la IS del diseño de la IU. Las diferencias terminológicas no terminan en la diferencia de apreciación de qué significa diseñar la IU, pero resulta un indicador relevante de la existencia de obstáculos terminológicos para el objetivo de la integración de las prácticas de usabilidad en el desarrollo.

Los métodos IPO, como hemos indicado, están descritos de forma mayoritaria para su aplicación como un proceso paralelo al de la IS y relativamente independiente. Así, cuando se aplican en un desarrollo, muchas veces su aplicación se reduce a momentos puntuales del

ciclo de vida, reduciéndose de esta forma su efectividad de cara a mejorar la usabilidad del producto final. Un enfoque centrado en el usuario requiere una preocupación por la usabilidad a lo largo de todo el desarrollo. A este respecto Gulliksen indica que "de varios estudios de diseño centrado en el usuario en la práctica, podemos observar dificultades en la adopción del diseño centrado en el usuario a lo largo de todo el ciclo de vida" [Gulliksen, 99].

Podemos concluir que la complejidad que conlleva la definición de cómo se integran las prácticas de usabilidad en el proceso se debe principalmente a los siguientes obstáculos identificados:

- Los procesos de usabilidad y de desarrollo están desacoplados sin existir puntos de convergencia claramente definidos.
- No se sabe cómo presentar mejor a los desarrolladores las técnicas y actividades de usabilidad que deben formar parte de su proceso.
- Existen importantes diferencias de terminología entre la IS y la IPO.
- Existe una falta de adopción de un enfoque centrado en el usuario a lo largo de todo el desarrollo.

1.3 Aproximación a la Solución

La usabilidad es el atributo de calidad en el que centran su atención los métodos y técnicas IPO. La IS considera la usabilidad como uno de los atributos de calidad y, aunque dicho criterio está cobrando más relevancia recientemente, esto no significa que sea el único atributo de calidad que hay que tener en cuenta en el desarrollo de software. El objetivo de que el sistema a desarrollar alcance un nivel de usabilidad adecuado requiere de la aplicación de técnicas de usabilidad, pero este objetivo debe compaginarse con otros objetivos clásicos de la IS como son la eficiencia, mantenibilidad, fiabilidad, etc. Así, las técnicas de usabilidad deben aplicarse de forma coordinada con las actividades y técnicas de la IS, para lograr no sólo un sistema usable, sino con todas las características de calidad.

Una posible aproximación para la integración de las técnicas de usabilidad en el proceso de desarrollo, consistiría en la elección de un modelo de proceso software estándar (como pudiera ser el estándar IEEE 1074 [IEEE1074, 98]), al cual se incorporarían las técnicas de usabilidad. Este enfoque puede resultar atractivo desde un punto de vista de modelado del proceso. Sin embargo, ligar la integración de la usabilidad en el proceso a un modelo de proceso específico puede resultar demasiado restrictivo para muchas organizaciones de desarrollo de software, lo cual podría llevar a una limitación en el uso de la solución propuesta en el presente trabajo de investigación. Organizaciones de desarrollo de software que tengan un proceso de desarrollo consolidado, no es probable que accedan a cambiarlo por uno completamente diferente, aun cuando consigan beneficios en un criterio de calidad concreto (en este caso, la usabilidad). Por tanto, **no** basaremos nuestra propuesta en la selección de un proceso genérico que obligue a la organización a tener implantado un modelo de proceso específico para utilizar nuestros resultados de investigación.

Por el contrario, para conseguir nuestro objetivo adoptaremos un enfoque flexible, basado en la enumeración de las condiciones mínimas que tengan que cumplirse para considerar que un proceso de desarrollo software puede incorporar aspectos de usabilidad con garantías de éxito. La propuesta para mejorar el proceso con los aspectos de usabilidad se describirá, por tanto, de una forma suficientemente genérica como para ser aplicada a cualquier proceso que cumpla los prerequisites mínimos establecidos. Consideramos que esta aproximación

incrementará la aplicabilidad práctica de los resultados de esta investigación. El atractivo para una organización de desarrollo de software yace en el hecho de que no tiene que abandonar el proceso de desarrollo que tenga establecido para mejorar en la usabilidad de sus productos, puesto que es suficiente con modificarlo incluyendo técnicas de usabilidad según la información que ofrece el marco de integración que constituye la solución propuesta.

Así, el escenario para el cual vamos a elaborar nuestra solución contempla una organización de desarrollo de software con las siguientes características:

- Una cultura de desarrollo basada en los principios de la IS en cuanto a los conceptos y terminología manejada.
- Un conocimiento escaso o nulo de conceptos y técnicas IPO por parte de la mayoría de los integrantes de los equipos de desarrollo.

Consideramos que este escenario, lejos de ser restrictivo, responde a las características de la mayoría de organizaciones de desarrollo software de tamaño mediano o grande.

Para que ingenieros software puedan adoptar una perspectiva centrada en el usuario, y puedan comprender cómo y cuándo aplicar técnicas de usabilidad, será necesario ofrecerles un marco de integración que describa las actividades y técnicas de usabilidad en el proceso de desarrollo, según los conceptos y terminología de la IS. También será necesario hacer una selección de las técnicas IPO que son más cercanas a la IS para que el resultado esperado de integración pueda alcanzarse con mayores probabilidades de éxito.

La definición del marco de integración de la usabilidad en el proceso de desarrollo que presenta el presente trabajo de investigación se ha dividido en los siguientes capítulos:

- El primer capítulo introduce el trabajo de investigación y es el presente capítulo.
- La revisión del estado del arte referente al problema de investigación planteado se detalla en el capítulo 2.
- En el capítulo 3 se aborda el planteamiento del problema y el enfoque que se ha tomado para aportar una solución.
- En el capítulo 4 se incluye un estudio de la literatura IPO para obtener un conjunto representativo de actividades de usabilidad debido a que, como se ha comentado, distintos autores proponen conjuntos distintos de actividades. A continuación se presenta un estudio de las técnicas de usabilidad, clasificadas según las actividades identificadas, y entre ellas se realiza una selección de aquellas que mejor pueden servir al propósito de integración en el proceso de desarrollo.
- Para que los desarrolladores puedan conocer a qué tipo de actividad corresponde cada técnica de usabilidad, se va a relacionar cada actividad de usabilidad con la correspondiente actividad según la terminología de la IS. Dichas relaciones sirven de base para la asignación de las técnicas de usabilidad a los distintos tipos de actividades del desarrollo, según se describen en la IS. El contenido de este punto se desarrolla en el capítulo 5.
- Se ha realizado, así mismo, un estudio sobre las características que debe cumplir un proceso para poder incorporar técnicas de usabilidad. Dichas condiciones se reducen a estar basado en un enfoque iterativo. Puesto que la solución va a requerir un desarrollo iterativo como proceso base sobre el que integrar las técnicas de usabilidad, y los ciclos de los que se compone un desarrollo iterativo no son todos iguales, creemos necesario transmitir a los desarrolladores la información sobre el

mejor momento de aplicación de cada técnica de usabilidad. Por momento entendemos una agrupación de ciclos iterativos consecutivos que están divididos por hitos genéricos. En el capítulo 6 se detalla este estudio sobre el mejor momento de aplicación de cada técnica de usabilidad en un desarrollo iterativo, precedido por el estudio sobre las condiciones a cumplir por un proceso candidato a la integración de la usabilidad.

- A modo de resumen de la solución planteada, el capítulo 7 recoge toda la información destilada en los capítulos anteriores en forma de marco de integración. El marco se detalla desde las distintas perspectivas que pueden interesar a un desarrollador que quiera aplicar la solución propuesta. En este capítulo también se indica cuál es el uso esperado del marco propuesto y se introduce la herramienta web desarrollada para ofrecer una aplicación del marco de integración más accesible a los desarrolladores.
- En el capítulo 8 se presenta la aplicación de la solución propuesta y la evaluación de los resultados obtenidos.
- Finalmente, en el capítulo 9 se detallan las conclusiones obtenidas de la realización del presente trabajo de investigación y las líneas futuras que se plantean.
- Tras la bibliografía consultada en la realización del presente trabajo de tesis, en los anexos se incluye un resumen de las técnicas propuestas por cada fuente consultada para el estudio de técnicas del capítulo 4 (Anexo A), los cuestionarios utilizados para evaluar la aplicación de la solución detallada en el capítulo 8 (Anexos B a E), y un índice de acrónimos para facilitar la lectura del presente documento (Anexo F).

1.4 Publicación de Resultados

Diversas partes del presente trabajo de investigación han sido presentadas en distintos foros de carácter científico, según se detalla a continuación.

Debido a la novedad del tema de la usabilidad en el campo de la IS, y a la laguna de conocimientos existente en dicha comunidad, algunos resultados preliminares sobre la relación entre la IS y la usabilidad y los conceptos básicos de usabilidad se publicaron en una fase muy temprana de la investigación. Concretamente, han sido publicados en las siguientes revistas y actas de congresos:

- "Principios Básicos de Usabilidad para Ingenieros Software"
V Jornadas de Ingeniería del Software y Bases de Datos 2000 [Ferré, 00]
Este artículo es la primera aproximación al problema en forma de introducción al tema de la usabilidad dirigido a ingenieros software. Únicamente presenta una visión general de las prácticas de usabilidad que puede ser de utilidad a los ingenieros software.
- "Usability Basics for Software Developers"
IEEE Software [Ferré, 01a]
De forma más elaborada que el anterior, este artículo introduce la idea de la usabilidad y el proceso para conseguirla, para una audiencia de desarrolladores software. Supone un primer intento de mostrar las posibilidades que las técnicas de usabilidad pueden ofrecer al desarrollo de software general, y de cómo acomodarlos en el desarrollo.

- "Incorporating Usability into an Object Oriented Development Process"
Workshop *Usability Throughout the Entire Systems Development Lifecycle*
INTERACT 2001 [Ferré, 01b]

Este artículo presenta una aproximación temprana que consistió en añadir actividades de usabilidad a un proceso de desarrollo orientado a objetos basado en el método de Larman. Esta aproximación fue abandonada tras contrastar su viabilidad con el mundo empresarial, por estar ligada a un proceso de desarrollo concreto.

El presente trabajo de investigación ha sido llevado a cabo en el marco del proyecto STATUS (*Software Architecture that Supports Usability*), financiado por la Comisión Europea (IST - 2001 - 32298) [STATUS, 01]. El proyecto incluye entre sus fines, y como objetivo del WP5 (Paquete de Trabajo 5), la definición de guías metodológicas para la integración de las técnicas de usabilidad en el proceso de desarrollo. Parte de los resultados de investigación del presente trabajo constituyen las recomendaciones que se han presentado como resultado de dicha línea de investigación del proyecto. Entre los socios del proyecto se encuentran dos compañías de desarrollo de software: *The Information Highway Group* (IHG) y LogicDIS. Estas compañías han participado en el presente trabajo de investigación aplicando resultados preliminares y enriqueciendo la solución propuesta con sus comentarios y con la definición de los requisitos de integración de usabilidad reales que tienen que afrontar en su papel como organizaciones de desarrollo de software.

Los siguientes entregables del proyecto corresponden a los capítulos indicados entre paréntesis del presente trabajo de investigación:

- Deliverable D.5.1 Selection of the Software Process and the Usability Techniques for Consideration. 2002. [Ferré, 02a]
En este entregable se incluye el estudio de actividades de usabilidad, y el de técnicas junto a una versión preliminar de la selección de las técnicas más apropiadas según los objetivos fijados (capítulo 4).
- Deliverable D.5.2 Specification of the Software Process with Integrated Usability Techniques. 2002. [Ferré, 02b]
Este entregable es una continuación del anterior, e incluye la asignación de actividades de usabilidad a actividades del desarrollo general, la asignación de técnicas de usabilidad a actividades del desarrollo, el estudio sobre momentos de aplicación en un desarrollo iterativo, y una versión preliminar del resumen de los resultados presentados para una audiencia compuesta por desarrolladores (capítulos 5, 6 y 7).
- Deliverable D.6.6. Final Results on the Integrated Software Process. 2004. [Ferré, 04c]
Debido al refinamiento de la propuesta que se produjo durante el desarrollo del proyecto, este entregable recoge la versión final de la propuesta de integración en el proceso, como actualización de la información reflejada en los entregables D.5.1 y D.5.2. Así, recoge el estudio de actividades IPO, la clasificación y caracterización de técnicas IPO, la selección de las más idóneas para el objetivo de integración, su asignación a tipos de actividades del desarrollo y el estudio sobre momentos de aplicación de las técnicas (capítulos 4, 5 y 6).
- Deliverable D.1.7 Main Achievements and Findings of the Project. 2004. [Folmer, 04]
En este entregable se recogen las principales aportaciones que el proyecto ha realizado. El capítulo 5 está dedicado al resultado del WP5, y su contenido es el marco de integración de la usabilidad en el proceso de desarrollo que constituye el resultado del presente trabajo de investigación (capítulo 7).

Finalmente, los resultados de este trabajo de investigación, tal y como se muestran en esta tesis doctoral, han sido publicados en los siguientes congresos y revistas:

- "Framework for Integrating Usability Practices into the Software Process"
6th Int. Conf. on Product Focused Process Improvement (PROFES 2005) [Ferré, 05c]
(aceptado)
Esta ponencia es una versión resumida de la solución propuesta en el presente trabajo de investigación (capítulos 4 a 7).
- "Which, When and How Usability Techniques and Activities Should Be Integrated"
En *Human-Centered Software Engineering - Integrating Usability in the Development Process*. 2005 (en prensa) [Ferré, 05a]
Este capítulo es una versión resumida de la solución completa, excepto la parte final del marco de integración con sus vistas (capítulo 7). La parte dedicada a cuándo integrar técnicas IPO en el desarrollo es una versión preliminar del estudio sobre los momentos de aplicación de las técnicas (capítulo 6).
- "Integration of HCI Practices into Software Engineering Development Processes: Pending Issues"
En *Encyclopedia of Human Computer Interaction*. 2005 (en prensa) [Ferré, 05b]
Este capítulo recoge una versión resumida del estado de la cuestión y del planteamiento (capítulos 2 y 3).
- "Integración de la IPO en el Proceso de Desarrollo de la Ingeniería del Software: Propuestas Existentes y Temas a Resolver"
V Congreso Interacción Persona-Ordenador (Interacción 2004) [Ferré, 04a]
En este artículo se publica una versión resumida del estado del arte (capítulo 2).
- "Improving Software Engineering Practice with HCI Aspects"
1st. Int. Conf. on Software Engineering Research and Applications (SERA 2003) Selected Revised Papers. Lecture Notes in Computer Science. 2004 [Ferré, 04b]
Este artículo presenta el estudio de actividades y técnicas de usabilidad, una selección preliminar realizada sobre las técnicas, y parte de la asignación entre actividades de usabilidad y actividades del desarrollo general (capítulo 4 y parte del capítulo 5).
- "Integration of Usability Techniques into the Software Development Process"
Workshop *Bridging the Gaps Between Software Engineering and Human-Computer Interaction*, ICSE 2003 [Ferré, 03]
El contenido de este artículo incluye la asignación de actividades de usabilidad a actividades del desarrollo general, y la asignación de técnicas de usabilidad a estas últimas. También incluye una versión preliminar del estudio sobre los momentos de aplicación de las técnicas (capítulos 5 y 6).

Capítulo 2

Estado de la Cuestión

2.1 Introducción

En el presente capítulo se describen las aproximaciones existentes a la integración de las técnicas y actividades de usabilidad en el proceso de desarrollo software, con el fin de analizarlas y realizar un estudio comparativo de sus virtudes y sus carencias.

Para poder realizar la comparación entre las distintas aproximaciones se van a considerar los siguientes criterios:

- **Grado de iteratividad del enfoque.** Un enfoque iterativo del desarrollo es imprescindible en cualquier proceso centrado en el usuario. De hecho, todas las propuestas de integración de la usabilidad en el proceso que se van a considerar son presentadas por sus autores como iterativas. Sin embargo, se van a distinguir las propuestas que adoptan un enfoque verdaderamente iterativo de aquellas que llevan implícito un enfoque de ciclo de vida en cascada, el cual es incompatible con un enfoque iterativo. Consideraremos que se adopta un enfoque de ciclo de vida en cascada cuando se abordan las tareas de análisis de requisitos con el objetivo de finalizarlas antes de pasar al diseño del sistema, y si se vuelve a retomar el análisis realizado es únicamente para corregir defectos. En el primer caso este criterio tomará el valor *iterativo*, mientras que en el segundo caso tomará el valor de *nominalmente iterativo*.
- **Nivel de descripción del proceso.** Un proceso de desarrollo puede definirse a distintos niveles de detalle. Algunas propuestas únicamente tratan las actividades de alto nivel, mientras que otras detallan las actividades de grano fino, las técnicas a aplicar en cada una y los productos que se obtienen. Los valores que puede tomar son *actividades detalladas* o *actividades de alto nivel*. Para dar valor a los criterios únicamente vamos a basarnos en la documentación públicamente disponible, lo cual es cierto también para este criterio en concreto que se refiere al nivel de detalle de la propuesta de integración.
- **Alcance del proceso de desarrollo propuesto.** La identificación de la usabilidad con la Interfaz de Usuario (IU) ha llevado en algunos casos a proponer que la integración se produzca únicamente en lo concerniente al diseño de la IU. En otros casos se centra únicamente en aspectos de evaluación, o en algún otro aspecto concreto del sistema. El valor para este criterio se asignará siguiendo la descripción que cada autor realice de su propuesta. Los valores que puede tomar este criterio son *sistema*

completo, diseño de la parte interactiva del sistema, diseño de la IU, actividades de diseño centrado en el humano, y evaluación.

- **Grado de integración de aspectos de usabilidad.** La integración entre actividades y técnicas de usabilidad y el resto de las actividades del desarrollo puede describirse a varios niveles, y también se puede ligar a un método, o familia de métodos en particular. Los valores que vamos a considerar para este criterio son los siguientes:
 - *La integración debe ser planificada:* La necesidad de integración con otras actividades es admitida, y se especifica que ésta debe ser planificada, pero no se dan indicaciones de cómo llevarla a cabo.
 - *Indicaciones generales:* Únicamente se ofrecen criterios genéricos para llevar a cabo la integración.
 - *Relación con actividades de alto nivel,* pero sin entrar en detalles. En el caso que tal relación se lleve a cabo con un método o enfoque particular éste se especificará.
 - Un caso particular de este nivel es cuando únicamente se considera que los *productos de usabilidad se alimentan como entradas a actividades* tradicionales del desarrollo.
 - *Relación con actividades detalladas* de un método en particular. En este caso se especificará el método con el cual se establecen dependencias.
 - *Detallada para un abanico amplio de proyectos.*
- **Adhesión a un enfoque centrado en el usuario.** Un enfoque centrado en el usuario se caracteriza por cumplir las siguientes premisas: Estar basado en un enfoque iterativo, la implicación activa de usuarios finales, conocimiento multidisciplinar¹ y una adecuada comprensión de los requisitos referentes al usuario y a las tareas que debe completar. De estas premisas, para el valor asignado a este criterio no se va a considerar la primera de ellas (estar basado en un enfoque iterativo), puesto que tiene su propio criterio: Grado de iteratividad del enfoque. Así, para este criterio los valores que puede tomar son *centrado en el usuario y parcialmente centrado en el usuario.*

Las aproximaciones que se presentan en este estudio se han organizado en tres categorías según el objeto principal de cada trabajo en el que están descritas. Así, en primer lugar se van a estudiar los estándares de proceso, para evaluar si la información que contienen es suficiente para poder llevar a cabo la integración de técnicas y actividades de usabilidad de forma apropiada. A continuación, se van a considerar los métodos IPO que describen su integración en procesos IS. Dichos trabajos tienen como finalidad principal la descripción de un método especialmente dedicado a la usabilidad, aunque incluyen apuntes con distinto nivel de detalle acerca de su integración con el proceso de desarrollo habitual. Finalmente, se abordan aquellas propuestas que se centran en la integración de aspectos de usabilidad en las prácticas de la IS.

2.2 Estándares de Proceso de Desarrollo Software

Como primer paso para la integración de técnicas y actividades de usabilidad en el proceso de desarrollo, se puede tomar como referencia un modelo de proceso, para evaluar si su definición permite tal integración y, en ese caso, si la descripción de los elementos del proceso facilita dicha integración. En esta sección se describen, primero, los dos estándares de

¹ El conocimiento multidisciplinar está presente en la mayor parte de las propuestas mediante la inclusión en el proceso de técnicas IPO que incorporan dicho conocimiento, por lo que únicamente se destacará esta condición en aquellas propuestas que no la cumplan.

proceso de desarrollo auspiciados por la asociación IEEE: ISO/IEC 12207:1995 (IEEE 12207.0 según la nomenclatura IEEE) [ISO12207, 95] e IEEE 1074-1997 [IEEE1074, 98]. A continuación, se analiza la primera enmienda al estándar ISO/IEC 12207: 1995 [ISO12207_1, 02], publicada en el año 2002, que aunque no está todavía aceptada por IEEE dentro de su cuerpo de estándares de IS, sí que es parte del cuerpo de estándares de ISO.

2.2.1 Estándar ISO/IEC 12207:1995

2.2.1.1 Descripción

En 1996, IEEE decidió adoptar el estándar ISO/IEC 12207 (*ISO/IEC International Standard: Information Technology. Software Life Cycle Processes*) [ISO12207, 95] como base para los estándares relacionados con el proceso de ciclo de vida en la colección de estándares de IEEE sobre ingeniería del software, incluyéndolo en dicha colección con el nombre IEEE 12207.0 [IEEE12207.0, 98]. La adopción del estándar ISO/IEC 12207 por parte de IEEE ha sido clave, pues a partir de dicha decisión ha empezado un proceso de revisión del resto de estándares de IS para asegurar la consistencia con el estándar ISO/IEC 12207. En concreto, IEEE adaptó su estándar 1074 [IEEE1074, 98], manteniéndolo como estándar aparte del estándar 12207.0 porque aporta una visión complementaria del proceso software: Mientras el estándar 12207.0 proporciona requisitos para los procesos, el estándar 1074 detalla los bloques básicos con los que construir un proceso software.

El estándar internacional ISO/IEC 12207 define un marco común para procesos de desarrollo software (procesos de ciclo de vida en la terminología del estándar), con una terminología precisa, que puede usarse como referencia por la industria del software. Contiene procesos, actividades y tareas a aplicar durante la adquisición de un sistema que contenga software, un producto software independiente, y un servicio software; y durante el suministro, desarrollo, operación y mantenimiento de productos software. El modelo de proceso que propone este estándar puede utilizarse también para definir, controlar, y mejorar los procesos de ciclo de vida software.

El estándar está redactado de manera que el modelo de proceso descrito pueda particularizarse para cada proyecto. Por tanto, el nivel de detalle no es muy grande, para cada actividad descrita se enumeran los principios generales que deben regir la realización de ese tipo de actividad. No prescribe el nombre, formato ni contenido de la documentación a generar por cada actividad.

En la Figura 2.1 se muestra la estructura de procesos de este estándar internacional, dividida en tres tipos de procesos: Principales, de soporte y organizacionales. Cada uno de estos tipos de procesos se describen a continuación:

- **Procesos Principales:** Comprenden cinco procesos que prestan servicio a las principales partes implicadas en el desarrollo de software. Estas partes implicadas de mayor relevancia son el adquirente, el suministrador, el desarrollador, el operador y el mantenedor del producto software.
- **Procesos de Soporte:** Comprenden ocho procesos, los cuales dan soporte a otros procesos como una parte integral de los mismos, pero con distinto propósito, y contribuyen al éxito y a la calidad del proyecto software.
- **Procesos Organizacionales:** Comprenden cuatro procesos, los cuales son usados por una organización para establecer e implementar un estructura base formada a partir de procesos de ciclo de vida y personal, y de una mejora continua de dicha estructura y procesos. Normalmente se usan en un ámbito más amplio que el de proyectos

específicos, sin embargo la experiencia reunida de dichos proyectos contribuye a la mejora de la organización.

Nótese que en la figura la numeración de los tipos de procesos refleja el número de la cláusula del estándar en donde está descrito cada uno de ellos.

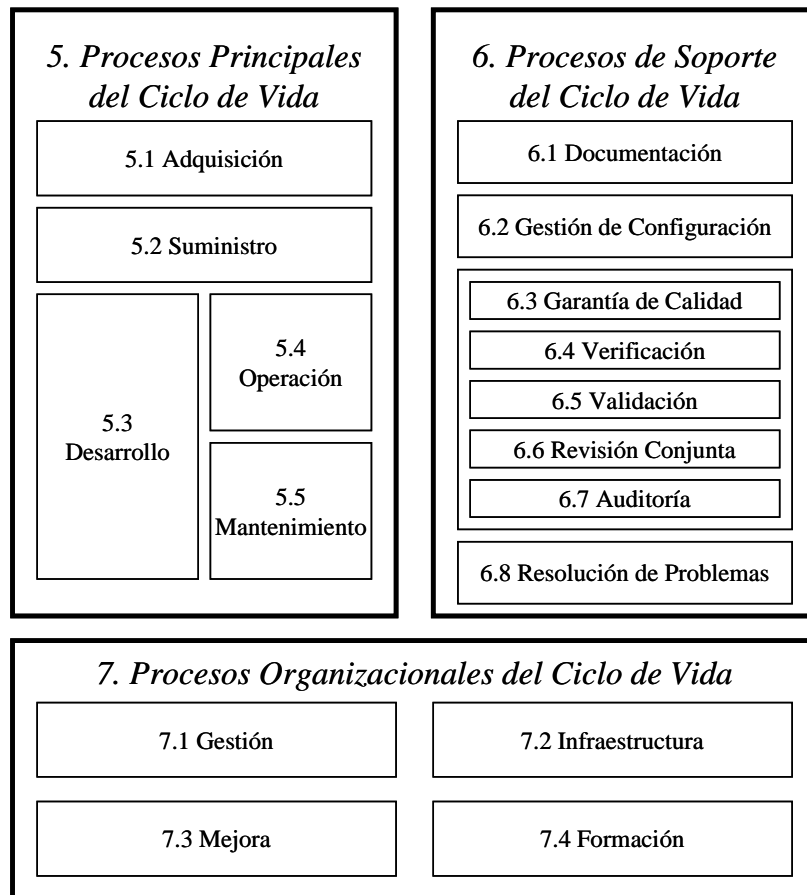


Figura 2.1 - Estructura de Procesos del Estándar ISO/IEC 12207

Los procesos descritos en el estándar deben particularizarse para cada proyecto software. En el Anexo A del estándar se describe el proceso de particularización que se debe seguir, mientras que en el Anexo B se ofrecen una serie de recomendaciones de carácter no obligatorio acerca de las cuestiones a tener en cuenta en la particularización. Este Anexo B transmite el carácter flexible que este estándar tiene, pues en la particularización es donde el estándar puede acomodarse para dar solución a proyectos de carácter muy distinto. Dicho anexo plantea, por ejemplo, que, para productos software de tamaño pequeño o de carácter no crítico, un control de la gestión exhaustivo no se justificaría en un análisis coste-beneficio, a diferencia de proyectos cuyo producto software sea de carácter crítico o de gran tamaño.

2.2.1.2 Análisis

En los procesos descritos en el estándar ISO/IEC 12207 no se especifica el papel de las prácticas de usabilidad más que de forma tangencial.

Al tratarse en el Anexo C del estándar las relaciones entre los distintos procesos del ciclo de vida y sus interrelaciones desde las distintas perspectivas de uso del estándar, únicamente se tiene en cuenta a los usuarios en el proceso de operación. El proceso de operación está orientado al usuario porque contiene las actividades y tareas del operador del sistema (es

decir, la persona que operará con el software), y prescribe que se establezcan mecanismos para la gestión de las peticiones de cambios por parte de los usuarios, que se realicen pruebas del sistema en el entorno de operación, y que se establezcan los procedimientos y mecanismos de soporte a los usuarios del sistema. Así pues, en esta parte del estándar, los usuarios únicamente se consideran como agentes que pueden pedir cambios a tener en cuenta en la etapa de mantenimiento del sistema, permaneciendo pasivos hasta entonces.

En el proceso de suministro, en la actividad de planificación, la tarea 5.2.4.5, apartado j, establece que un tema a considerar en la planificación del proyecto es la participación de usuarios. El estándar aclara que tal participación se refiere a ejercicios de establecimiento de requisitos, demostraciones de prototipos y evaluaciones. Se desprende de esta definición que el estándar tiene un enfoque de la posible participación del usuario en el que éste cumple un papel pasivo en las actividades en las que participa.

En el proceso de validación, en la actividad de validación, la tarea 6.5.2.3 especifica cómo desarrollar las pruebas de validación del producto software. En el apartado c de dicha tarea se dice que se debe: “Probar que usuarios representativos pueden llevar a cabo las tareas deseadas satisfactoriamente usando el producto software”. En esta tarea tendrían cabida todas las técnicas y actividades de evaluación de la usabilidad.

Aunque el estándar no aborda el tratamiento de la usabilidad del producto software más que de forma indirecta, su carácter flexible permite la inclusión de técnicas y actividades de usabilidad. No obstante, no se encuentra en el estándar ningún tipo de ayuda para aquellas organizaciones de desarrollo de software que quieran implementar las actividades de usabilidad.

2.2.2 Estándar IEEE 1074-1997

2.2.2.1 Descripción

El estándar IEEE 1074-1997 (*IEEE Standard 1074 for Developing Software Life Cycle Processes*) [IEEE1074, 98] está destinado al arquitecto de proceso, que debe definir el proceso software que se va a aplicar en una organización de desarrollo. El estándar define las actividades que deben realizarse, junto con sus entradas y salidas, aunque no define el formato de dichos documentos o resultados de la aplicación de las actividades. No prescribe un ciclo de vida en concreto, únicamente detalla las actividades a llevar a cabo sin establecer restricciones temporales más allá de las dependencias relativas a los documentos de entrada o salida de las distintas actividades.

En la Tabla 2.1 se pueden observar los grupos de actividades según la organización en secciones que sigue este estándar.

En la actividad *A.1.3.1 Gestión de riesgos*, la usabilidad se nombra como una de las posibles consideraciones que deben valorarse a la hora de gestionar los riesgos del proyecto. A pesar de que el grupo de actividades *A.2.1 Actividades de exploración de conceptos* son candidatas a la aplicación de técnicas de usabilidad, no se menciona dicha posibilidad. El grupo *A.2.2* de actividades de asignación del sistema se encarga de la asignación de funciones al sistema software, a la parte hardware (donde sea aplicable) y a las personas, siendo la asignación de funciones entre personas y máquina una actividad típica de los procesos IPO. Tanto este grupo de actividades como el resto que se refiere a actividades de ingeniería de requisitos (*A.2.1* a *A.2.3* y *A.3.1*), admitirían el uso de técnicas de usabilidad, pero tal posibilidad no se nombra en el estándar. Por ejemplo, prototipado se nombra en la actividad *A.3.1.1 Definir y desarrollar requisitos software*, como una posible técnica para la especificación de requisitos, pero no se especifica la posibilidad del uso de prototipos de baja fidelidad para validación de

requisitos, ni para educir nuevos requisitos por parte del usuario, mientras que ambas prácticas son habituales en los métodos IPO. Por otra parte, en la actividad A.3.1.2 *Definir requisitos de interfaz* se menciona explícitamente que la IU es crítica para determinar la usabilidad del sistema, y se indica que la definición de dicha interfaz no debe indicar sólo el intercambio de información sistema-usuario, sino también el modo en que un usuario maneja el sistema.

Tabla 2.1 - Agrupación de Actividades

Título de Sección	Cláusula	Grupos de Actividades
Gestión del Proyecto	A.1	Iniciación del Proyecto Planificación del Proyecto Monitorización y Control del Proyecto
Pre-Desarrollo	A.2	Exploración de Conceptos Asignación del Sistema Importación de Software
Desarrollo	A.3	Requisitos Diseño Implementación
Post-Desarrollo	A.4	Instalación Operación y Soporte Mantenimiento Retiro
Integral	A.5	Evaluación Gestión de Configuración del Software Desarrollo de Documentación Formación

En el grupo A.5.1 *Actividades de evaluación*, los ejemplos nombrados se centran en evaluación del código o modelos, y no nombran ningún tipo de evaluación de usabilidad. En concreto, donde podrían encajar las técnicas de evaluación de usabilidad es en las revisiones en el proceso (*In-process reviews*), sin embargo sólo se nombra que este tipo de revisión se realiza para eliminar defectos. Por tanto, no se toma en cuenta la evaluación formativa que está en la base de la concepción de las pruebas de usabilidad como parte del proceso de diseño [Nielsen, 93].

2.2.2.2 Análisis

El estándar IEEE 1074-1997 nombra la usabilidad únicamente como objetivo genérico sin llegar en ninguna actividad a identificar de qué manera se puede lograr que el sistema desarrollado alcance un nivel suficiente de usabilidad. Así mismo, no se nombra como entrada o salida de ningún proceso ningún tipo de documento de los necesarios según el campo de IPO, con la posible excepción de la declaración de necesidades (*Statement of need*), que con una interpretación libre podría acomodarse con un Diseño del Concepto del Producto.

Basándonos en la definición de los requisitos de interfaz de la actividad A.3.1.2, mencionada en la sección anterior, podríamos interpretar que la especificación de tareas de usuario que produce el análisis de tareas de la IPO puede formar parte del producto de esta actividad.

Las actividades descritas en el estándar se pueden implementar mediante la aplicación de prácticas de usabilidad, puesto que el estándar no lo impide. La evaluación (actividad A.5.1) se plantea en el estándar con un enfoque muy marcado a pruebas de código e inspecciones, dados los ejemplos que ilustran las descripciones de dichas actividades. A pesar de que no se nombra la evaluación formativa, puesto que las revisiones se circunscriben a la eliminación de

defectos, se puede interpretar de forma amplia e incluir prácticas de evaluación de usabilidad en las actividades de evaluación.

Consideramos que el contenido del estándar no asegura de ningún modo que la usabilidad del producto software creado siguiendo un proceso de desarrollo basado en este estándar haya sido tratada adecuadamente. Por otro lado, la integración de técnicas de usabilidad en un proceso de desarrollo basado en este estándar podría llevarse a cabo interpretando las actividades afectadas por usabilidad según el enfoque centrado en el usuario.

Finalmente, es preciso destacar que ayuda poco a la interpretación del estándar en un concepto amplio centrado en el usuario, el hecho de que el ejemplo de aplicación de su Anexo B se refiera a una versión simplificada del ciclo de vida en cascada, el cual no soporta un enfoque centrado en el usuario por no estar basado en el desarrollo iterativo. Como muestra, en dicho ejemplo la actividad de Diseño de la Interfaz de Usuario es la que comienza más tarde de todas las actividades de diseño. Se trata únicamente de un ejemplo, pero sirve para comprender la visión general de desarrollo de software que subyace en el estándar, que no es centrada en el usuario.

Por tanto, podemos concluir que el estándar no imposibilita un adecuado tratamiento de la usabilidad mediante la aplicación de las técnicas de usabilidad pertinentes, pero la concepción general que el estándar transmite no es nada favorable a tal integración, la cual debería realizarse con gran esfuerzo por parte del arquitecto de proceso, y con una interpretación elástica del estándar.

2.2.3 Enmienda 1 al Estándar ISO/IEC 12207

2.2.3.1 Descripción

La enmienda primera al estándar ISO/IEC 12207 [ISO12207_1, 02] aprobada por la organización de estandarización en el año 2002, se trata de una revisión provisional al estándar original, que acomoda los requisitos de estándares e informes técnicos actuales y en desarrollo del subcomité SC7 (correspondiente a la IS). En concreto, en la introducción de esta enmienda se indica que en el desarrollo del estándar ISO/IEC/TR 15504-2 identificaron ciertas cuestiones referentes a la granularidad del proceso definido en el ISO/IEC 12207, en el sentido de que resultaba difícil derivar un componente de valoración del proceso para tareas de evaluación y mejora del proceso. Esta enmienda resuelve el problema de granularidad y proporciona para los procesos un propósito y salidas para establecer un Modelo de Referencia del Proceso de acuerdo con los requisitos del ISO/IEC/TR 15504-2. Esta enmienda incluye también cuatro nuevos procesos (proceso de usabilidad, de gestión de *assets*, de gestión del programa de reutilización y de ingeniería del dominio), y renombra el proceso de formación por proceso de recursos humanos.

Esta enmienda al estándar incluye el nuevo proceso de usabilidad entre los procesos de soporte del ciclo de vida, y dicho nuevo proceso está basado en la información descrita en el estándar ISO 13407:1999 (Procesos de diseño centrados en el humano para sistemas interactivos) [ISO13407, 99], el cual se trata del estándar de referencia en proceso de desarrollo del campo de la IPO. El proceso de usabilidad de esta enmienda describe las actividades y tareas que debe desempeñar el rol de especialista en usabilidad en un proyecto de desarrollo de software. Cubre las actividades descritas en el estándar ISO 13407 y, por tanto, la descripción del proceso de usabilidad es bastante completa, incluyendo una mención a las principales actividades en un proceso centrado en el usuario.

Las actividades del proceso de usabilidad que incorpora esta enmienda al estándar ISO/IEC 12207 son las siguientes:

- **Implementación del proceso:** Consiste básicamente en especificar y gestionar cómo las actividades de usabilidad se van a desarrollar y cómo van a encajar con el resto de actividades del proyecto (el presente trabajo de investigación pretende ser una ayuda en este trabajo de implementación del proceso de usabilidad). Incluye, por ejemplo, la planificación de la participación de usuarios, el asegurar el enfoque centrado en el humano por parte del equipo de desarrollo completo y la selección de los métodos y técnicas centrados en el humano que se van a utilizar.
- **Diseño centrado en el humano:** Esta actividad consiste de varias tareas, dirigidas, primero a la especificación de los requisitos de las partes involucradas y organizacionales, de los objetivos del sistema, identificación de los usuarios y sus necesidades, y el establecimiento de los objetivos de calidad en uso (o requisitos de usabilidad). Las anteriores tareas deben realizarse por parte del especialista en usabilidad en asociación con el desarrollador, mientras que las siguientes son función del especialista en usabilidad: Identificación y documentación de las tareas del usuario, descripción de los atributos de los usuarios, y del entorno de uso. También incluye la ayuda por parte del especialista en usabilidad al desarrollador en las tareas de asignar funciones, producir el modelo de tareas, explorar el diseño del sistema y desarrollar prototipos. Finalmente, también incluye la especificación por parte del especialista en usabilidad de todo lo relativo a evaluación de usabilidad (evaluación de prototipos, evaluación en cuanto a las necesidades del usuario y organizacionales, etc.).
- **Aspectos humanos de estrategia, introducción y soporte:** Incluye tareas para el trabajo conjunto del especialista en usabilidad con los especialistas en marketing y estrategia, para analizar tendencias en los usuarios, elaborar estrategias de mercado y representar a las partes interesadas y usuarios en general. También incluye actividades para gestionar la introducción del sistema a desarrollar y la operación del mismo (gestión del cambio, personalización, formación de usuarios).

2.2.3.2 Análisis

La inclusión del proceso de usabilidad como parte del estándar de procesos de ciclo de vida del software es un paso adelante muy importante para la integración de las técnicas de usabilidad en el desarrollo de software. Esta inclusión significa que el tratamiento de la usabilidad pasa de ser una labor realizada en organizaciones de desarrollo de software donde se produce un interés especial hacia la usabilidad como atributo de calidad, a formar parte de cualquier proceso que alegue conformidad con el estándar.

Uno de las cuestiones más críticas en la incorporación de la usabilidad a los procesos de desarrollo es el rol del especialista en usabilidad. La visión tradicional de la IPO es que el tema de la usabilidad debe ser tratado, no por los desarrolladores, sino por un especialista que se integre con aquellos. Por una parte, este punto resulta difícil de satisfacer en organizaciones de desarrollo pequeñas y medianas. Por otra parte, como se ha indicado en el capítulo primero, en las actividades de Educación y Análisis de Requisitos, que son críticas en el desarrollo de software, es necesario un grado de integración muy amplio entre las actividades de usabilidad y las tradicionales de requisitos, por lo que el trabajo de un ingeniero de requisitos puede enriquecerse con las técnicas y actividades de usabilidad, sin necesidad de un enfoque mayoritariamente centrado en la usabilidad. Este estándar, a pesar de mencionar al especialista en usabilidad no lo define estrictamente, dejando abierta la posibilidad de estar refiriéndose a un rol que pueda compartirse con otros del desarrollo. Una posible interpretación sería aquella en la que parte de las responsabilidades del rol de especialista en usabilidad son asumidas por un “campeón de usabilidad” en cada equipo de desarrollo,

encargado de promocionar el enfoque centrado en el usuario (o centrado en el humano, según la terminología de ISO) y de asesorar en la aplicación de técnicas de usabilidad.

Tomando como base el proceso de usabilidad definido en este estándar se puede abordar la integración de técnicas de usabilidad en el proceso de desarrollo con una base normativa. Se trata de un estándar que incluye un enfoque centrado en el usuario, puesto que se basa en el estándar ISO 13407 para todo lo referente a usabilidad. De todas formas, el estándar no define cómo se deben integrar dichas actividades y técnicas de usabilidad con el resto de actividades del desarrollo. Tan solo indica que dicha integración debe tratarse en cada proyecto.

2.3 Métodos IPO

En esta sección se detallan trabajos que, a pesar de no ser propuestas de integración propiamente dichas, puesto que su objetivo es la descripción de un modelo o método de desarrollo centrado en el usuario, sí que constituyen propuestas de integración de forma indirecta, puesto que incluyen recomendaciones relevantes acerca de la integración del método descrito con el proceso de desarrollo.

2.3.1 Ciclo de Vida en Estrella

2.3.1.1 Descripción

Hix y Hartson [Hix, 93] proponen el ciclo de vida en estrella, un proceso de desarrollo centrado en el usuario que establece las principales actividades IPO que deberían llevarse a cabo. Dicho ciclo de vida está representado en la Figura 2.2. Se trata de un ciclo de vida centrado en la evaluación de la usabilidad. Los resultados de cada actividad se evalúan antes de continuar con la siguiente. No prescribe un orden particular ni una secuencia entre las actividades, lo cual destaca el carácter altamente iterativo de este enfoque. En particular, los autores indican que este tipo de proceso soporta iteraciones más cortas que las sugeridas por el ciclo de vida en espiral. Las restricciones de orden entre actividades son reducidas al mínimo, por ejemplo no es necesario haber establecido los requisitos para abordar tareas de diseño.

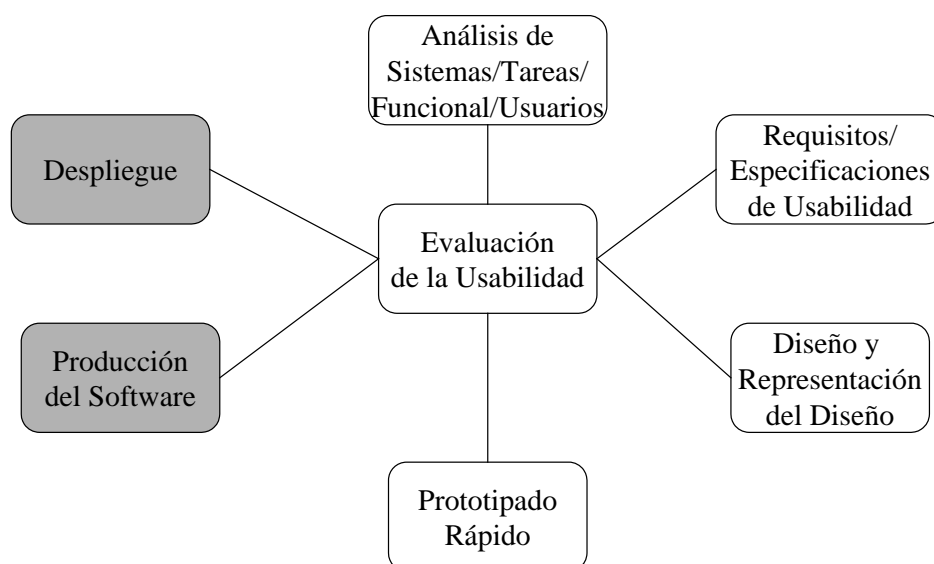


Figura 2.2 Ciclo de Vida en Estrella

En la Figura 2.2 aparecen con fondo blanco las actividades que los autores consideran como parte de su área de interés, puesto que son las actividades relacionadas con el desarrollo de la interacción con el usuario en su parte comportamental. Las actividades de Producción del Software o implementación, y Despliegue tienen fondo gris porque los autores las consideran como actividades que pertenecen al dominio construccional, en el sentido de la construcción o manejo de un elemento software.

Las tareas relativas al análisis de sistemas incluyen: Análisis de la necesidad, de usuarios y de tareas, y asignación de funcionalidad. Ninguna de ellas es explicada en profundidad.

En cuanto al diseño, sí que se detallan guías de diseño y los estilos de interacción, en cuanto al diseño como producto. Sin embargo, en cuanto al diseño como proceso, aparte de admitir que es una de las actividades de desarrollo menos comprendidas, únicamente se introducen las tareas de diseño del concepto del producto y del diseño inicial de escenarios (*initial scenario design*, un prototipado inicial de la IU), sin explicarlos en profundidad. La especificación de la interacción usuario-sistema sí que recibe una gran atención por parte de los autores, basada en su notación particular: UAN (*User Action Notation*). UAN es un tipo de modelado cognitivo de tareas, similar a las técnicas de la familia GOMS, que permite representar formalmente una IU.

El tema de las especificaciones de usabilidad es tratado también en profundidad, relacionado con la evaluación formativa de la usabilidad a través de test con usuarios reales. También se abordan las distintas variantes de la técnica de prototipado, como base del enfoque iterativo.

Los autores abogan por una aproximación metódica al desarrollo de la IU. Para que el diseño de la interacción quede especificado con detalle, proponen el uso de la notación UAN, para que a la hora de programar la IU no quede ningún fleco suelto que permita la construcción de una IU que no siga las consideraciones de diseño hechas previamente. Combinan los esfuerzos en cuanto a especificación de la IU con los esfuerzos de pruebas de usabilidad con usuarios reales, para reconducir el diseño de la interacción.

En cuanto a la relación del desarrollo de la IU con el resto de actividades, detallan las vías de comunicación que deberían relacionar ambos tipos de actividades. El planteamiento se basa en la presentación de las actividades de desarrollo como dos caminos paralelos: Por un lado las actividades relacionadas con el desarrollo de la IU, y por el otro el desarrollo de la parte de software no-interfaz, como se puede observar en la Figura 2.3. Hay dos actividades compartidas: Análisis de sistemas, que alimenta con requisitos a ambos tipos de actividades, y la evaluación y pruebas basados en el usuario (tanto de la parte de interfaz como la de no-interfaz), que también es alimentada por ambos. Acerca de las necesarias vías de comunicación entre ambos caminos paralelos, afirman que son fundamentales, pero no las describen porque no están todavía bien entendidas. Los autores reconocen que es necesario investigar para comprender mejor y dar soporte a las necesidades reales de comunicación de este complejo proceso.

2.3.1.2 Análisis

Se trata de la aproximación al desarrollo de sistemas interactivos que mejor transmite el enfoque iterativo que debe ser inherente a todo el desarrollo. Esta visión del desarrollo, en el que los tipos de actividades y la secuencia en la que se realizan están completamente separadas, se muestra claramente en la representación del ciclo de vida en estrella de la Figura 2.2.

La aproximación de Hix y Hartson trata muy bien las técnicas de especificación de usabilidad y la evaluación formativa de usabilidad a través de test de usabilidad con usuarios reales. No trata en profundidad, sin embargo, las tareas de análisis de sistemas, las cuales son comunes al

desarrollo de la interacción con el usuario y al desarrollo de la parte interna del sistema (ver Figura 2.3). Por esta razón son fundamentales para la integración de la usabilidad en el proceso de desarrollo. Los autores admiten que las interconexiones entre los dos ámbitos deben ser estudiadas en profundidad para poder definir las con el nivel de detalle requerido, lo cual indica que el tema no se resuelve en esta propuesta.

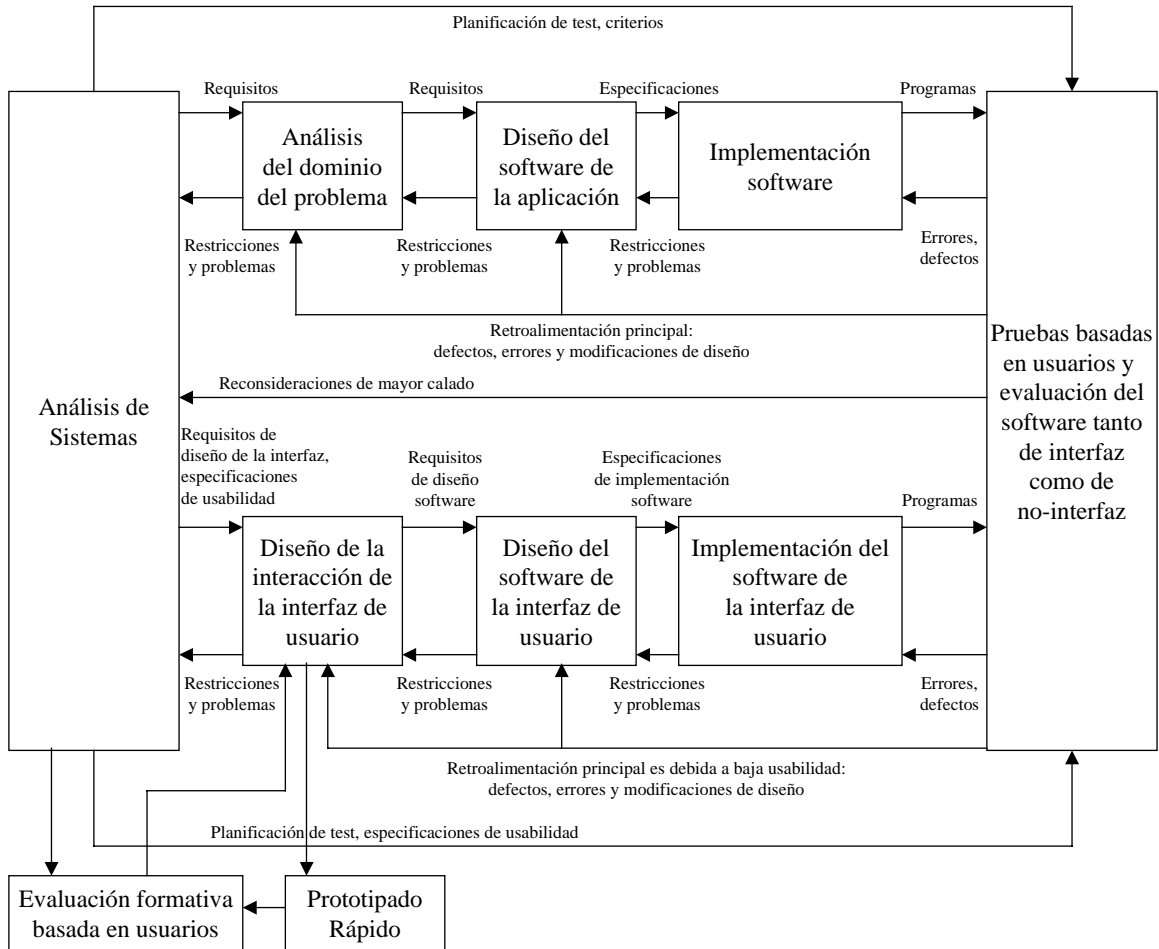


Figura 2.3 - Conexión entre las Actividades de Desarrollo de la IU y el Resto de Actividades

Donde los autores sí se centran, es en la especificación formal de la IU. Defienden que la utilización de una notación adecuada es mejor que la utilización del lenguaje natural para describir los elementos de la IU y su comportamiento. Plantean un escenario de desarrollo en el que el diseño de la interacción con el usuario no tiene por qué realizarla el mismo equipo que va a encargarse de la implementación de la IU, y el uso de especificaciones formales tipo UAN permite que la comunicación entre ambos equipos se pueda realizar sobre una base común sólida. Este no es el caso en muchas organizaciones de desarrollo de software, en las que no hay una distinción neta entre diseñadores e programadores, y en las que no se utilizan las especificaciones formales de la IU por la excesiva carga que suponen en sistemas de complejidad media-alta. Los autores admiten que puede resultar sorprendente al lector el grado de detalle que se alcanza en una descripción UAN, y lo presentan como una alternativa mejor a la de dejar que el programador tome las decisiones sobre los detalles referentes al comportamiento de la IU. Este tipo de modelado formal cuenta con unas importantes limitaciones, como que únicamente estudia la eficiencia del usuario experto, asume que la operación por parte del usuario está libre de errores, y se centra en la ejecución de tareas rutinarias para el usuario [Card, 80].

Al no detallar suficientemente las actividades referentes al análisis de sistemas, el proceso queda mermado en la parte en la que se requiere una mayor integración con el resto de actividades del desarrollo. Por otra parte, en la Figura 2.3 se muestra la evaluación del software, tanto de la parte de la IU como la interna, como común al desarrollo de la interacción con el usuario y al desarrollo del resto del sistema. Ambos tipos de evaluación son diferentes, la parte interna del software se evalúa mediante pruebas de software (*software testing*), que comprende una serie de técnicas independientes de las pruebas con usuarios reales que son necesarios para evaluar la IU.

En cuanto a la integración con las actividades tradicionales, el trabajo de Hix y Hartson admite que no detalla suficientemente los canales de intercomunicación entre desarrollo de la interacción y el resto del desarrollo, y que se trata de un tema abierto de investigación. Por lo tanto, el ciclo de vida en estrella sirve de punto de partida desde el campo de la IPO para plantear la integración de un proceso centrado en el usuario con el resto de actividades del desarrollo, pero no se trata propiamente de una solución para abordar tal integración.

Podemos resumir la propuesta de Hix y Hartson en cuanto a la integración de la usabilidad en el proceso de desarrollo, destacando que sí que adopta un proceso iterativo, pero que no está bien definido como para su aplicación directa por una organización de desarrollo de software, y no especifica cómo llevar a cabo la necesaria integración con las actividades del proceso de desarrollo general. En concreto, en las actividades referentes a los requisitos y el análisis, las cuales son las que necesitan un mayor grado de integración con las actividades tradicionales, realizan una descripción somera de las actividades de usabilidad, que no permite siquiera desarrollar una integración ad-hoc suficiente.

2.3.2 Estándar ISO 13407

2.3.2.1 Descripción

El estándar ISO 13407 (*ISO 13407 Standard. Human-Centred Design Processes for Interactive Systems*) [ISO13407, 99] provee de una guía sobre las actividades de Diseño Centrado en el Humano (DCH²), a través del ciclo de vida de sistemas interactivos basados en ordenador. Se ocupa tanto de componentes software como hardware.

En su cláusula 4 el estándar perfila las razones para adoptar un proceso de DCH. Identifica un proceso de ese estilo con la producción de un sistema más usable, y detalla las ventajas de un producto desarrollado así:

- es más fácil de entender y usar, por tanto reduciendo los costes de formación y soporte,
- mejora la satisfacción del usuario y reduce la incomodidad y el estrés,
- mejora la productividad de los usuarios y la eficiencia operacional de las organizaciones, y
- mejora la calidad del producto, atrae a los usuarios y puede aportar una ventaja competitiva.

² ISO ha optado por emplear la denominación “centrado en el humano” en vez de la tradicional en el campo de la IPO de “centrado en el usuario”.

El estándar se define como no completo, en el sentido de que no cubre todas las actividades necesarias para asegurar un diseño del sistema efectivo. Se plantea como complementario a métodos de diseño existentes y provee una perspectiva centrada en el humano que puede ser integrada en distintas formas de procesos de diseño de una forma adecuada al contexto particular. Caracteriza la incorporación de un enfoque centrado el humano de la siguiente forma:

- la implicación activa de usuarios y una clara comprensión de los requisitos de usuarios y tareas;
- una asignación adecuada de funcionalidad entre usuarios y tecnología;
- la iteración de soluciones de diseño;
- diseño multidisciplinar.

En su cláusula 6 el estándar define las directrices para planificar un proceso de DCH. Dicho plan especifica cómo encajan las actividades centradas en el humano en el proceso de desarrollo general del sistema. Según este estándar, un plan del proceso de DCH debe identificar las actividades del proceso centrado en el humano, los procedimientos para integrarlas con actividades del desarrollo, los responsables de cada actividad y su rango de capacidades, procedimientos efectivos para establecer retroalimentación y comunicación en actividades de DCH cuando afectan a otras actividades, hitos para las actividades de DCH, y escalas de tiempo apropiadas para permitir retroalimentación y cambios de diseño a incorporar en la planificación del proyecto. Este plan del proceso de DCH debe ser parte del plan de proyecto del desarrollo general del sistema.

Las actividades de DCH que deben realizarse durante un proyecto de desarrollo de un sistema se muestran en la Figura 2.4. El proceso debe comenzar en la etapa más temprana del proyecto (por ejemplo, cuando el concepto inicial del producto o sistema se está formulando), y debe ser repetido iterativamente hasta que el sistema cumpla los requisitos.

La necesidad de un enfoque DCH debe identificarse a partir de los objetivos operacionales del sistema, por ejemplo, satisfacer los requisitos de usabilidad del cliente.

Cuando se planifica un proyecto, la descripción de cada actividad y sus sub-tareas debe estudiarse y usarse como guía para diseñar o seleccionar los métodos y técnicas DCH para llevar a cabo la actividad, y documentar progreso y resultados. El esfuerzo dedicado y la importancia relativa de cada actividad dependerá del tamaño y tipo del producto a desarrollar.

A continuación se describen las actividades de diseño centrado en el usuario que detalla el estándar 13407:

- **Comprender y especificar el contexto de uso:** El contexto de uso del que se ocupa esta actividad debe identificarse según lo siguiente:
 - Las características de los usuarios previstos.
 - Las tareas que los usuarios realizarán, incluyendo los objetivos generales de uso del sistema. El estándar menciona que las tareas no deben ser descritas únicamente en base a las funciones o funcionalidades provistas por un producto o sistema.
 - El entorno en el que los usuarios usarán el sistema: hardware, software y materiales que se van a utilizar.

También debe incluir las características relevantes del entorno físico y social. La salida de esta actividad debe ser una descripción de las características relevantes de los usuarios, tareas y entorno que identifican los aspectos que tienen un impacto importante en el diseño del sistema.

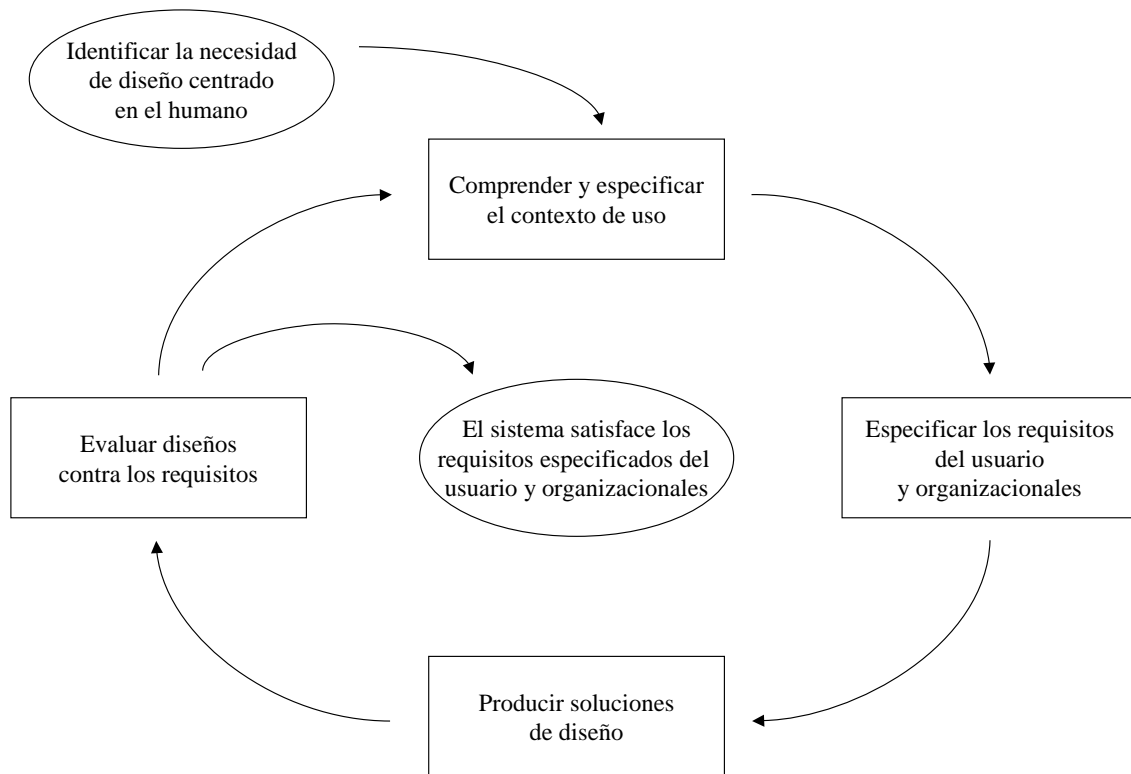


Figura 2.4 - Interdependencia entre las Actividades de Diseño Centrado en el Humano

- **Especificar los requisitos organizacionales y del usuario:** Extiende la actividad tradicional de requisitos para crear una declaración explícita de los requisitos de usuario y organizacionales en relación a la descripción del contexto de uso. Para definir estos requisitos hay que considerar los siguientes aspectos:
 - Rendimiento requerido para el nuevo sistema según los objetivos financieros y operacionales.
 - Requisitos estatutarios o legales de relevancia.
 - Cooperación y comunicación entre usuarios y otras partes relevantes.
 - El trabajo de los usuarios (incluyendo la asignación de tareas, el bienestar de los usuarios y su motivación).
 - Rendimiento de tareas.
 - Diseño del trabajo y organización.
 - Gestión del cambio, incluyendo la formación y el personal involucrado.
 - Viabilidad de la operación y mantenimiento.
 - La interfaz humano-ordenador y el diseño de las estaciones de trabajo.

La especificación debería definir también la asignación de funciones: La división de las tareas del sistemas entre aquellas realizadas por humanos y aquellas realizadas por la tecnología.

- **Producir soluciones de diseño:** El proceso de producir soluciones de diseño implica las siguientes actividades:

- Usar el conocimiento existente para desarrollar propuestas de diseño con entrada multidisciplinar.
 - Hacer más concretas las soluciones de diseño utilizando simulaciones, modelos, maquetas (*mock-ups*), etc. Es decir, se refiere al uso de cualquier tipo de prototipo.
 - Presentar las soluciones de diseño a los usuarios y permitirles realizar tareas (o tareas simuladas).
 - Alterar el diseño en respuesta a la retroalimentación de los usuarios e iterar este proceso hasta que los objetivos del DCH se cumplan.
 - Gestionar la iteración de soluciones de diseño. Consiste en registrar los resultados de las 4 actividades anteriores en una documentación que incluiría las fuentes de conocimiento existente o estándares utilizados, los pasos tomados para asegurar que el prototipo cubre requisitos clave y sigue buenas prácticas, y la naturaleza de los problemas encontrados y los subsiguientes cambios realizados al diseño.
- **Evaluar diseños contra requisitos:** La evaluación es un paso esencial en el DCH, y debe realizarse en todas las etapas del ciclo de vida del sistema. Se puede utilizar la evaluación para obtener retroalimentación, la cual puede usarse para mejorar el diseño, para evaluar si los objetivos del usuario y organizacionales se han alcanzado, y para observar el uso a largo plazo del producto o sistema. Cuanto más cerca se está del comienzo del desarrollo, más se centran las actividades de evaluación en el objetivo de retroalimentación que guíe el diseño, mientras que más adelante en el desarrollo, cuando ya se tiene un prototipo completo, se pueden realizar evaluaciones contra los objetivos de usuario y organizacionales.

Se debe confeccionar un plan de evaluación que identifique:

- Los objetivos de DCH.
- Quién es responsable de la evaluación.
- Qué partes del sistema se van a evaluar y cómo.
- De qué forma se van a realizar las evaluaciones y los procedimientos para llevar a cabo las pruebas.
- Los recursos necesarios para la evaluación, análisis de los resultados y acceso a los usuarios.
- La planificación de actividades de evaluación en relación a la planificación del proyecto.
- La retroalimentación y utilización de los resultados en otras actividades de diseño.

La evaluación puede utilizarse para:

- Demostrar que un diseño particular cumple los requisitos centrados en el humano.
- Evaluar conformidad a estándares estatutarios, corporativos, locales, nacionales o internacionales.

Los resultados de evaluación deben ser registrados de forma sistemática. El Anexo B del estándar incluye un ejemplo de la estructura de un informe utilizado para ofrecer retroalimentación al diseño basado en pruebas con usuarios.

El estándar detalla los temas relativos a cada tipo de informe de evaluación: Informe de retroalimentación al diseño, informe de pruebas del diseño contra estándares específicos, e informe de pruebas con usuarios.

2.3.2.2 Análisis

El estándar es ampliamente citado y se ha convertido en una referencia básica sobre procesos de desarrollo centrados en el usuario. Las actividades del proceso de desarrollo descrito son suficientemente generales como para acomodar los principales métodos de desarrollo centrado en el usuario: El ciclo de vida de la ingeniería de usabilidad [Mayhew, 99], el ciclo de vida en estrella [Hix, 93], y el diseño centrado en el uso [Constantine, 99]. Otras propuestas menos formalizadas como la de Nielsen [Nielsen, 93] también admiten acomodo en el marco que proporciona este estándar, puesto que siguen su espíritu básico.

Es centrado en el usuario, pues se trata de una referencia básica en la IPO de lo que supone dicho enfoque de desarrollo. En cuanto a la iteratividad, la Figura 2.4 refleja un enfoque verdaderamente iterativo.

El problema principal de este estándar es que no es suficientemente detallado como para ser directamente aplicado por una organización de desarrollo de software, pues no especifica técnicas para todas las actividades. Nombra el prototipado en la producción de soluciones de diseño, y nombra también técnicas de evaluación de usabilidad, pero no menciona ninguna técnica para la comprensión y especificación del contexto de uso, ni para la especificación de requisitos de usuario y organizacionales. En estas dos últimas actividades únicamente se describe qué debe incluir la especificación del contexto de uso y la especificación de requisitos del usuario y organizacionales, respectivamente. En la cláusula inicial de ámbito ya se indica que el estándar no ofrece cobertura detallada de los métodos y técnicas requeridos para el DCH.

En cuanto a la integración de las actividades de usabilidad con el resto de actividades del desarrollo, el estándar indica que hay que definir para cada proyecto cómo se integran ambos tipos de actividades. Sin embargo, únicamente nombra una actividad tradicional, la de especificación de requisitos, que se dice que se extiende con los requisitos de usuario y organizacionales. No hay más referencias a actividades tradicionales de la IS. Por tanto, se presenta como descriptor de las actividades complementarias a procesos de desarrollo existentes, pero deja la integración de ambos tipos de actividades como tarea a resolver casi completamente por la organización de desarrollo de software que desee aplicar prácticas de usabilidad en su proceso.

2.3.3 Ciclo de Vida de la Ingeniería de Usabilidad

2.3.3.1 Descripción

Mayhew [Mayhew, 99] propone el Ciclo de Vida de la Ingeniería de Usabilidad para el desarrollo de IUs usables. El enfoque de este método es el de rediseñar el proceso de desarrollo completo en torno de la experiencia, métodos y técnicas de la ingeniería de usabilidad: “Presento las tareas de ingeniería de usabilidad en el primer plano, con menos énfasis en las tareas globales de ingeniería del software, para comunicar mi creencia de que los requisitos de usuario y el diseño de la IU deberían dirigir el proceso de desarrollo general, en vez de ser dirigidas por éste o ser incidentales al mismo” [Mayhew, 99]. Para tal fin, incluye para cada tarea apuntes de cómo integrarla con las actividades del método OOSE (*Object Oriented Software Engineering*) [Jacobson, 93]. En concreto, para cada actividad detalla si debe hacerse antes, en paralelo, o después de una determinada actividad de OOSE, y referencias generales a cómo cuadran los productos con los de OOSE. También apunta conexiones a nivel general con enfoques de diseño basados en el prototipado rápido.

La autora resume la filosofía del ciclo de vida de la ingeniería de usabilidad en los siguientes puntos:

- El diseño de la IU es clave
- La integración de la ingeniería de usabilidad con la IS debe ser particularizada
- El análisis de requisitos vale la pena
- El diseño puede aproximarse en un proceso estructurado de descomposición (*top-down*)
- El diseño, las pruebas y el desarrollo deberían ser iterativos
- El ciclo de vida completo puede ser estratificado en subconjuntos de funcionalidad
- Hay una variedad de técnicas para llevar a cabo cada tarea del ciclo de vida
- Las técnicas alternativas hacen que el ciclo de vida sea flexible y adaptable
- Una implementación óptima del ciclo de vida requiere la participación completa de equipos multidisciplinares.

El ciclo de vida propuesto estructura las actividades en tres fases: Análisis de Requisitos, Diseño/Pruebas/Desarrollo, e Instalación, según se muestra en la Figura 2.5.

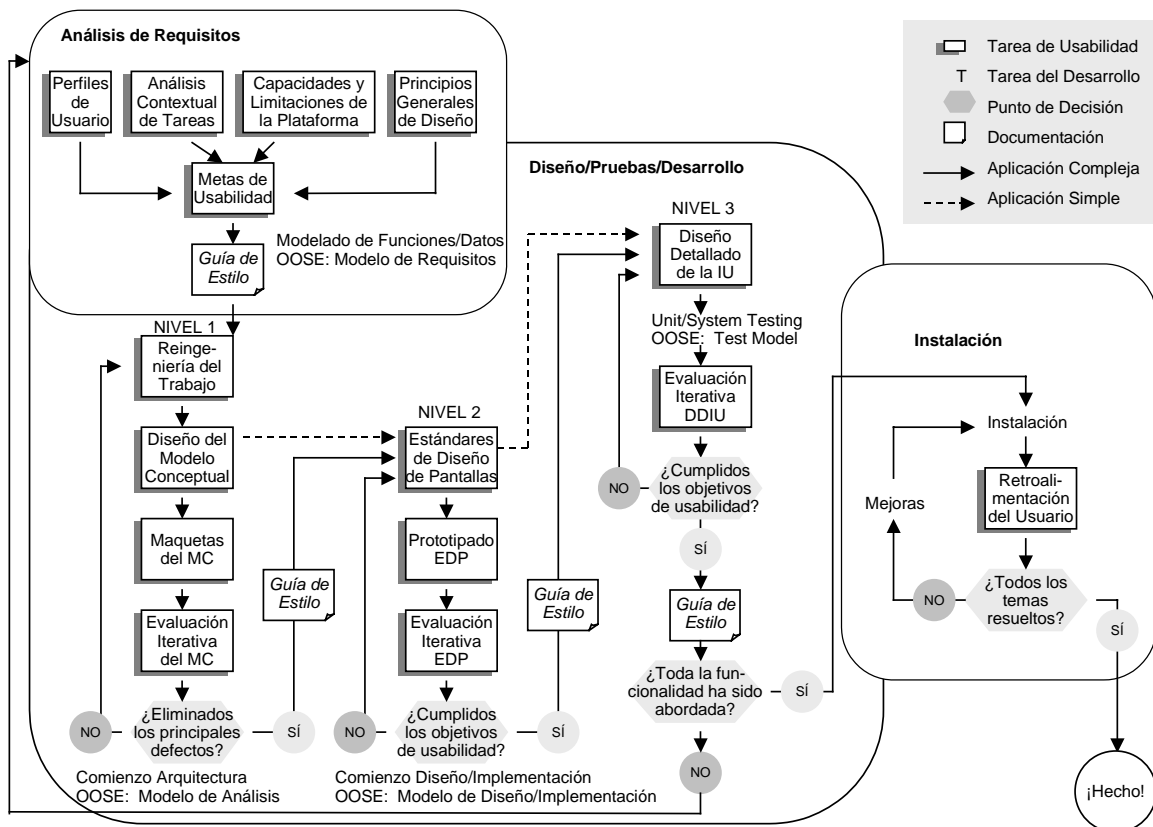


Figura 2.5 - Ciclo de Vida de Ingeniería de Usabilidad de Mayhew

Dedica un capítulo a cada tarea del ciclo de vida, detallando para cada una: su propósito; una descripción general; un apunte de su integración con otras tareas de usabilidad y con las tareas

del proceso de desarrollo general; roles y recursos necesarios; y una técnica base que explica paso a paso, junto con ejemplos de productos y plantillas. Da indicaciones también para cada tarea del nivel de esfuerzo que requiere su aplicación. Finalmente, en algunas actividades incluye referencias a técnicas alternativas a la detallada como técnica base, atajos para acortar la duración de la tarea (similar al concepto de *discount usability engineering* de Nielsen [Nielsen, 93]), e indicaciones sobre cómo realizar la tarea en el campo específico del desarrollo web.

Según la autora, las tareas del ciclo de vida de la ingeniería de usabilidad (y en particular la tarea del Análisis Contextual de Tareas) se centran en proyectos en los cuales un producto específico ha sido identificado, definido y delimitado. Nombra otras técnicas de análisis de tareas que se centran en labores de reingeniería de proceso que incluyen la identificación de oportunidades para nuevos productos, o en la identificación de las características básicas que deberían incorporarse en nuevos productos. La tarea de Análisis Contextual de Tareas es presentada como la más adecuada cuando se tiene ya un conjunto de funcionalidades identificadas y delimitadas para su automatización. La autora da mucha importancia al Análisis Contextual de Tareas, pues de las 15 tareas de las que cuenta su ciclo de vida, a la del Análisis de Tareas es a la que dedica el capítulo más extenso, entrando en amplio detalle en la técnica base descrita para esta tarea.

Las actividades de diseño/pruebas/implementación propuestas están centradas en el diseño y evaluación de la IU. Divide tal diseño en tres niveles: modelo conceptual, estándares de diseño de las pantallas, y diseño detallado de la IU. Cada nivel es diseñado, a continuación se construye un prototipo que ilustre el diseño, y se evalúa su usabilidad antes de proceder al siguiente nivel de diseño de la IU. En el nivel más abstracto se encuentra el modelo conceptual, que consiste en la definición de la orientación básica de la IU (a proceso o a producto), las ventanas (*displays*) principales y la navegación entre los mismos, y las reglas de presentación a alto nivel para cada producto o proceso y para las ventanas. En el siguiente nivel está los estándares de diseño de las pantallas, que aseguran la consistencia y simplicidad en el diseño detallado a lo largo de todos las ventanas de la interfaz de un producto, en cuanto a uso de controles, localización y formato de los elementos estándar de la interfaz, terminología, uso de las fuentes y tipo de letra, etc. En el nivel más detallado se encuentra el diseño detallado de la IU, en el que se documenta el diseño de todos los caminos, ventanas e interacciones, conforme a las reglas establecidas en los dos niveles superiores. El conjunto de las decisiones de diseño de los dos niveles superiores se refleja en la guía de estilo del producto.

Las actividades relacionadas con el diseño iterativo del modelo conceptual se presentan como tareas que deberían, por un lado, seguir al desarrollo del Modelo de Requisitos de la fase de Análisis de OOSE, y por otro, preceder o llevarse a cabo en paralelo con el desarrollo del Modelo de Análisis de la misma fase. La razón para esta precedencia, según la autora, es que la arquitectura de la aplicación debería diseñarse para dar soporte al diseño del modelo conceptual de la IU.

Las actividades de diseño de la IU al nivel de estándares de pantalla se desarrollan en paralelo o van después del Modelo de Análisis de la fase de Análisis de OOSE, puesto que, según la autora, lo más probable es que no afecte al diseño de la arquitectura. Se llevan a cabo en paralelo con el Modelo de Diseño de la fase de Construcción de OOSE, porque tampoco se considera que afecten al diseño de alto nivel. En cambio, sí preceden al Modelo de Implementación de la fase de Construcción de OOSE.

2.3.3.2 Análisis

Es un método que está explicado de forma detallada y que abarca todas las actividades principales en un proceso de desarrollo de usabilidad. Se trata de la propuesta de proceso del campo de la IPO mejor definida como proceso, la más guiada y completa en cuanto a descripción de actividades y técnicas a llevar a cabo y productos de las mismas.

La autora admite que el ciclo de vida que propone debe ser particularizado para cada proyecto que se lleve a cabo en cuanto a su integración con el proceso de la IS. De hecho, los apuntes sobre integración de cada tarea con las actividades de OOSE o de un método de prototipado rápido no son detalladas, están descritas a un nivel muy general. Para cada actividad únicamente se refiere a los modelos de OOSE a los que precede, sigue, o con los que se desarrolla en paralelo. La integración se aborda desde la perspectiva de que el desarrollo de la IU es una actividad que es bastante independiente del desarrollo del resto del sistema. El método se plantea como de diseño de la IU, sin embargo incluye actividades típicamente de análisis como es el Análisis Contextual de Tareas, con actividades de observación de los usuarios en su entorno de trabajo. La autora no aborda esta contradicción, la cual está en la base de la visión del desarrollo en la que éste gira en torno al ciclo de vida de la ingeniería de usabilidad. La parte de requisitos o análisis no se integra en mayor medida con el proceso de IS, cuando se trata de una parte fundamental para el desarrollo de productos usables (como admite implícitamente la autora al dedicar varias actividades a tareas de análisis).

El enfoque del método de rediseño del desarrollo en torno a la usabilidad es su principal inconveniente. Plantear que el desarrollo esté dirigido por la usabilidad requiere un cambio demasiado radical para un gran número de organizaciones de desarrollo de software. La autora, por una parte, sostiene que el desarrollo debe basarse en los principios de la ingeniería de usabilidad, y, por otra, afirma que su propuesta encaja con el proceso de desarrollo que se lleve a cabo, lo cual es contradictorio. Además, la autora demuestra un cierto desconocimiento de la IS al afirmar que “[La IS] se trata de un enfoque al desarrollo de software que implica definir requisitos de la aplicación, establecer objetivos, y diseñar y testear en ciclos iterativos hasta que los objetivos se cumplen” [Mayhew, 99]. Aunque el enfoque iterativo es la tendencia que actualmente recibe una mayor atención en la IS, no se trata de una definición válida de la disciplina. Tales errores de concepto respecto a la IS pueden resultar desconcertantes para un ingeniero software, haciendo que aborde con escepticismo la propuesta en su conjunto.

La principal carencia de esta propuesta desde un punto de vista centrado en el usuario, es que el proceso no es verdaderamente iterativo. La autora, al explicar los puntos básicos de la filosofía del proceso, indica que diseño, pruebas y desarrollo deberían ser iterativos, que los distintos niveles de abstracción del diseño se mezclan en una fase, y que están separados en conjuntos de funcionalidad. Sin embargo no nombra el análisis. Aunque sí nombra al proceso de diseño como iterativo, entre análisis y diseño no parece haber tal iteración. La vuelta atrás desde la implementación al análisis está etiquetada como “¿Toda la funcionalidad ha sido abordada?”. En la descripción que da la autora de este punto de decisión, se entiende que la vuelta atrás es únicamente para añadir funcionalidad. Por tanto el enfoque es incremental, pero no iterativo. De hecho indica explícitamente que “diseño, pruebas, y desarrollo deberían ser iterativos” [Mayhew, 99], dejando fuera el análisis. Para la autora el análisis no entra dentro de las iteraciones, a diferencia del modelo de proceso centrado en el usuario descrito en el estándar ISO 13407 [ISO13407, 99], en donde la iteración comprende también la actividad de “comprender y especificar el contexto de uso”. Por otra parte, en la representación del ciclo de vida de la Figura 2.5, dentro de la etapa de Diseño/Pruebas/Desarrollo hay tres niveles estancos correspondientes a los tres niveles de diseño de la IU, sugiriendo esta representación que no se pasa al siguiente nivel hasta haber definido completamente el anterior. Este tipo de aproximación indica, una vez más, una concepción del desarrollo basada en el ciclo de vida en cascada.

La autora cree que su propuesta es válida para todo tipo de proyectos, pero da una gran importancia al Análisis Contextual de Tareas. Como se ha indicado en la descripción de la aproximación, esta técnica es presentada como apropiada para la automatización de funcionalidades ya delimitadas, evitando así las complicaciones de la definición de los requisitos del sistema. Deja en gran medida de lado las actividades destinadas a desarrollar el concepto del producto en sentido amplio, las que ayudan a delimitar qué tipo de sistema se va a construir, que sí están presentes en otros autores de la IPO [Hix, 93] [Preece, 94] [Shneiderman, 98] [Constantine, 99]. Este enfoque justificaría parcialmente la adopción de un ciclo de vida en cascada, pues el riesgo de la inestabilidad de requisitos es mucho menor en un sistema cuya funcionalidad está clara y bien delimitada. Para proyectos en los que se requiera un alto grado de innovación o con requisitos no delimitados completamente, esta propuesta no sería adecuada, limitando así su aplicabilidad general. Sí que aborda, por el contrario, el tema del concepto del producto en un sentido reducido, circunscrito únicamente a la organización de la IU. Se aborda en el primer nivel de diseño de la IU en su ciclo de vida, el Diseño del Modelo Conceptual, en el cual se define un esquema (*framework*) basado en reglas de presentación. Según la autora, este esquema proporcionará una base unificadora para todas las decisiones de diseño de la interfaz detallada que vendrán en momentos posteriores del ciclo de vida. El papel que da al modelo conceptual del sistema es menor, al reducirlo únicamente a una decisión de cómo va a ser la IU. La autora indica que mantener separados el Modelo de Organización de Tareas Remodelado (*Reengineered Task Organization Model*) y el Modelo Conceptual, facilita en gran medida el proceso de diseño al disociar los temas de organización de los de presentación. Este enfoque difiere, por ejemplo, del de la metodología LUCID, descrita por Shneiderman [Shneiderman, 98], que define el concepto del producto antes de entrar a diseñar las tareas, puesto que el concepto del producto es la base de la metodología. En la descripción que dan Preece et al. [Preece, 94] de los modelos conceptuales, también se refieren al sistema (como opuesto a únicamente la IU). En concreto, incluye el comportamiento del sistema, la manera en la que se utiliza el sistema, que se recoge en el modelo de tareas. Por tanto, parece que separar el modelo conceptual del modelo de tareas como sugiere Mayhew implica cambiar el significado del término, tal y como éste es empleado por otros autores del campo de la IPO.

Podemos concluir que, a pesar de que la propuesta describe el proceso de forma detallada, no es aplicable directamente de forma general por referir la relación únicamente con la metodología OOSE, por estar centrada en proyectos cuyo objetivo es la automatización de tareas y, principalmente, por no tener un carácter verdaderamente iterativo, el cual es un componente fundamental de cualquier desarrollo centrado en el usuario.

2.3.4 Diseño Centrado en el Uso

2.3.4.1 Descripción

Constantine y Lockwood [Constantine, 99] denominan a su aproximación diseño centrado en el uso para diferenciarlo del diseño centrado en el usuario, al cual identifican con un enfoque con tintes mucho más participativos en los que el usuario toma decisiones. Los autores presentan su obra como dedicada a desarrolladores medios, no a especialistas en usabilidad y, según sus palabras, se trata de un método racionalizado y sistemático para la construcción de software que se ajusta a las necesidades reales de los usuarios. Según los autores, su método se basa en una reducción a los modelos que más pueden contribuir en la tarea de desarrollar software usable. Cada modelo se alimenta de modelos previos para abordar algún punto concreto del desarrollo de software.

A pesar de que Constantine es una figura reconocida en el campo de la IS, y de que el Diseño Centrado en el Uso no es considerado como parte de la IPO por algunos especialistas de dicho

campo, hemos considerado que se trata de una aproximación proveniente del campo de la IPO por las siguientes razones:

- Según sus autores, el Diseño Centrado en el Uso se centra en el desarrollo de sistemas interactivos para el uso humano. Esta definición encaja con la de la disciplina de la IPO que se ofrece en el *ACM SIGCHI Curricula for Human Computer Interaction*: "La IPO es una disciplina que se ocupa del diseño, evaluación e implementación de sistemas informáticos interactivos para uso humano y del estudio de los principales fenómenos que les rodean" [ACM-SIGCHI, 92].
- El Diseño Centrado en el Uso no se presenta como una extensión a ningún método concreto de desarrollo de la IS, más allá de relacionarlo con los enfoques orientados a objetos en general. En particular, Constantine y Lockwood no nombran ninguna relación con el Diseño Estructurado, método por el cual Constantine adquirió su fama en IS.
- El propio Constantine se etiquetó a sí mismo como perteneciente al campo de la IPO en el transcurso del *workshop "Bridging the Gaps Between Software Engineering and Human-Computer Interaction"* en la conferencia ICSE'2003, al cual asistió también el autor del presente trabajo de investigación. Dicha adscripción se produjo cuando los organizadores del *workshop* pidieron que los participantes definieran alzando la mano a qué campo pertenecían: IS o IPO.

Contra la corriente dominante en el campo de la IPO, Constantine y Lockwood abogan por un cuidadoso diseño como herramienta fundamental. La idea central es intentar hacer un buen diseño en primer lugar, aunque siempre se utilicen test de usabilidad para iterar hacia la solución que mejor satisface las necesidades del usuario. Plantean su método como una aproximación sistemática al diseño de software usable, frente a la de *usability testing*, que identifican como una versión elaborada de un desarrollo exploratorio (ensayo y error). Según los autores, las pruebas de usabilidad ofrecen un mayor valor si se realizan sobre un diseño bien concebido, con una lógica propia en su diseño, que si se realizan sobre diseños creados de cualquier manera.

El método de diseño centrado en el uso se presenta según cinco elementos clave:

- Guías pragmáticas de diseño: Heurísticas de diseño de la interfaz.
- Un proceso de diseño dirigido por modelos. Estos modelos están interrelacionados y proporcionan especificaciones a distintos niveles de abstracción. Los tres principales modelos son los siguientes:
 - Modelo de Roles de Usuario: Modela los tipos de usuarios y las relaciones entre los mismos. Consta de los roles de usuario y el mapa de roles de usuario.
 - Modelo de Tareas: Está basado en los casos de uso y el mapa de casos de uso.
 - Modelo de contenido: Modela los elementos y mecanismos que va a proporcionar la IU. Incluye un mapa de navegación entre los distintos espacios de interacción de la IU.

Otros dos modelos representan información adicional: El modelo operacional, que representa el contexto en el que el sistema va a ser operado, y el modelo de implementación, que detalla el diseño visual de la IU y su modo de operación.

- Una serie de actividades de desarrollo organizadas. Esto es, un proceso de desarrollo, aunque los autores evitan llamarlo metodología por las connotaciones de rigidez que conlleva dicho término. De todas formas, a pesar de esta afirmación de los autores, la obra no trata de los aspectos de proceso más que en el capítulo de introducción al enfoque de diseño centrado en el uso, pues en el resto de la obra únicamente se describen las técnicas individuales, sin una definición del proceso detallada. En cuanto a la secuencia entre actividades, aunque los autores colocan las actividades de su proceso en una especie de cascada, defienden una aproximación de ingeniería concurrente, en la que hay actividades que se realizan en paralelo, y con cierto solapamiento entre actividades, según se muestra en la Figura 2.6. En dicha figura se puede observar que los autores mezclan actividades en sentido estricto, como el modelado de tareas, con conceptos de desarrollo como la construcción concéntrica o la iteración arquitectural. Las actividades de la figura con fondo gris pertenecen, según los autores, al proceso de diseño desarrollo software más amplio.

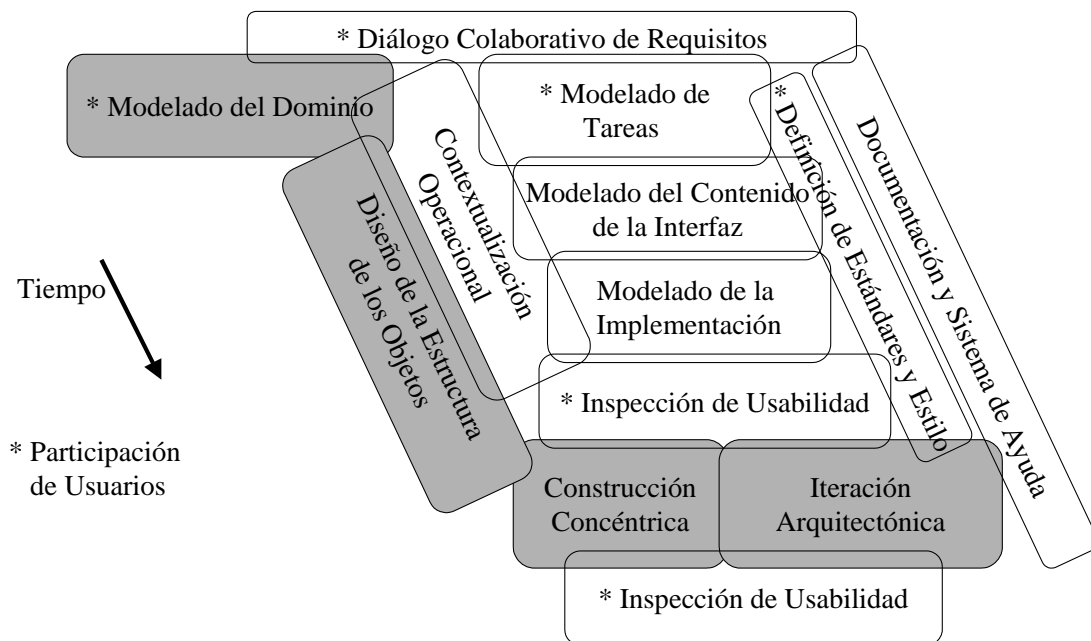


Figura 2.6 Modelo de Actividades del Diseño Centrado en el Uso

- Mejora iterativa.
- Medidas de calidad. En cuanto a evaluación, los autores abogan por el uso de Inspecciones de Usabilidad Colaborativas, métricas de diseño de la IU y pruebas de usabilidad. También nombran otras aproximaciones a la evaluación de la usabilidad, como Evaluaciones Heurísticas o Recorridos (*Walkthroughs*) Cognitivos.

Los usuarios participan de algunas de las actividades del proceso: Diálogo Colaborativo de Requisitos, Modelado del Dominio, Modelado de Tareas, Definición de Estándares y Guías de Estilo, e Inspecciones de Usabilidad. Para la primera de ellas, proponen los autores la técnica JEM (*Joint Essential Modeling*), una versión mejorada de la técnica JAD.

Los autores identifican la esencia del método, el mínimo irreducible del enfoque centrado en el uso, en el modelado de tareas con casos de uso esenciales. Estos casos de uso son una reinterpretación de los casos de uso utilizados en los métodos de desarrollo orientado a objetos. El enfoque que se propone a la hora de construirlos consiste en evitar toda referencia a elementos concretos de interacción. Se detallan únicamente la intención del usuario y las responsabilidades del sistema.

El modelo de actividades de la Figura 2.6 es presentado por los autores como una posible integración de las actividades de usabilidad en el proceso de desarrollo general, pero no la única. Los autores afirman que los enfoques efectivos tienen en común una colección flexible y organizada de actividades con una implicación de usuarios dirigida a objetivos en puntos concretos del proceso. Los autores realizan algunas consideraciones acerca de cómo puede abordarse la integración de la usabilidad en el desarrollo de software, sugieren estudiar el proceso de desarrollo actual para ver dónde participan los usuarios y cómo los distintos procesos contribuyen a la usabilidad. Aconsejan comenzar por la introducción de la usabilidad en las etapas iniciales del ciclo de vida, e ir a continuación gradualmente introduciendo nuevos elementos en etapas cada vez más avanzadas, hasta cubrir el ciclo de vida completo. Los autores identifican las siguientes características comunes en esfuerzos de integración que han resultado exitosos:

- La usabilidad necesita convertirse en un tema importante y visible
- Formación adecuada para el personal de desarrollo
- Inspecciones
- Integración de las nuevas prácticas con las antiguas. Los autores sostienen que una buena estrategia de integración de usabilidad en el ciclo de vida acomoda las nuevas prácticas con las antiguas, modificando las prácticas para incorporar la usabilidad en los procesos de análisis y diseño, al tiempo que se particulariza el diseño centrado en el uso a la organización y sus prácticas. De todas formas, dejan el tema abierto, pues no dan indicaciones de cómo conseguir este objetivo.
- Guías de diseño internas
- Introducción de test de usabilidad dirigidos a temas específicos, cuando se llega a un punto en el que la sofisticación de los diseñadores y la sutileza de los aspectos de usabilidad tratados permiten garantizar tal introducción.
- Medidas para refinar el proceso

Aunque los autores afirman que el conjunto de técnicas propuesto funciona mejor cuando se aplica de forma conjunta, en el marco del proceso centrado en el uso, indican que muchas técnicas también pueden ser útiles aplicadas de forma aislada. Las prácticas que indican como mejor dotadas para ser aplicadas individualmente son: Principios de diseño de usabilidad, modelado de casos de uso esenciales, prototipos abstractos con modelos de contenido, mapas de navegación para el modelado de transiciones entre contextos, contextualización operacional, inspecciones colaborativas de usabilidad, construcción concéntrica priorizada, refinamiento arquitectónico iterativo, y métricas de calidad del diseño de la IU.

2.3.4.2 Análisis

Es el enfoque perteneciente al campo de la IPO que más se aproxima al modo de hacer las cosas de la IS. En particular, los casos de uso esenciales (la pieza clave del método según sus autores), pueden ser un puente entre ambas disciplinas. Este nuevo enfoque de una técnica ampliamente usada en el campo de la IS, puede permitir la integración del enfoque centrado en el uso en el desarrollo de software. Constantine y Lockwood critican el uso de la técnica de casos de uso original para el diseño de IUs, porque tales casos de uso típicamente contienen muchas suposiciones implícitas o escondidas sobre la forma de la IU que se va a diseñar. Están demasiado cercanos a la implementación y no se ciñen suficientemente a los problemas a los que se enfrentan los usuarios. La viabilidad del uso de los casos de uso esenciales en

procesos de desarrollo de la IS está avalada por la mención que de esta técnica se hace en dos populares obras de referencia del campo de la IS: Una centrada en los casos de uso ([Cockburn, 01]) y otra en el desarrollo orientado a objetos ([Larman, 02]).

El método está destinado a todo tipo de desarrolladores, pues los autores constatan que, aunque tal vez sería deseable que las decisiones que afectan a la usabilidad del producto fueran tomadas únicamente por expertos en usabilidad, en la práctica los desarrolladores toman una gran cantidad de decisiones en sus labores de desarrollo, las cuales dan forma y determinan la usabilidad última del producto software. Así, el método propuesto no está centrado únicamente en el diseño de la IU, sino que aborda el diseño del sistema completo, al menos en la elaboración de los casos de uso esenciales, que resultarían una labor fundamental de las actividades relacionadas con los requisitos.

Algunas técnicas descritas son cercanas al enfoque de modelado de la IS, con un énfasis en la especificación de los detalles importantes para que puedan tenerse en cuenta en etapas posteriores del desarrollo, frente a otros objetivos típicos de otros enfoques del campo de la IPO en los que se busca la obtención de una comprensión más profunda del modo de pensar de los usuarios y su visión de las tareas que desempeñan.

Aunque los autores incluyen unas consideraciones sobre cómo introducir la usabilidad en el proceso software, no van más allá de dar recomendaciones generales. El método es muy útil usado por sí mismo, pero de difícil integración, en su forma presente, en un proceso de desarrollo de software de la IS. Esto es debido a que el proceso integrado propuesto gira completamente en torno a la usabilidad, exigiendo a la organización de desarrollo de software que quiera implementarlo abandonar su proceso actual y adoptar el proceso propuesto. La única conexión con tareas del desarrollo tradicional es a través del modelado del dominio y de la estructura de objetos, pero no se da ninguna indicación de cómo debería abordarse tal conexión. En cuanto al diseño de objetos, únicamente se dan directrices de cómo realizar un diseño orientado a objetos de la parte de la aplicación correspondiente a la IU, la cual es tan solo una parte del diseño de objetos global.

Para la integración con el proceso de desarrollo, los autores indican que se tiene que partir del proceso de desarrollo actual y empezar a estudiarlo para ver dónde se construye la IU, e identificar puntos flacos o fallos en las prácticas actuales. Esto lleva a una adaptación caso por caso. Como se ha mencionado en la descripción de la aproximación, el modelo de actividades de la Figura 2.6 lo presentan los autores como una posible aproximación entre otras al tema de la integración. En esta sección nos vamos a basar en dicha aproximación en concreto (la del modelo de actividades reflejado en la Figura 2.6).

El principal inconveniente para adoptar el proceso propuesto estriba en que dicho proceso no está suficientemente definido, puesto que la asignación de técnicas a actividades no es explícita. De esta forma, una organización que se plantee la introducción de las técnicas propuestas en su proceso de desarrollo tiene que partir de cero en la tarea de enlazar las actividades en las que se aplican las técnicas de usabilidad con el resto de actividades del desarrollo, y los modelos propuestos por Constantine y Lockwood con los productos de las actividades preexistentes en el proceso.

Debido a la indefinición del proceso de desarrollo, no se puede identificar el verdadero grado de iteratividad que subyace en la propuesta más que por los comentarios de los autores acerca de la iteración arquitectónica. Así, aunque la Figura 2.6 podría sugerir un ciclo de vida en cascada con cierto grado de solapamiento, cuando los autores detallan su enfoque de iteración arquitectónica, sí que abogan por un enfoque iterativo donde el problema a solucionar es la primera versión del producto software. Así, evitan el peligro de la parálisis del análisis que conlleva un ciclo de vida en cascada. Debido a estas puntualizaciones sobre enfoque de desarrollo, consideramos que la propuesta es verdaderamente iterativa.

2.3.5 Modelo ISO/TR 18529

2.3.5.1 Descripción

El informe técnico ISO 18529 (*Technical Report ISO/TR 18529. Human-Centred Lifecycle Process Descriptions*) [ISO18529, 00] presenta un modelo de proceso de ciclo de vida centrado en el humano, el cual, según sus autores, extiende y formaliza el presentado en el estándar ISO 13407 [ISO13407, 99]. Este modelo de proceso define qué procesos deberían ser llevados a cabo por una empresa para alcanzar metas técnicas definidas. Los procesos están descritos en el formato definido en ISO/IEC 15504 [ISO/IEC15504, 98] para evaluación de procesos software, el cual proviene del campo de la IS.

El modelo de proceso propuesto consta de 7 procesos de DCH. Dichos procesos y la relación entre los mismos se refleja en la Figura 2.7. La figura intenta reflejar la naturaleza cíclica o iterativa de los procesos de DCH y la relación entre los mismos.

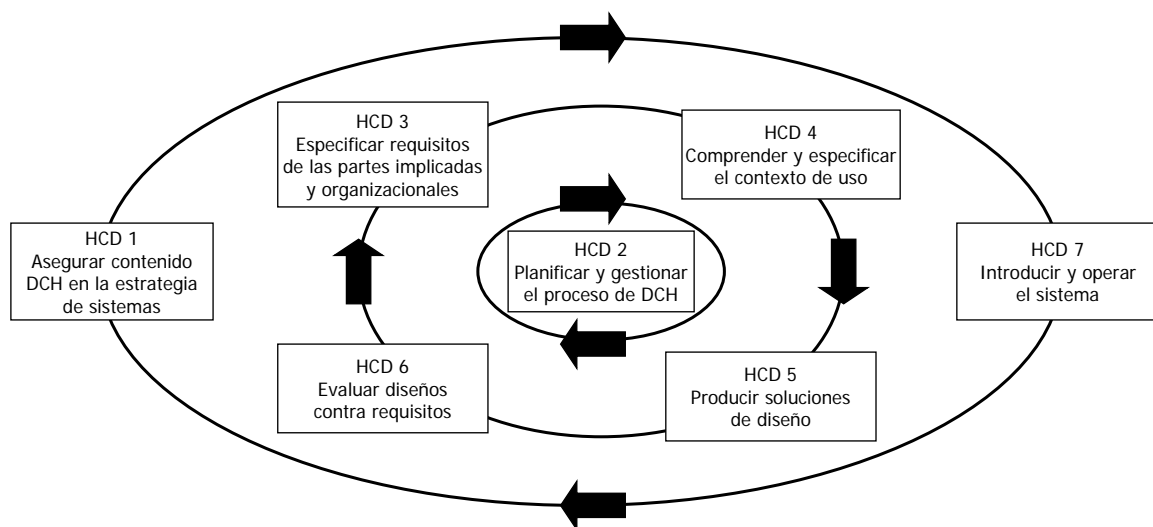


Figura 2.7- Relación entre los Procesos Centrados en el Usuario en el Ciclo de Vida

En cuanto a los procesos que forman parte del modelo, se especifica concretamente que no deberían confundirse con las etapas de un ciclo de vida. Los procesos se llevan a cabo en más de una etapa del ciclo de vida.

El subproceso HCD.2.5 "Planificar las actividades de DCH" es especialmente relevante para el objetivo de integración en el proceso software, puesto que se ocupa del desarrollo de un plan que especifique cómo las actividades centradas en el humano se integran en el proceso de desarrollo del sistema general. En concreto, el modelo indica que el plan especifica cómo se usan para el diseño entradas desde actividades de DCH.

El modelo incluye las prácticas o sub-procesos que forman los procesos mostrados en la Figura 2.7. Sin embargo, el modelo no incluye métodos de DCH, técnicas y herramientas particulares. Cada proceso utiliza y produce productos de trabajo asociados, los cuales se detallan en el Anexo A del documento.

En el Anexo B del informe técnico se indica que este modelo puede utilizarse para diseñar un proceso de DCH. Para los métodos particulares que permitan implementar las prácticas, el texto remite al lector a los libros de texto y a los proveedores de servicios de factores humanos.

Acerca de la utilización de este informe técnico para aumentar el conjunto de procesos definidos según otros modelos de proceso, se indica que, normalmente, tal aumento se realiza al llevar a cabo una evaluación de la capacidad. Así mismo, se incluyen las siguientes matizaciones para este caso (cláusula B.3.2):

- El proceso HCD2 se requerirá casi siempre.
- Los procesos HCD3 al HCD6 probablemente se requerirán en la mayor parte de las evaluaciones.
- El proceso HCD1 puede ser más relevante para desarrollo genérico de sistemas (como productos domésticos).
- El proceso HCD7 es más relevante para sistemas de gran tamaño (como, por ejemplo, los sistemas de información del sector público).
- Los procesos HCD2 al HCD 4 pueden ser más relevantes en momentos tempranos del ciclo de vida, pero nótese que algunas prácticas de HCD5 y HCD6 se requieren muy pronto en el desarrollo de un sistema.
- La naturaleza iterativa del ciclo de vida centrado en el usuario también significa que los elementos HCD3 y HCD4 son todavía llevados a cabo durante el mantenimiento del sistema.

2.3.5.2 Análisis

Este texto ofrece una descripción más detallada de las actividades a realizar de la que se incluye en el estándar ISO 13407. De todas formas, no llega a ser tan detallada como para incluir las técnicas a aplicar en cada actividad.

En cuanto a la iteratividad del enfoque, éste puede considerarse verdaderamente iterativo, pues todas las descripciones y figuras transmiten adecuadamente la idea de un desarrollo iterativo. En las precisiones que se incluyen para la evaluación formativa, se destaca explícitamente que los resultados de este tipo de formación sirven para los procesos de requisitos y diseño, enfatizando así la componente iterativa de los procesos de DCH. De igual forma, el resto de características que definen un enfoque de desarrollo centrado en el usuario las cumple, por estar basado directamente del estándar ISO 13407.

Según se ha indicado en la descripción general de la aproximación, en este documento se remite al lector a libros de texto y proveedores de servicios de factores humanos, si quiere encontrar información sobre los métodos específicos que implementan las prácticas descritas. Debido a la gran heterogeneidad del campo de la IPO en cuanto a terminología de actividades, técnicas y productos, el trabajo de interpretar las prácticas y productos particulares indicados en este informe para cada método IPO del cual se quiera utilizar alguna de sus prácticas recomendadas, resulta una tarea ardua. Así, Bevan et al. [Bevan, 01b] indican tras utilizar este modelo para incorporar la usabilidad en el proceso de desarrollo de dos organizaciones, que fue preciso seleccionar y particularizar para cada una de ellas las técnicas centradas en el usuario, para satisfacer las necesidades del entorno de desarrollo.

Jokela [Jokela, 02] en su descripción de la utilización de este modelo en dos evaluaciones del proceso, indica que en la empresa no se llegó a entender el modelo, lo cual sugiere que no se trata de un modelo directamente asumible por desarrolladores.

El modelo propuesto no incluye información sobre su integración con el proceso de desarrollo más amplio, más allá de algunas indicaciones generales sobre los mejores momentos de aplicación de algunos procesos. No encontramos información acerca de las actividades del desarrollo más amplio con las que hay que integrar cada proceso de este modelo.

Por otra parte, la descripción de cada proceso o actividad que forma parte del modelo únicamente se limita a un listado de los productos obtenidos, y a una breve descripción de las prácticas o subactividades en las que se subdivide cada proceso. Esta información, aunque resulta de utilidad para la evaluación del proceso con respecto a la usabilidad, no es suficientemente detallada como para servir de guía para la integración de la usabilidad en el proceso de desarrollo.

2.4 Extensión de Métodos IS con Aspectos de Usabilidad

En esta sección se incluyen las propuestas que se centran en la extensión de métodos y procesos IS con aspectos de usabilidad.

2.4.1 Propuesta de Coutaz

2.4.1.1 Descripción

Como muestra de esfuerzos de integración tempranos, Coutaz [Coutaz, 94] propone explorar la intersección entre IS e IPO desde el punto de vista de las técnicas de evaluación. La autora indica que las pruebas de usabilidad han sido subestimadas en los esfuerzos de garantía de calidad software. Indica también que, aunque se considera al usuario final como importante, no se va más allá de hablar en términos elusivos del tipo “usabilidad” y “formación”. Propone las pruebas de usabilidad como punto de anclaje entre la IPO y las técnicas de evaluación de la IS.

Presenta una descripción de las técnicas de evaluación de la IPO, dividiéndolas entre predictivas y experimentales, según se muestra en la Figura 2.8.

Los modelos predictivos permiten la evaluación de IUs en la etapa de diseño, antes de implementarlo. La autora apunta como principal desventaja que la especificación de información para un modelo predictivo puede ser tan costoso en tiempo como una implementación. Por otra parte, la autora indican que las evaluaciones heurísticas requieren al menos tres expertos para descubrir un número razonable de los problemas de usabilidad existentes. Las técnicas experimentales son presentadas por la autora como un intento de solucionar estos problemas de las técnicas predictivas.

Sobre las técnicas experimentales se extiende la autora mucho menos que en las técnicas predictivas, y únicamente detalla un poco más la técnica del Mago de Oz (*Wizard of Oz*). La autora indica que los sistemas desarrollados para aplicar esta técnica han sido usados primariamente para estudiar el uso de lenguaje natural para sistemas de recuperación de información. La desventaja apuntada a este tipo de sistemas es que recogen un gran volumen de información comportamental, y además es difícil identificar los parámetros apropiados para el experimento.

En esta propuesta de integración se destacan un tipo de técnicas de usabilidad del campo de la IPO. En concreto, la autora hace mucho hincapié en las especificaciones formales de tareas como UAN, y lo une con los esfuerzos para crear generadores automáticos de IUs. El énfasis

en el automatismo lo lleva a las pruebas de usabilidad, donde plantea la corriente que integra herramientas para realizar la técnica del Mago de Oz con la captura automática de datos comportamentales específicos.

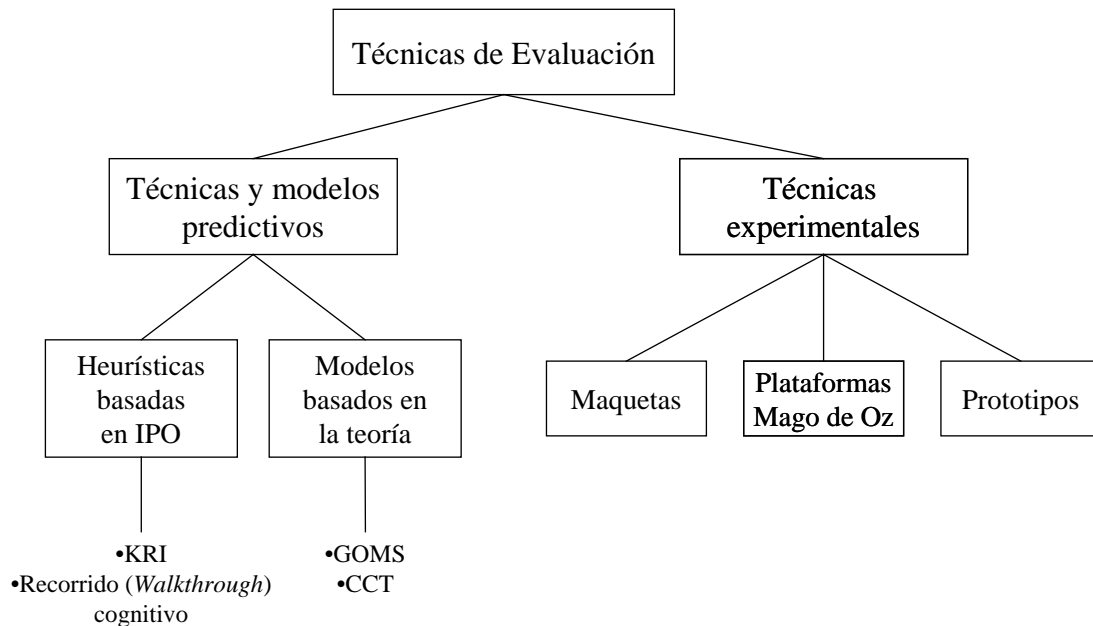


Figura 2.8 - Perspectiva de las Técnicas de Evaluación de la IPO según Coutaz

Cuando trata del enfoque de “diseño para usabilidad” lo describe como centrado en el usuario en todo el desarrollo e iterativo. La autora admite que el modelo de ciclo de vida en cascada no acepta tal condicionante, y el modelo V de proceso encaja un poco mejor pero sigue planteando las pruebas demasiado tarde. El modelo en espiral, sin embargo, es aceptado por la autora como un esquema adecuado para el problema tratado, esto es, es consistente con el diseño para usabilidad. Cada ciclo de la espiral podría contener sub-espirales para realizar pruebas concretas de usabilidad sobre subpartes exploratorias del sistema. Identifica la autora al modelo en espiral como apropiado para el desarrollo de sistemas prospectivos e innovativos cuyas funcionalidades no están completamente definidas, pero lo etiqueta como menos práctico para proyectos bien estructurados cuyo desarrollo está limitado en el tiempo. Propone entonces incluir pruebas de usabilidad en cada paso del desarrollo del modelo en V, para que soporte el diseño iterativo y pruebas incrementales de usabilidad.

La autora indica que las pruebas de usabilidad están pobremente integradas en los pasos iniciales del modelo en V, en el análisis de requisitos. Como primera recomendación, sobre el documento de requisitos indica que se debería utilizar el análisis de tareas como base para la especificación funcional, y usar herramientas como las notaciones MAD, TKS o UAN para expresar formalmente los resultados del análisis. Sobre el documento del Plan de Calidad, se identifica que el enfoque de evaluación de la ingeniería de usabilidad es consistente con el enfoque de garantía de calidad software, lo que, según la autora, abre la puerta a su integración. Apunta también que el principal obstáculo para tal integración están en la falta de educación interdisciplinar. Achaca a la definición vaga de usabilidad el que la comunidad software no aborde los objetivos adecuados de calidad para ellos, y nombra que hay intentos por definir criterios más específicos de usabilidad (pone como ejemplo un trabajo que la descompone en observabilidad, predictabilidad y honestidad).

En cuanto al diseño del sistema, la etapa en la que se desarrolla el documento de especificaciones externas, propone el uso de notaciones formales como UAN para poder realizar pruebas de usabilidad automáticas antes de la implementación.

Como conclusión, la autora resume los puntos en los que cada tipo de técnica de usabilidad se aplicaría:

- Las herramientas basadas en la teoría como GOMS, CCT, PUM e ICS, deberían emplearse en las fases iniciales del proceso de desarrollo como herramientas predictivas.
- Las heurísticas IPO y el Recorrido (*Walkthrough*) Cognitivo son aplicables una vez se tienen las especificaciones externas.
- Los enfoques experimentales basados en la captura de información comportamental para tests bien dirigidos (*well-targeted*) son mejores una vez se ha realizado la implementación. Ésta puede generarse “a mano” a partir de *toolkits* y esquemas de aplicaciones, o bien automáticamente vía generadores de IU.

2.4.1.2 Análisis

En cuanto al planteamiento general, observamos una contradicción en que la autora plantea la integración de técnicas de evaluación de la usabilidad, cuando en realidad propone el uso del análisis de tareas para la especificación de requisitos, que no es una técnica de evaluación.

En cuanto a la evaluación de la usabilidad experimental, podemos destacar que no detalla las típicas especificaciones de usabilidad en las pruebas con usuarios en las que se mide el rendimiento de tareas. Los aspectos comportamentales de los tests de usabilidad (los que tienen más relación con notaciones formales como UAN) son destacados muy por encima de otros aspectos. Además, es preciso mencionar que no plantea el estudio de usuarios y su contexto explícitamente, un pilar básico del enfoque centrado en el usuario. Así, únicamente lo podemos considerar parcialmente centrado en el usuario.

Respecto al enfoque iterativo, el método en V consideramos que es iterativo de una forma muy limitada, pues la iteración se reduce a realizar dos pasadas sobre las principales actividades del desarrollo. La autora aconseja el uso de V para proyectos bien estructurados y cuyo desarrollo está limitado en el tiempo. Para este tipo de proyectos, por tanto, la propuesta de Coutaz no sería verdaderamente iterativa, y, por tanto, consideramos que la propuesta en general no puede ser etiquetada como verdaderamente iterativa.

La propuesta es útil como referencia para la colocación en el proceso de desarrollo de las técnicas que nombra (Análisis Cognitivo de Tareas como GOMS y UAN, o la técnica de Recorrido Cognitivo). Sin embargo, la taxonomía de técnicas de evaluación no está actualizada en cuanto a las técnicas experimentales, pues únicamente nombra tipos de técnicas de prototipado, sin incluir las diversas variantes de los test de usabilidad y de los estudios de sistemas instalados. Estos últimos están descritos, por ejemplo, en [Mayhew, 99] y [Shneiderman, 98].

Podemos concluir destacando que cuando la autora habla de lo que podemos hacer para mejorar la integración, se decanta por las soluciones típicamente formales, apartadas de la esfera de los humanos y los problemas organizacionales, pues aboga por desarrollar especificaciones formales (por ej. para Análisis de Tareas) para alimentar a las técnicas predictivas, y soporte software para realizar test de usabilidad (por ejemplo, extendiendo las técnicas Mago de Oz del momento). Responde a una visión del desarrollo de software prevalente en la IS de los años 90, hacia una formalización cada vez mayor, sustentada por el uso de herramientas CASE. Este enfoque no es el más aceptado actualmente, en el que se tiende a un mayor grado de flexibilidad que permita afrontar los problemas de rigidez derivados de enfoques basados en el ciclo de vida en cascada.

2.4.2 MUSE (Method for USability Engineering)

2.4.2.1 Descripción

El método MUSE [Lim, 94] se centra principalmente en el diseño, pues, según los autores, una revisión de la literatura existente en la fecha de publicación (1994) mostraba que la contribución de factores humanos³ a las especificaciones de diseño es generalmente inadecuada e implícita. Para las tareas posteriores de evaluación, las técnicas existentes de evaluación de usabilidad sirven para dar soporte al método. Por tanto, la gran contribución del método, según sus autores, es en lo referente a la fase de especificación del diseño.

Para desarrollar MUSE, sus autores realizaron un estudio previo en el que se establecieron las características que un método debería tener para poder realizar una incorporación efectiva de técnicas de IPO. Estos requisitos identificados por los autores son los siguientes:

- El ámbito, producto y proceso de las técnicas IPO debería hacerse explícito.
- Las notaciones de las técnicas IPO deberían desarrollarse para dar soporte a descripciones de diseño más específicas.
- Las técnicas IPO deberían ser asociadas explícitamente con un marco de diseño estructurado (*structured design framework*). Este requisito lo basan en dos razones, a saber:
 - El entorno de diseño en etapas bien definido de un esquema estructurado puede soportar un encaje más preciso de las técnicas existentes con las necesidades y contextos del diseño de sistemas. Por tanto, las técnicas IPO relevantes pueden ser identificadas para su incorporación en las etapas apropiadas del desarrollo del sistema.
 - Las técnicas IPO existentes asociadas con un esquema de diseño estructurado podrían extenderse e integrarse para derivar un método estructurado IPO. Tal método podría dar soporte al desarrollo de sistemas de forma global, y también facilitar la identificación de temas de diseño de la IPO y de la IS que se solapan. Así, el producto, proceso y notación de las contribuciones de diseño de la IPO pueden posicionarse y temporizarse con respecto a las etapas del desarrollo del sistema de forma más apropiada.
- Deberían desarrollarse herramientas informáticas que siguieran la especificación de un método estructurado IPO razonablemente completo. En concreto, debería proveerse soporte computerizado para facilitar la generación, documentación y entrega de descripciones de diseño según prescriba el método IPO estructurado.

MUSE responde a estos requisitos. Es más, su elaboración partió concretamente de estas necesidades identificadas por sus autores.

³ Factores humanos (*human factors*) es el término que los autores usan para la IPO, según ésta se entiende habitualmente en la literatura. Aunque los autores indican que la disciplina IPO (*Human-Computer Interaction*) se conceptualiza como incluyendo las sub-disciplinas de factores humanos e IS, se trata de una visión que choca completamente con lo establecido en IS. Por tanto, en adelante nos vamos a referir siempre a IPO donde los autores usan el término factores humanos, para facilitar la comparación con otras propuestas

MUSE se estructura en tres fases, cada una de las cuales comprende una serie de etapas de diseño, según se observa en la Figura 2.9. El ámbito de cada fase es como sigue:

1. La Fase de Educación de Información y Análisis se ocupa de la captura de requisitos de usuario y del análisis de tareas. Sus etapas de diseño comprenden el Análisis de Sistemas Existentes y el Modelo de Tareas Generalizado.
2. La Fase de Síntesis de Diseño aborda la derivación de un diseño conceptual del sistema objetivo. Sus etapas de diseño comprenden la Declaración de Necesidades del Usuario, el Modelo Compuesto de Tareas, y el Modelo de Tareas del Usuario y el Sistema.
3. La Fase de Especificación de Diseño se centra en la definición funcional y el diseño de la IU. Comprende las etapas del Modelo de Tareas de Interacción, Modelo de la Interfaz y Diseño de Presentación.

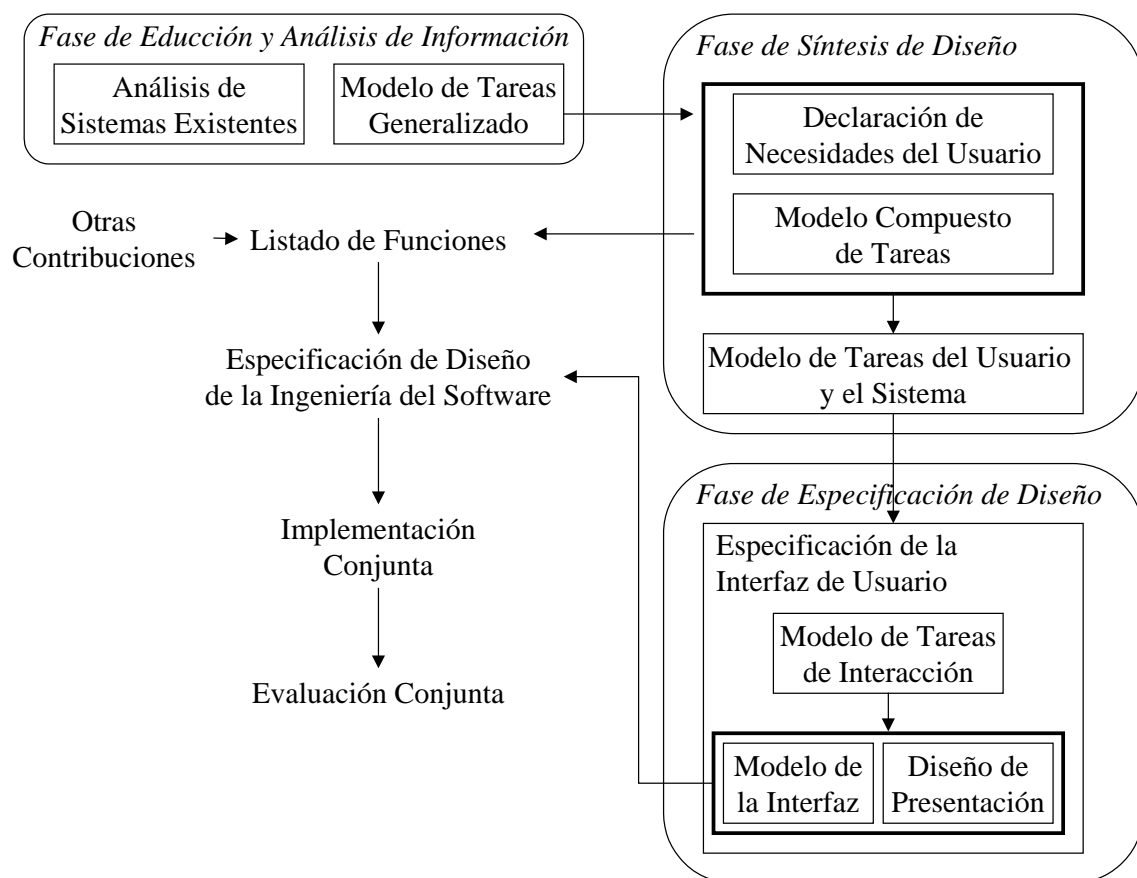


Figura 2.9 - Esquema Estructurado de MUSE

Cada etapa corresponde, según los autores, a una agrupación coherente de procesos de diseño que transforma las entradas en las salidas deseadas. Por tanto, corresponde a una concepción del desarrollo transformacional: Entrada-proceso-salida.

La estructura metodológica del método sigue una serie de principios de diseño aceptados, según sus autores:

- Dilación de los compromisos de diseño, asegurando que el diseño detallado es precedido por un análisis del diseño y una definición conceptual apropiadas.

- La realización de evaluaciones tempranas del diseño, bien de forma analítica por parte del diseñador, bien de forma empírica mediante un prototipo.
- La realización de diseño iterativo.
- La realización de un desarrollo del diseño modular o incremental. Específicamente, el diseño detallado puede ser llevado a cabo modularmente siguiendo la especificación de un diseño conceptual.

En la Tabla 2.2 se muestran las notaciones escogidas por los autores para los productos generados en cada etapa, agrupadas según el nivel que describe cada tipo de especificación. En la tabla se han destacado en cursiva aquellas técnicas que vienen del campo de la IS, algunas de las cuales se utilizan en su forma original, y otras han sido extendidas para reflejar elementos relevantes para la usabilidad del sistema. Se puede observar que las dos notaciones que más ampliamente se utilizan son las redes semánticas y los diagramas de estructuras, tanto los normales como los anotados. La técnica de diagramas de estructuras es extendida para permitir la asignación de funciones entre usuario y sistema, y para representar elementos referentes al diseño de la interacción.

En cuanto a las técnicas utilizadas en cada actividad, los autores describen las notaciones utilizadas para representar los productos (ver Tabla 2.2), mientras que para cada actividad describen los pasos que hay que tomar para generar el producto.

Se puede resumir la propuesta de MUSE indicando que enfatiza el análisis de tareas según se entiende en la IPO, haciendo que gire en torno a esta técnica tanto la educación de requisitos como el diseño de la interacción con el usuario. El método parte de las tareas y necesidades de los usuarios observadas, las refleja en un modelo de tareas, y transforma dicho modelo en un diseño de la interacción concreto. La integración con las actividades de la IS se hace mediante la asignación de ciertas tareas a la máquina, y mediante la especificación de las tareas interactivas en actividades de MUSE, que son alimentadas a actividades del proceso de IS (según se observa en la Figura 2.9).

2.4.2.2 Análisis

Se trata de una propuesta muy seria desde el punto de vista científico, para la integración de técnicas IPO en un marco de desarrollo estructurado. Los autores abogan por este enfoque debido a las supuestas bondades de un marco de desarrollo estructurado en sí mismo, y a las posibilidades de integración de la usabilidad en todo el desarrollo que ofrece el utilizar construcciones habituales de la IS. Los autores elaboran su propuesta tras un estudio minucioso de cuáles son los requisitos que debe cumplir un método estructurado de ingeniería de usabilidad como el que persiguen, y toman como objetivo el producir un método que permita integrar la usabilidad en la especificación del diseño (y no únicamente en tareas de evaluación).

Por un lado, podemos destacar que no describen explícitamente las técnicas como separadas de las actividades en las que son de aplicación. En vez de ofrecer una serie de técnicas para cada actividad, el método utiliza una técnica para cada actividad sin indicaciones de por qué ha sido escogida. Por otra parte, observamos que los autores no se ocupan de las actividades de evaluación de la usabilidad, puesto que esa parte de la IPO se identifica como ya cubierta. A pesar de que se nombra en las actividades que se deben realizar iterando, no se especifica explícitamente ninguna actividad de evaluación como tal en todo el proceso de diseño. Únicamente se hacen referencias en algunas actividades a que se debe realizar como último paso la validación de los productos con los expertos o usuarios (actividades dentro de la etapa de Análisis de Sistemas Existentes), y en otras (etapas de Modelo de la Interfaz y Diseño de la

implementación) se indica que se debe dar especial consideración al prototipado y a las pruebas con usuarios de los diseños propuestos. A pesar de que la evaluación de la usabilidad es una de las áreas más desarrolladas dentro de la IPO, todavía es bastante desconocida entre los desarrolladores, y sorprende su no inclusión en una propuesta de integración de la usabilidad en el proceso de desarrollo.

Tabla 2.2 - Notaciones Empleadas en MUSE para cada Nivel de Especificaciones de Diseño

Nivel de Especificación de Diseño	Especificaciones de Diseño del Sistema según la IPO	Notación que da Soporte a las Especificaciones IPO
Nivel organizacional (metasistema)	Estructura de las relaciones formales (e informales) de trabajo entre subsistemas	<i>Redes semánticas</i>
	Control de ejecución de los flujos de información entre subsistemas	<i>Diagramas de redes</i>
	Semántica del dominio de trabajo general	<i>Redes semánticas</i>
	Requisitos de rendimiento de cada subsistema	<i>Diagramas de flujo de funciones</i>
Nivel de tarea (subsistema)	Relaciones estructurales entre tareas	<i>Diagramas de estructuras</i>
	Asignación de funciones entre usuario y sistema	<i>Diagramas de estructuras anotados</i>
	Control de la presentación por parte del sistema de las funciones de soporte a las tareas como respuesta a ejecuciones de tareas del usuario	<i>Diagramas de estructuras anotados</i>
	Control de ejecuciones de tareas del usuario	<i>Diagramas de estructuras</i>
	Semántica del dominio de la tarea	<i>Redes semánticas</i>
	Requisitos de rendimiento de usuario y sistema	<i>Tablas, diagramas de flujo de funciones</i>
Nivel de interacción (entrada - salida)	Relaciones estructurales entre objetos de interacción	<i>Redes semánticas</i>
	Semántica de los objetos y acciones de interacción	Tablas de texto
	Representación de objetos de interacción	Imágenes
	Control de ejecución de las entradas del usuario	<i>Diagramas de estructuras</i>
	Comportamiento de los objetos de interacción en respuesta a acciones del usuario/sistema	<i>Diagramas de estructuras anotados</i>
	Composición (forma y contenido) de las ventanas de presentación (incluyendo los mensajes)	Imágenes y texto
	Control de presentación de las ventanas como respuesta a las acciones del usuario/sistema	<i>Diagramas de estructuras</i>

Sin embargo, la carencia principal en el enfoque de los autores yace en la elección de los métodos estructurados. El presente auge de los desarrollos basados en el paradigma orientado a objetos ha hecho decrecer la importancia de los métodos estructurados. Las propuestas más recientes de métodos IPO no hacen ni siquiera referencia a los métodos estructurados y, por el contrario, sí que la hacen a la posible integración con métodos orientados a objetos [Constantine, 99] [Nunes, 01] [Mayhew, 99] [Soares, 03a] [Flückiger, 03]. Las ventajas que ofrece MUSE a la hora de integrarlo con un método estructurado se convierten en dificultades

si se intenta integrar en un método orientado a objetos, puesto que los autores de MUSE hacen especial hincapié en que su método es estructurado.

Por otro lado, los autores presentan a MUSE como iterativo. Pero también indican que MUSE está basado en el principio de retrasar el compromiso con el diseño, con el fin de asegurarse de que el diseño detallado es precedido por un análisis del diseño y una definición conceptual apropiados. De forma adicional, al comparar MUSE con el enfoque de prototipado rápido, los autores afirman que MUSE, como método estructurado, enfatiza una fase de análisis y documentación del diseño anterior a la especificación de una solución basada en conjeturas (*first-best guess solution*). Así, MUSE tiene en su base un enfoque basado en el ciclo de vida en cascada, el cual es un obstáculo a la aplicación de un enfoque verdaderamente iterativo. Por lo tanto, no consideramos que el método soporte un enfoque centrado en el usuario.

Sorprende el papel preponderante de los diagramas de estructuras entre las técnicas que componen el método. Esta técnica de IS es ampliada para representar tanto la asignación de funciones entre usuario y sistema como para representar el control de ejecución de pantallas y el comportamiento de los objetos de interacción. Esta técnica tiene un grado de uso muy bajo en la actualidad. En este sentido, se puede destacar que uno de los creadores de la técnica de los diagramas de estructuras, Larry Constantine, cuyo trabajo se centra en la actualidad en el desarrollo de software usable [Constantine, 99], no la propone en el conjunto de técnicas de su método centrado en el uso. La complejidad de una IU de un sistema medio sería demasiado alta para poder manejarla con diagramas de estructuras como proponen los autores. Aparte de la complejidad de uso de algunas técnicas propuestas, la descripción del método en sí resulta farragosa. En este sentido, es preciso referirse a las conclusiones del proyecto BIUSEM [BIUSEM, 95], en el que se aplicó el método MUSE a tres proyectos de desarrollo en distintos campos, en conjunción con distintos métodos de IS, con el fin de evaluar la aplicabilidad de MUSE. A pesar de los resultados positivos del proyecto, en cuanto a que la aplicación de MUSE mejoró la calidad del producto y el hecho de compartir con el equipo de IS la visión de usabilidad ayudó a la educación de requisitos centrados en el usuario, en el informe final del proyecto se afirma que "el libro y los artículos publicados describiendo MUSE son innecesariamente complicados y actúan de forma disuasoria para un uso más amplio" [BIUSEM, 95].

Puesto que MUSE es un método estructurado, sus autores afirman que resulta fácil su integración con cualquier método estructurado, presentando en detalle su integración con el método JSD (*Jackson System Development*) como ejemplo. El método JSD es etiquetado por los autores como un método que se utiliza principalmente para el desarrollo de sistemas de tiempo real. Este tipo de sistemas no suelen ser interactivos por tratarse de sistemas embebidos, pero incluso considerando aquellos sistemas de tiempo real que sí son interactivos, éstos representan una parte muy pequeña del conjunto de sistemas interactivos que se desarrollan. Por lo tanto, el detalle de la integración de MUSE con JSD no resulta muy útil desde un punto de vista general. Al detallar la integración de MUSE con JSD, se observa que únicamente existen dos puntos de contacto entre ambos métodos: La parte de análisis de usuarios y sus tareas, y la de diseño de la IU. En realidad, la parte más importante de integración, que es la parte de análisis de tareas del usuario, dentro de las actividades de requisitos, no es abordada por el método JSD. Los autores indican que MUSE y JSD se complementan en este sentido, puesto que JSD se centra únicamente en modelar lo que debe hacer el sistema y no las tareas de los usuarios. El otro punto de contacto consiste en la parte de diseño de la IU, la cual se desarrolla a partir de la información de flujos de salida modelados en JSD. En resumen, podemos decir que la elección de JSD como muestra de integración de MUSE con un método de IS es poco ilustrativa en cuanto a integración de actividades de usabilidad con tradicionales, especialmente en lo referente a actividades de requisitos. En los métodos actuales de IS sí que se realiza una especificación de las tareas del usuario con el sistema, por ejemplo mediante la técnica de casos de uso [Jacobson, 93], que es ampliamente utilizada actualmente en el desarrollo orientado a objetos. Sí es cierto, no

obstante, que métodos anteriores de desarrollo orientado a objetos como OMT (*Object Modeling Technique*) [Rumbaugh, 91] y el método de Booch [Booch, 94] adolecían de carencias en esa parcela del desarrollo. Por tanto, podemos concluir que MUSE fue un esfuerzo muy notorio en cuanto a la integración de las técnicas y actividades de usabilidad en el proceso de desarrollo de la IS, visto según la perspectiva del desarrollo en la fecha de su publicación (1994), pero hoy día adolece de importantes carencias que hacen que su integración con los principales procesos de desarrollo actuales de la IS presente un alto grado de dificultad.

2.4.3 Propuesta de Hakiel

2.4.3.1 Descripción

Hakiel [Hakiel, 97] identifica la situación en la que los jefes de proyectos software únicamente se preocupan porque el software desarrollado no tenga defectos en su código, y se olvidan de la importancia de la usabilidad del producto, en cuanto su visión de la calidad se reduce a la falta de defectos en el código. Propone expresar las actividades de usabilidad en el desarrollo mediante los entregables que producen, y su relación con los productos del desarrollo de software general. Propone que la organización de desarrollo de software realice una transición entre una visión orientada al software a una orientada al producto. Para esta transición, el primer paso desde un punto de vista de factores humanos, consiste en definir los entregables que expresan una perspectiva de usabilidad en el diseño y en el proceso de desarrollo en general.

El autor expresa la necesidad para los expertos en usabilidad, de cara a promover la apreciación de sus habilidades, de describir su contribución en términos de actividades de desarrollo del producto y entregables, en vez de utilizar terminología propia de la IPO.

Como ventajas de su enfoque, en el que se enfatizan entregables en vez de actividades, procedimientos y métodos, el autor nombra las siguientes:

- Las relaciones entre actividades de usabilidad y factores humanos y otros entregables del ciclo de vida están claramente identificadas y pueden ser gestionadas fácilmente.
- Lo que ofrecen los factores humanos y la usabilidad se expresa en términos del ciclo de vida del producto, no en términos de métodos y técnicas o terminología IPO.
- La relación entre entregables de usabilidad y factores humanos y los puntos de control del diseño y el desarrollo es independiente del modelo de proceso particular utilizado. Pueden variar en el orden de las actividades, pero no en los entregables generados por el proceso.
- Las actividades de usabilidad y factores humanos son parte del proceso de desarrollo, no un proceso separado cuya integración hay que gestionar.
- Lo que ofrecen los factores humanos y la usabilidad está especificado independientemente de métodos y técnicas particulares de factores humanos, y pueden ser seleccionadas y escaladas de acuerdo a la planificación del proyecto y a la disponibilidad de recursos.
- Una especificación de un proceso de usabilidad (o de cualquier proceso) orientada a entregables es una precursora de una especificación del ciclo de vida más rigurosa y

orientada a objetos, y facilita la formulación de la gestión de proyectos soportada por ordenador y de procesos de flujos de trabajo.

Para establecer los entregables de usabilidad para el diseño y el desarrollo el autor parte de la hipótesis de que ya se han llevado a cabo actividades anteriores de marketing que identifican las oportunidades del producto y han derivado los requisitos de mercado a abordar por parte de un proceso de diseño y desarrollo. Para el autor los procesos de diseño y desarrollo simplemente identifican las actividades y los entregables a través de los cuales los requisitos se transforman en un producto. El autor identifica seis etapas típicas de cualquier proceso de diseño y desarrollo, y el enfoque de IPO orientado a entregables que propone identifica las actividades de usabilidad y factores humanos refiriéndose a estas etapas:

- **Requisitos de producto:** El entregable IPO que contribuye a los requisitos de producto es una especificación de requisitos de uso. Estos se derivan de la identificación y análisis de:
 - Las tareas afectadas por el sistema.
 - Los usuarios y escenarios de uso.
 - Las características de usabilidad del producto actual o competidores.
 - Los deseos y necesidades del cliente y el usuario.

- **Diseño de alto nivel del producto:** El entregable IPO que contribuye al diseño de alto nivel es la especificación de los atributos de uso clave del producto. Estos atributos identifican:
 - Las tareas particulares y los requisitos de uso a tomar en cuenta por el producto.
 - Objetivos medibles de uso y usabilidad para el producto.
 - Las tareas del usuario soportadas o requeridas por el diseño del producto.
 - Cambios a tareas actuales implicados por el producto.
 - Requisitos de interacción para cada tarea incluyendo los conceptos de aplicación a través de los que las interacciones se van a expresar.

- **Diseño de bajo nivel del producto:** El entregable IPO correspondiente es la especificación del diseño de la IU. Incluye tres elementos:
 - Los elementos del modelo de objetos de la IU.
 - Los modos de interacción a explotar en la creación de las interfaces de usuario (GUI, voz, teclado telefónico).
 - Diseños específicos de la IU para cada actividad del usuario, para cada uno de los modos de interacción a implementar, incluyendo las disposiciones de pantalla, menús, terminología, ayuda en línea y otros componentes del producto con los que los usuarios van a tener contacto directo.

- **Desarrollo de código y pruebas funcionales:** No hay ningún entregable IPO en esta etapa, pero sí que pueden necesitarse expertos en factores humanos y usabilidad para ayudar a resolver problemas en la construcción de las IUs.

- **Pruebas de sistema:** El entregable IPO correspondiente a estas pruebas es la evaluación de usabilidad tradicional. Se supone que ya se han hecho evaluaciones con prototipos de los entregables de las etapas anteriores, pero debido a consideraciones de escala, estrés e integración puede ser necesario también evaluar el sistema finalizado.

- **Disponibilidad:** Los entregables IPO producidos en esta actividad son los productos de evaluaciones de campo, menos formales, sobre la facilidad de uso de un producto, y se basan en la experiencia de usuarios del producto en el sitio del cliente. Estos entregables incluyen valores de satisfacción y requisitos de uso para ser considerados en la siguiente versión, así como defectos de usabilidad a ser mejorados en la versión actual.

En cada etapa antes de la codificación (desarrollo de código y pruebas funcionales) incluye la validación de los entregables mediante una evaluación basada en prototipado e iteración.

En la Figura 2.10 se puede observar un ejemplo de planificación del diseño de la usabilidad.

entregables de diseño de usabilidad	requisitos	diseño de alto nivel	diseño detallado
requisitos de uso * impactos primarios de uso * escenarios de uso actuales * usabilidad "mejor de la camada" * requisitos de usuario/cliente			
diseño de usabilidad de alto nivel * requisitos de uso del producto * objetivos de uso del producto * escenarios y tareas del producto * impacto de las tareas del producto * requisitos de interacción del usuario			
diseño de la interacción del usuario * métodos de interacción * modelo de objetos de la interfaz * diseño detallado de la IU			

Figura 2.10 - Ejemplo de Planificación del Diseño de la Usabilidad

El autor afirma que no impone una secuencia temporal rígida entre las actividades en cuanto al momento de comienzo, pero sí impone un orden entre los entregables referente a su momento de compleción. Por ejemplo, el diseño detallado de la IU no puede ser completado antes de que se haya completado el modelo de objetos de la interfaz, aunque sí pueden ir desarrollándose con cierto grado de paralelismo, según el autor.

Finaliza el autor cuestionando la política de calidad basada en el código "cero defectos", que identifica calidad únicamente con rendimiento y fiabilidad. La calidad incluye también, según el autor, por ejemplo, instalabilidad, usabilidad, mantenibilidad y documentación.

2.4.3.2 Análisis

Esta propuesta parece estar centrada en la producción de software dirigido al mercado (*off-the-shelf*), pues nombra que los requisitos de producto deben haber sido establecidos por los responsables de marketing, y en la disquisición final sobre calidad, habla sobre la necesidad de que el software cumpla los requisitos de usabilidad del mercado al que va dirigido, puesto que en caso contrario nos arriesgamos a perder cuota de mercado frente a los competidores que sí tengan un visión de perspectiva múltiple acerca del desarrollo. También nombra el autor que hay que recuperar la filosofía de desarrollo de sistemas, por lo que su aproximación se basa en la experiencia de desarrollo de sistemas embebidos y similares, un mercado mayoritariamente de consumidores finales (frente a los sistemas a medida).

La propuesta parte de una necesidad ya identificada por estudios de mercado o similares, por tanto no tiene en cuenta la importante etapa inicial de desarrollo del concepto del producto y el establecimiento de la necesidad. Esta concepción de comenzar ya con el concepto del producto definido hace que la propuesta esté basada en gran medida en la idea de requisitos congelados, y por tanto adolece de cierta tendencia a pensar en el desarrollo según el ciclo de vida en cascada. El ejemplo que el autor incluye de planificación de las actividades de usabilidad, descrito en la Figura 2.10, refleja claramente una concepción del desarrollo según el ciclo de vida en cascada, a pesar de que el autor indique en el texto que no se trata de una secuencialidad rígida. Por tanto, consideramos que la propuesta no cuenta con un verdadero enfoque iterativo.

Para ser considerada como centrada en el usuario, aparte de lo relacionado con la iteratividad del enfoque, la propuesta debería tratar una adecuada comprensión de los requisitos referentes al usuario y a las tareas que debe completar, además de implicar activamente a usuarios finales. Mientras la descripción de entregables sí permite suponer que se aborda la comprensión de los requisitos de usuarios y sus tareas correctamente, sin embargo el autor no hace referencia a la posible implicación activa de usuarios. Por tanto, y sin considerar el grado de iteratividad, únicamente podemos considerar la propuesta como parcialmente centrada en el usuario.

La idea de presentar la integración mediante entregables es buena porque ese enfoque suele coincidir con el de los jefes de proyecto, les permite introducir aspectos de usabilidad en su mapa mental del desarrollo. No requiere que tengan conocimiento de los métodos IPO como parece que requieren otros autores que abogan por la adopción por parte de los ingenieros software de los métodos IPO.

Como un primer paso para la integración es muy interesante, y apunta ideas que sería interesante desarrollar para convertirlo en algo más definido. De todas formas, el grado de detalle que da de los entregables no es suficiente como para poder aplicar esta propuesta directamente, y falta información sobre las técnicas a aplicar. El mismo autor apunta a lo inacabado de la propuesta, indicando que una especificación de un proceso de usabilidad orientada a entregables es un precursor de una especificación del ciclo de vida más rigurosa.

2.4.4 LUCID

2.4.4.1 Descripción

LUCID (*Logical User Centered Interaction Design*) es presentada por Shneiderman como ejemplo de metodología de desarrollo [Shneiderman, 98], y por Anderson et al. [Anderson, 01] como esquema básico de etapas para su propuesta ad-hoc de integración (ver sección 2.4.6).

LUCID no está públicamente disponible puesto que se trata de un producto comercial vendido por la empresa *Cognetics Corporation*. Al tratarse de una metodología propietaria, cuya compañía explota mediante servicios de consultoría, la literatura disponible sobre LUCID es escasa. La única información públicamente disponible encontrada por el autor del presente trabajo de investigación, son las tres páginas que le dedica Shneiderman en [Shneiderman, 98], la información en la página web de la compañía [Cognetics, 03], y la descripción que Kreitzberg, creador de LUCID y director de la compañía, realiza en [Kreitzberg, 96] de una versión preliminar de LUCID denominada "*The Cognetics QUE Design Methodology*", en la que la terminología de etapas es diferente y hay algunas diferencias con la descripción de cada etapa ofrecida en las referencias más recientes. Hemos incluido esta última referencia por tratarse de la que proporciona Shneiderman acerca de LUCID en [Shneiderman, 98].

Tras una petición de información más amplia al creador de LUCID (director de la compañía *Cognetics Corp.*) para poder incluir su estudio en el presente trabajo de investigación, el autor del presente trabajo únicamente ha recibido una presentación de 28 transparencias en formato *Microsoft Powerpoint*. Dicha presentación también ha sido considerada como fuente de información sobre LUCID en esta sección.

Según la compañía que vende LUCID, se trata de un marco (*framework*) que proporciona un contexto en el que llevar a cabo las actividades de usabilidad, de diseño de la IU, y de producto. También proporciona el fondo y herramientas para gestionar dichas actividades, aunque no hemos podido tener acceso a las mismas por tratarse de un producto comercial. Según sus autores, el marco propuesto en LUCID permite a los especialistas en usabilidad coordinar sus prácticas con el proceso de desarrollo general.

En la Figura 2.11 se muestran las seis etapas de LUCID. Cada una de ellas, según *Cognetics Corporation*, incluye asuntos de gestión, actividades de diseño y herramientas.



Figura 2.11 - Etapas de LUCID

Las etapas de LUCID son las siguientes:

- **Imaginación (*Envision*):** En esta etapa se establecen y conjugan los objetivos de negocio, usuario y tecnológicos. También se reúne el equipo del proyecto, y se consigue una comprensión común del concepto del producto, así como de los objetivos y limitaciones del proyecto. Se establecen unos objetivos de usabilidad preliminares.
- **Descubrimiento:** En esta etapa se analizan los usuarios, sus tareas, y la información que manejan. Se definen personas para clasificar usuarios, y escenarios para ilustrar sus tareas. De las personas y los escenarios se derivan los requisitos de alto nivel. También se revisan los objetivos de usabilidad.
- **Diseño Base:** En esta etapa se establece el concepto básico de la IU, sus objetos y sus metáforas. Se documenta la navegación, los tipos básicos de pantallas, y el modelo mental que subyace en dicha IU. Se crea un prototipo de ventanas clave, y se llevan a cabo sobre él evaluaciones de usabilidad de alto nivel.
- **Diseño Detallado:** En esta etapa se completa el diseño de la IU, generándose especificaciones para los desarrolladores. Tales especificaciones incluyen una guía de estilo, disposición de los elementos de las ventanas, y especificaciones elemento por elemento. Se llevan a cabo evaluaciones de usabilidad de ventanas o flujos de trabajo específicos.
- **Construcción:** En esta etapa se da soporte a los desarrolladores y se tratan las cuestiones de diseño de última hora. Si es necesario se pueden llevar a cabo evaluaciones de usabilidad de las ventanas críticas.
- **Lanzamiento (*Release*):** En esta etapa se diseña y prueba la experiencia inicial (*out-of-the-box*) del usuario. Se mide la satisfacción del usuario y la usabilidad del producto desarrollado. Con esta etapa se termina el proyecto.

Kreitzberg indica que las etapas no tienen por qué ser llevadas a cabo rígidamente de una forma secuencial, y que en algunos proyectos puede ser necesario acelerar o retrasar elementos del proceso de ingeniería de usabilidad para que puedan coordinarse con aspectos IS o con requisitos de marketing. Asegura también que en dichas ocasiones el proceso de gestión puede resultar bastante complejo.

2.4.4.2 Análisis

LUCID presenta las etapas necesarias para llevar a cabo un enfoque centrado en el usuario de forma ordenada, con modelos o productos a producir en cada etapa que se basan en los productos elaborados en etapas anteriores. Consideramos que la estructura en forma de etapas está basada en el ciclo de vida en cascada, pues la Figura 2.11 sigue la misma estructura de etapas consecutivas, sin que se muestre ninguna vuelta atrás que sugiera iteratividad. A pesar de la afirmación de los autores de que no tiene por qué ser secuencial, su representación así lo sugiere, por lo que puede fácilmente llevar a quienes lo apliquen a una aproximación de ciclo de vida en cascada. Por tanto, y a la vista de la escasa información disponible sobre LUCID, no podemos afirmar que siga un enfoque verdaderamente iterativo.

Las técnicas a aplicar en cada etapa no están descritas en la información disponible de forma inequívoca, puesto que hay que interpretar en la descripción de cada etapa qué técnicas se aplicarían. Por otra parte, al no referirse a las técnicas explícitamente, no se proporcionan referencias para dichas técnicas. Así, no podemos obtener un listado de las técnicas a aplicar en cada etapa a partir de la información disponible.

Respecto a una adecuada comprensión de los requisitos de usuarios y sus tareas, podemos afirmar que sí se considera, por ser la base de etapa de Descubrimiento. La participación de usuarios como parte importante de los esfuerzos de usabilidad se nombra en [Kreitzberg, 96], por lo que consideramos que debe estar incluida en LUCID una participación activa de usuarios. Así pues, podemos considerar que la propuesta es acorde con un enfoque centrado en el usuario, si exceptuamos la condición de iteratividad.

A pesar de que el autor presenta LUCID como un marco para integrar la usabilidad en el proceso de desarrollo más amplio, no se explica en qué se basa para realizar tal afirmación. Las etapas de LUCID son propias de esta propuesta, y no se justifica su relación con etapas típicas de desarrollos IS. En la parte de requisitos sí se nombran los requisitos de usuario y la gestión de riesgos, de los cuales se encargan normalmente los procesos de la IS, sin embargo no se hace referencia a cómo se integrarían las actividades de LUCID con las tradicionales. Así, el tema de la integración no se aborda en las fuentes disponibles en cuanto a la relación con el proceso de desarrollo.

2.4.5 Propuesta de Costabile

2.4.5.1 Descripción

Costabile [Costabile, 01] hace una propuesta de integración de las prácticas centradas en el usuario en el proceso software con el fin de incrementar la usabilidad del producto. El objetivo de su propuesta es clarificar el papel de los procesos y métodos de usabilidad en el ciclo de vida software. Introduce las nociones de lo que es un proceso centrado en el usuario, que se basa en los siguientes principios, según Costabile:

1. Analizar los usuarios y sus tareas.

2. Diseñar e implementar el sistema de forma iterativa a través de prototipos de complejidad creciente.
3. Evaluar las decisiones de diseño y el uso de prototipos con usuarios.

Costabile, para la modificación del ciclo de vida software para incluir la usabilidad, toma como base el modelo de ciclo de vida en cascada, al que etiqueta de "estándar". Detalla los principales inconvenientes de este modelo en su forma original, desde el punto de vista de la usabilidad:

- Las pruebas son al final.
- Los requisitos se suelen negociar con los clientes y no con los usuarios.
- Los requisitos están dedicados únicamente a la funcionalidad del sistema y no a lo fácil que será de aprender, la eficiencia de uso, etc.

Aumenta el ciclo de vida en cascada con tres cajas, una de Análisis de Usuarios y Tareas, otra de Especificación de Escenarios y de IU, y una tercera caja de Prototipado y Pruebas que se repite, según se muestra en la Figura 2.12. En la figura se han resaltado con fondo gris las nuevas actividades añadidas al ciclo de vida en cascada básico.

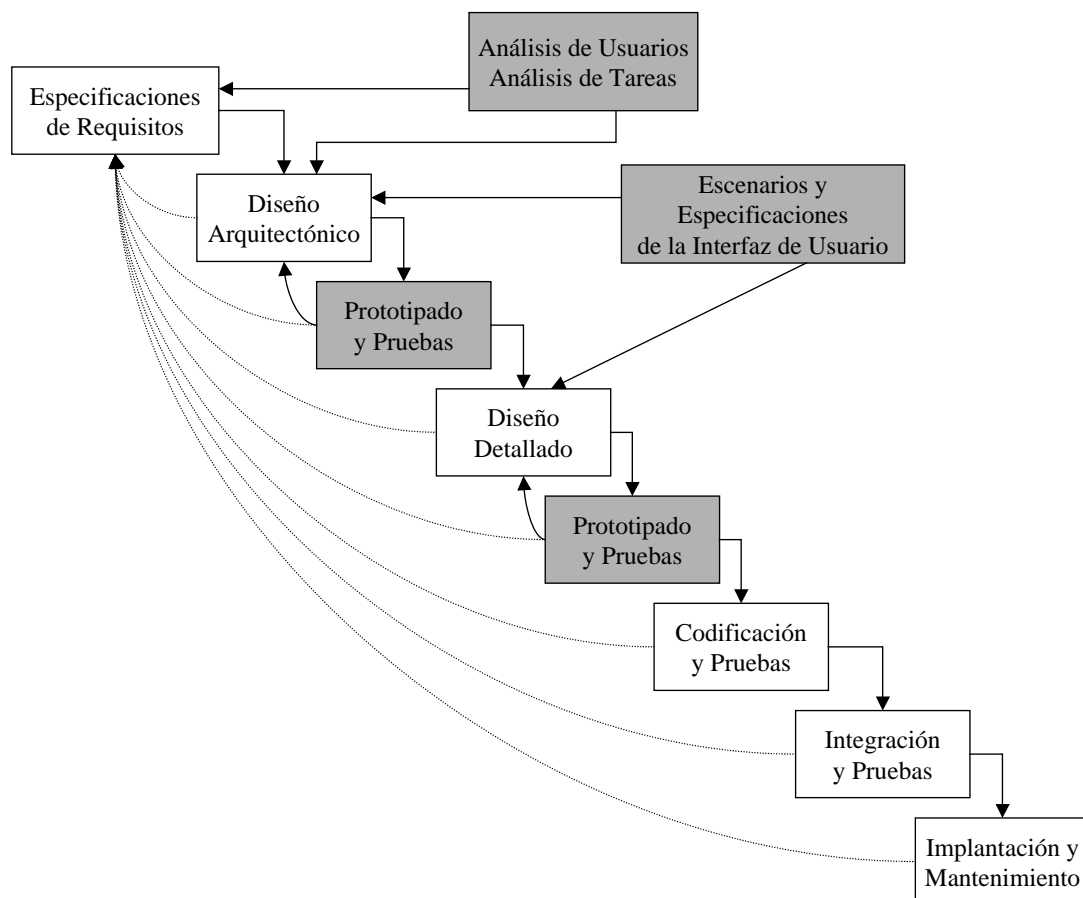


Figura 2.12 - Ciclo de Vida en Cascada Extendido por Costabile con Actividades de Usabilidad

Con la inclusión de vueltas hacia atrás justifica el enfoque iterativo necesario según un proceso centrado en el usuario. La autora intenta justificar la propuesta describiendo el proceso de ISO 13407 y el Ciclo de Vida en Estrella de Hix-Hartson, pero no razona cómo su propuesta se basa en lo que dicen ambos procesos centrados en el usuario.

En cuanto a las técnicas a emplear, refiere al trabajo de Hackos y Redish [Hackos, 98] para técnicas de análisis de usuarios y tareas, y para la caja de Escenarios y Especificaciones de la IU nombra la técnica de Escenarios referida en Preece et al. [Preece, 94].

Sí dedica la autora una sección a un análisis más detallado de los métodos de evaluación de la usabilidad. Entre los métodos de inspección de usabilidad nombra la Evaluación Heurística, pero también nombra el Recorrido (*Walkthrough*) Cognitivo, las Inspecciones Formales de Usabilidad y las revisiones de conformidad con guías de diseño o con estándares. Las técnicas basadas en usuarios, la autora las clasifica en métodos experimentales, métodos observacionales y técnicas de realización de encuestas. Entre los primeros nombra los experimentos controlados, entre los segundos la técnica de Pensar en Voz Alta (*thinking aloud*), y entre las terceras nombra los Cuestionarios y Entrevistas.

Introduce brevemente el UAF (*User Action Framework*), como un marco para organizar y unificar los esfuerzos de inspección de usabilidad, guías de diseño, clasificación e informes de usabilidad; integrados en una base de conocimiento. Introduce también la idea de patrones de evaluación que guían al inspector en su actividad, descritos en otras publicaciones de la autora.

2.4.5.2 Análisis

Esta propuesta se puede considerar como una primera aproximación a la problemática de introducir actividades de usabilidad en el desarrollo de software. El artículo únicamente da una visión general, representada en la Figura 2.12. En cuanto a la modificación del ciclo de vida para incluir usabilidad, no entra en suficiente detalle como para permitir a un desarrollador medio la integración de las actividades de usabilidad en su proceso de desarrollo software.

La decisión de tomar el ciclo de vida en cascada como base limita en gran medida la aplicabilidad de la propuesta. El enfoque del ciclo de vida en cascada tiene actualmente un contexto muy claro en la IS, proponiéndose únicamente para problemas bien definidos de complejidad baja o media, para dominios con los que el equipo de desarrollo está muy familiarizado. Glass indica que "los requisitos frecuentemente cambiaban a medida que el desarrollo avanzaba [...] Los expertos sabían que la cascada era un ideal inalcanzable" [Glass, 03], y afirma que el ciclo de vida en cascada está obsoleto. Larman identifica los siguientes problemas con el ciclo de vida en cascada: Mitigación tardía de los riesgos, especulación e inflexibilidad de los requisitos y el diseño, alta complejidad y baja adaptabilidad [Larman, 02]. Se puede afirmar que en el desarrollo de software en dominios no conocidos y para sistemas de complejidad media/alta, el equipo de desarrollo va avanzando en la comprensión del problema según se avanza en el diseño de la solución y en su adecuación a la realidad mediante la implementación. En el campo de la IPO, Hix y Hartson admiten en [Hix, 93] esta realidad al afirmar que, a pesar de lo que las metodologías les indican en cuanto a fases separadas y/o enfoques de descomposición (*top-down*), los diseñadores suelen trabajar en oleadas, variando el nivel al que se enfrentan en cada momento, y que una vez el sistema está construido documentan el proceso de tal manera que parezca que han aplicado las directivas que les habían sido suministradas. Un verdadero enfoque iterativo es más acorde con esta realidad humana, y afronta la mayoría de los problemas identificados en el ciclo de vida en cascada. Una organización de desarrollo de software que siga un verdadero enfoque iterativo podría plantearse seguir la propuesta de Costabile, pero para ello tendría que asignar las etapas de un desarrollo iterativo a etapas del ciclo de vida en cascada, lo cual es el error más común que hace que no se lleven a cabo desarrollos verdaderamente iterativos [Larman, 02] [Kroll, 03].

Por otra parte, en cuanto a la selección de técnicas de usabilidad para aplicar en el proceso de desarrollo, la propuesta no detalla los criterios seguidos para la selección de técnicas de Análisis de Usuarios y Tareas, ni para las de Especificación de la IU. Consideramos que la selección para estas dos actividades es pobre, puesto que el campo de la IPO es mucho más rico en cuanto a técnicas. En particular, podemos observar esta carencia en la parte de especificación de la IU, en la que únicamente se refiere a la técnica de Escenarios, mientras que hay un cierto número de técnicas que pueden ser útiles en este tipo de actividad, como, por ejemplo, los Árboles de Menús [Shneiderman, 98], los Mapas de Navegación entre Contextos [Constantine, 99], o la Guía de Estilo del Producto [Mayhew, 99]. En cuanto a la descripción de técnicas de evaluación, favorece excesivamente la Evaluación Heurística frente a otras técnicas como el Test de Usabilidad con usuarios reales, que es el principal método de evaluación de la usabilidad preconizado en la IPO.

En resumen, podemos afirmar que la propuesta no está suficientemente detallada como para ser aplicada directamente, la integración con las actividades tradicionales está descrita someramente y, lo más importante, no sigue un enfoque verdaderamente iterativo al tomar como ciclo de vida base el ciclo de vida en cascada.

2.4.6 Propuesta ad-hoc de Anderson et al.

2.4.6.1 Descripción

Anderson, Fleek, Garrity, y Drake [Anderson, 01] presentan en un artículo del número especial de la revista IEEE Software dedicado a la ingeniería de usabilidad [IEEESoftware, 01], la experiencia de integración de prácticas de usabilidad en un proceso de desarrollo basado en RUP, particularizado según las necesidades de la compañía para la que trabajan. Según detallan los autores, la empresa SMS decidió implementar una metodología de desarrollo orientada a objetos universal. La dirección de la empresa pretendía mejorar la satisfacción del cliente, y se vio la introducción de prácticas de usabilidad como el medio para lograr dicho objetivo. El objetivo era combinar las mejores prácticas de análisis y diseño OO con técnicas de usabilidad. El equipo formado para la integración contaba con miembros tanto del equipo IS como del equipo de usabilidad.

El equipo de usabilidad propuso basarse en el enfoque de investigación contextual de Beyer y Holtzblatt [Beyer, 98] para la Educación de Requisitos, en el método centrado en el uso de Constantine y Lockwood [Constantine, 99] para otras técnicas de usabilidad, y en LUCID [Cognetics, 03] como guía del proceso de usabilidad a través de las fases de proceso de desarrollo. El equipo de ingenieros software propuso utilizar RUP [Jacobson, 99] como base de conocimiento para representar el proceso de desarrollo de la compañía que se estaba definiendo.

Según los autores, se empleó bastante tiempo en intentar superar las dificultades de comunicación entre los dos equipos, en cuanto a que cada equipo usaba distintos procesos, vocabulario, herramientas y perspectiva. No se hizo verdadero progreso hasta que se empezaron a centrar en el propósito de cada actividad en ambos procesos, lo que permitió ir más allá de la terminología y diferencias de secuencia. Los temas de cambio y cultura organizacional se mostraron tanto o más desafiantes que los técnicos.

En esta propuesta, las actividades de usabilidad y sus entregables emergen durante los pasos de captura y análisis de requisitos y visión. La Figura 2.13 muestra cómo los requisitos se recogen, analizan y se diseña la visión del producto iterativamente durante las fases de inicio y elaboración. La definición de requisitos se divide en cuatro pasos:

1. Crear conceptos. El plan de negocio es el principal entregable de este paso.

2. Reunir requisitos. Usando la técnica de investigación contextual.
3. Analizar requisitos. Se compilan los perfiles de usuario.
4. Diseño de la visión del producto. El equipo desarrolla un modelo que transmita la visión de la estrategia del producto final para cumplir los requisitos.

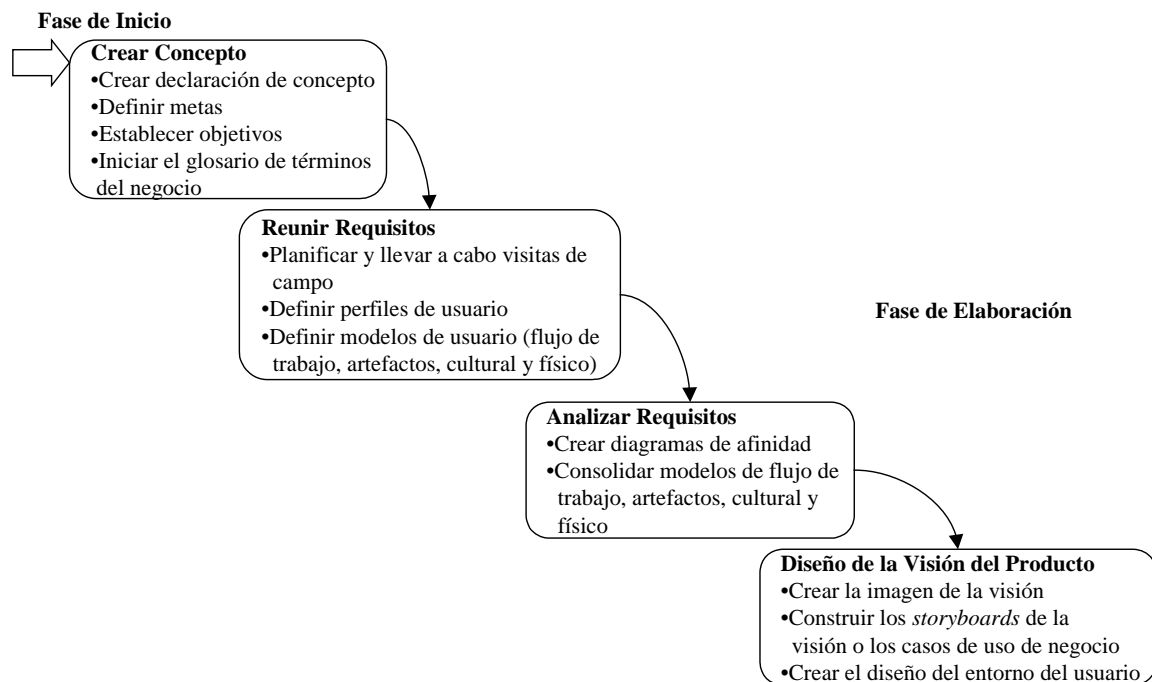


Figura 2.13- Una Vista de Alto Nivel de la Definición de Requisitos Durante el Desarrollo Software

Según los autores, esta visión del producto es lo que llaman Beyer y Holtzblatt el modelo de diseño del entorno del usuario, Cognetics lo llama *roadmap*, y Constantine y Lockwood lo llaman el modelo contextual. Se acordó que los equipos de desarrollo podrían utilizar una combinación de diagramas de contexto, casos de uso y *storyboarding* para crear la visión del producto, logrando así la integración de este componente clave del desarrollo centrado en el usuario en el proceso de desarrollo.

Las actividades de requisitos y visión alimentan al diseño tanto del sistema como de la IU, como puede observarse en la Figura 2.14.

Los autores destacan que el proceso de desarrollo propuesto puede parecer una solución genérica, pero se necesita buen juicio para adaptarlo y para elegir las actividades apropiadas para cualquier proyecto específico y sus condicionantes de tiempo. También indican que todas las actividades del proceso de desarrollo son potencialmente iterativas.

Para definir los roles de usabilidad en el equipo de desarrollo han realizado una interpretación relajada de la propuesta de Mayhew. Los roles definidos por los autores son cuatro: Ingeniero de usabilidad, diseñador de la IU, evaluador de usabilidad y el usuario. Es preciso destacar que la principal misión del ingeniero de usabilidad es relativa a los requisitos.

Actuando como consultor en un proyecto, uno de los autores se dio cuenta de que el alto número de modelos necesario para informar sobre las observaciones de campo consumía demasiado tiempo. Los autores crearon entonces un nuevo modelo que relacionara los modelos de secuencia y comunicación de Beyer y Holtzblatt con la notación UML: El modelo

de flujo de trabajo. Este modelo acomoda la información de los estudios de campo en un formato comprensible para los analistas.

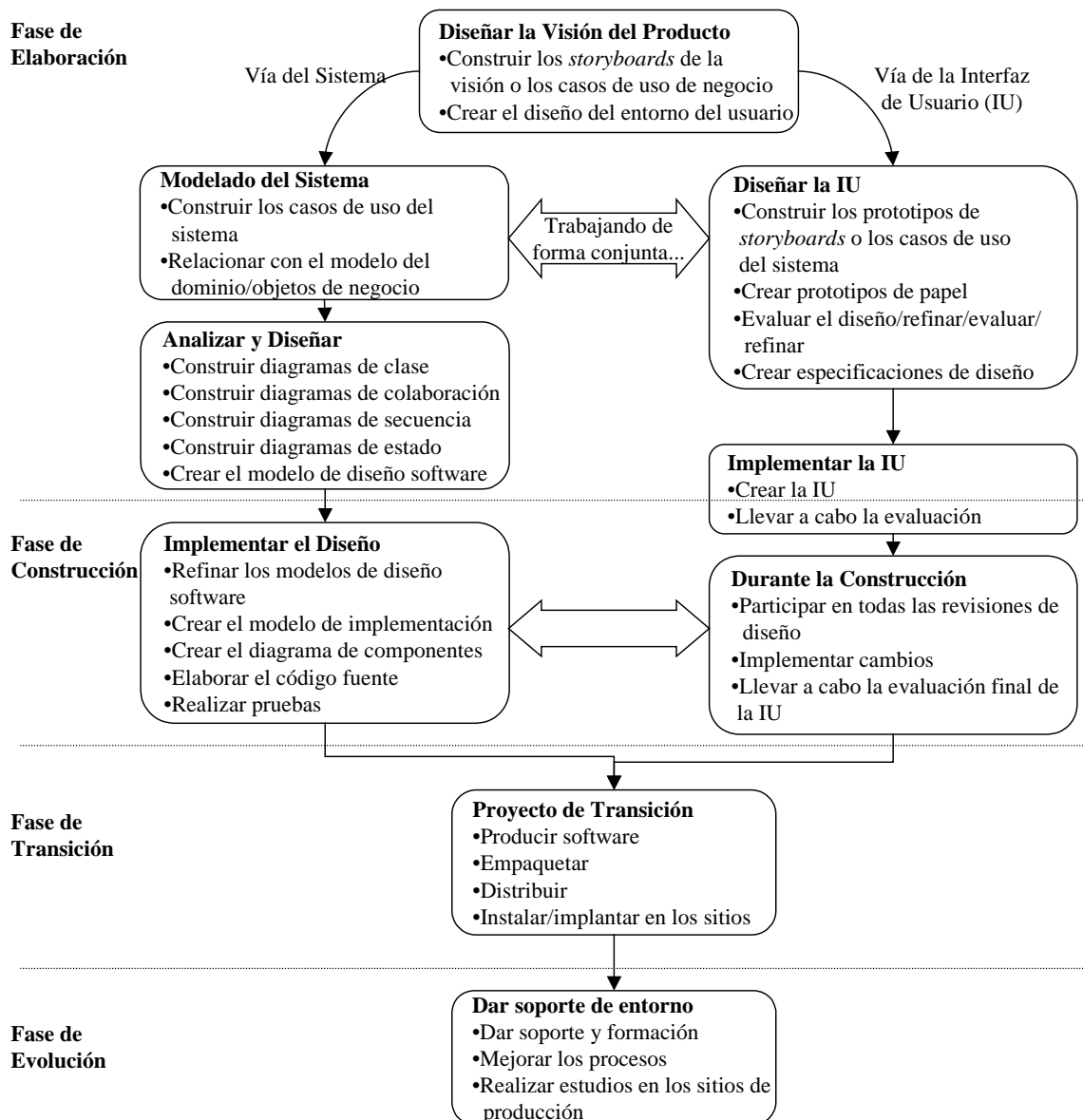


Figura 2.14 - Una Vista de Alto Nivel del Diseño y la Ejecución durante el Desarrollo Software

Como estrategia de incorporación del nuevo proceso en la compañía, primero lo emplearon en proyectos estratégicos clave, para más adelante introducirlo en equipos de desarrollo más establecidos.

Los autores se muestran de acuerdo con Cockburn en la apreciación de que los factores humanos prevalecen sobre cualquier otro factor en el desarrollo de software, y que el proceso de desarrollo debe considerar los factores humanos en el equipo de desarrollo, así como aquellos de los usuarios finales.

2.4.6.2 Análisis

Se trata de una solución ad-hoc que se ha creado para una organización de desarrollo de software. La propuesta está adaptada a las necesidades específicas de la organización para la que se ha desarrollado el proceso, puesto que abarca ampliamente temas de proceso de negocio junto con otros temas más técnicos. Por otra parte, la experiencia transmitida por los autores ilustra los grandes desafíos a los que cualquier organización de desarrollo de software puede tener que hacer frente, si decide incorporar prácticas de usabilidad en su proceso de desarrollo.

Los autores nos indican que la cultura organizacional necesita cambiarse para poder acomodar un enfoque centrado en el usuario en el desarrollo. Contaban con apoyo claro por parte de la dirección en el cometido de cambiar de un punto de vista centrado en el sistema a uno centrado en el usuario. Aún así, los autores han tenido que afrontar una serie de desafíos importantes en la introducción de la usabilidad en el nuevo proceso de desarrollo para toda la compañía. Todas estas dificultades revelan cuán lejos están los métodos y técnicas centrados en el usuario de las prácticas actuales de desarrollo de software.

La ubicación de técnicas de usabilidad junto con técnicas tradicionales en el marco de RUP resulta interesante por la relevancia actual de RUP en la IS, aunque lo conciso del artículo no permite llegar a los detalles de las actividades propuestas.

En cuanto a lo que los autores mencionan sobre la visión del producto, sorprende que los autores afirmen que Constantine y Lockwood llaman modelo contextual (*contextual model*) a la visión del producto, pues en [Constantine, 99] no aparece tal modelo. Se nombra el modelo de contexto de la interfaz (*interface context model*), pero únicamente cuando se trata de los modelos en los que se basa su propuesta. Puede ser que Anderson et al. se refieran al modelo del contenido o al mapa de navegación de contextos que sí son tratados con detalle en la obra de Constantine y Lockwood.

Consideramos que la representación del proceso propuesto no refleja un verdadero carácter iterativo. Da la impresión al observar las dos figuras que representan las actividades, que el componente iterativo es muy bajo, que la idea del ciclo de vida en cascada está detrás de la solución propuesta. En el texto se habla de la necesidad de iterar en cada ciclo, pero no parece estar reflejado en la linealidad de los pasos de requisitos y diseño. No se puede saber por lo que se dice en el artículo, pero uno puede pensar que, al ser la primera vez que trataban con RUP, en realidad están aplicando la visión errónea de RUP que consiste en superponer la cascada sobre las etapas del RUP (identificar inicio+elaboración con requisitos, elaboración +construcción con diseño, etc) [Larman, 02].

A pesar de que, probablemente, el proceso sí estará descrito con detalle en documentos internos de la compañía, en el artículo no se incluye suficiente información como para su aplicación por otra organización de desarrollo de software. En este sentido, lo consideramos poco detallado, pues nos basamos únicamente en la información de la que disponemos (el artículo).

En lo referente a la integración con las actividades tradicionales, en las etapas claves de definición de requisitos (ver Figura 2.13) prevalecen las actividades de usabilidad, de una forma similar a la concepción general de MUSE, entrando ya en las etapas de diseño (ver Figura 2.14) en la inclusión de actividades tradicionales como es la creación de modelos orientados a objetos. Las actividades de usabilidad son las que llevan el control del desarrollo en la fase de inicio, y en el comienzo de la fase de Elaboración, según sugieren tanto las actividades descritas en la figuras como el hecho de que el rol de ingeniero de usabilidad esté dedicado principalmente a requisitos. En este sentido, el grado de integración con las actividades tradicionales es pequeño, pues se reduce a generar los requisitos que serán la

entrada a la generación de modelos orientados a objetos. Con un cambio tan radical en cuanto a las prácticas desarrolladas en la empresa, no es de extrañar que los autores destaquen los problemas de comunicación y organizacionales encontrados en la tarea de integración que relatan en el artículo.

2.4.7 Wisdom

2.4.7.1 Descripción

Nunes presenta en su trabajo de tesis doctoral [Nunes, 01] el método Wisdom. Según el autor se trata de una propuesta de método de desarrollo de software ligero basado en UML y en extensiones al mismo para el modelado, que integra aspectos relevantes para el diseño de IUs y para el diseño centrado en el usuario, y que está especialmente adaptado para el desarrollo de aplicaciones software interactivas por parte de grupos pequeños de desarrollo. El proceso de Wisdom se basa en un modelo de arquitectura que incluye un conjunto de modelos basados en UML que dan soporte al desarrollo centrado en el usuario y al diseño de IUs. Así mismo, según el autor, la propuesta incluye una arquitectura a nivel de análisis que soporta la integración de elementos de la IU significativos desde el punto de vista arquitectónico.

El problema que aborda esta propuesta fue identificado en una serie de *workshops* llevados a cabo entre los años 97 al 99 (en CHI'97, CHI'98 y ECOOP'99), en los que el autor participa. En ellos se identificó la necesidad de utilizar el paradigma de la orientación a objetos para la integración de la IPO en la IS. Entre los temas tratados en estos *workshops*, se discutió la incorporación de análisis de tareas, de proceso y de trabajo en el desarrollo OO (Orientado a Objetos), y también la conveniencia de acercarse a la comunidad de modelado en UML, y de proponer un perfil UML para el desarrollo de sistemas interactivos. De estos workshop salió un esquema de integración para el uso de modelos OO para el diseño de la IU.

La premisa básica de la propuesta, según el autor, es el mismo argumento que llevó a la adopción de UML: Posibilitar la interoperabilidad de herramientas a nivel semántico y proporcionar un lenguaje común para especificar, visualizar y documentar sistemas intensivos en software. Esta premisa básica la aplica en impulsar la colaboración entre desarrolladores y diseñadores de la IU. En cuanto al proceso, la premisa en la que se basa Wisdom, según el autor, consiste en una estrategia de mejora que impulsa la exploración para asegurar la competitividad, pero que también permite un éxito repetible.

La Figura 2.15 ilustra el esquema de proceso Wisdom. Se presenta utilizando el tipo de esquema del Proceso Unificado: En la parte derecha de la figura se representan las cuatro fases del desarrollo (inicio, elaboración, construcción y transición), que en Wisdom se denominan fases evolutivas debido al carácter evolutivo que le caracteriza según su autor. En la parte izquierda están los flujos de trabajo (disciplinas en la terminología más reciente del Proceso Unificado [Kroll, 03]) de la IS, según el autor: Requisitos, Análisis y Diseño. El cuarto flujo de trabajo lo denomina "Evolución Aguasblancas (*Whitewater*)", y según el autor sustituye a la Implementación tradicional. Aguasblancas es una metáfora sacada del método Bridge, para ilustrar un proceso de desarrollo con gran semejanza a la intensa energía de los flujos rápidos de agua en los que la energía total (y aparentemente desordenada) hace que la actividad como un todo progrese muy rápidamente. La evolución, según el autor, complementa el concepto de aguasblancas porque el proceso evoluciona en una secuencia de prototipos incrementales que conducen, en último término, al producto final. Junto a los flujos de trabajo aparecen en la figura las actividades Wisdom, y la línea que hay junto a cada una representa el esfuerzo asociado a dicha actividad que usualmente se requiere según avanza el tiempo de desarrollo. En la parte inferior de la figura se representan las evoluciones. Según el autor, una evolución es un prototipo resultante de un cambio incremental realizado con la finalidad de conseguir un producto final deseado.

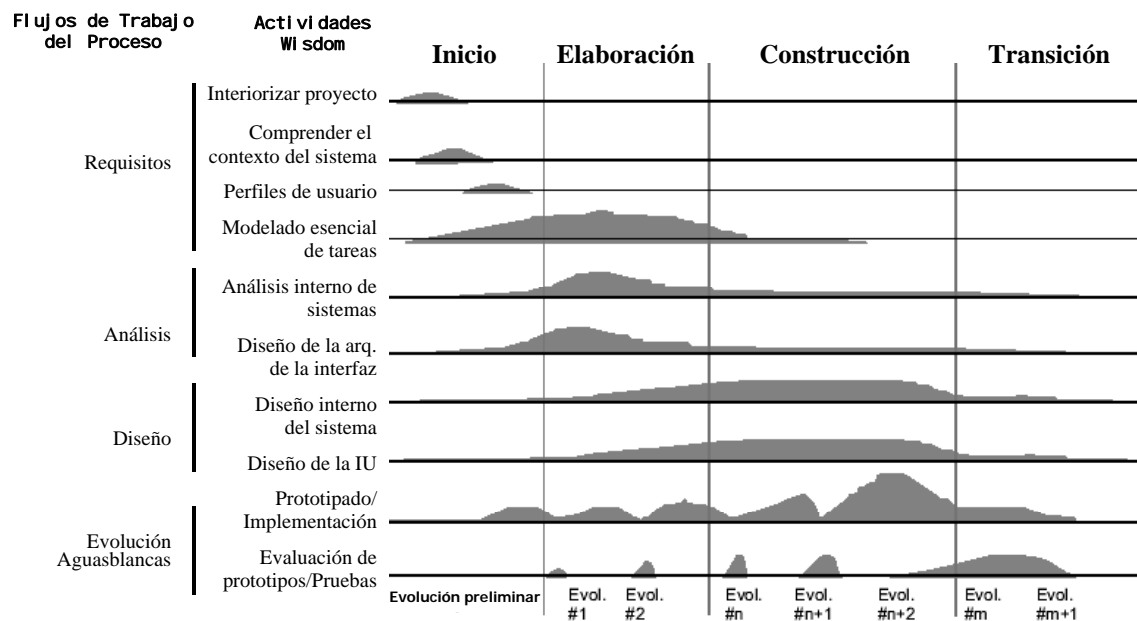


Figura 2.15- El Esquema de Proceso Wisdom

El autor presenta el esquema de proceso de Wisdom como especialmente apropiado para empresas pequeñas de desarrollo software con un modo de trabajo caótico. Según el autor, un modo de trabajo caótico es el habitual de este tipo de empresas, frente a organizaciones de mayor tamaño en las que encaja más un modelo de proceso tipo CMM (*Capability Maturity Model*).

La filosofía de Wisdom, según el autor, se basa en las siguientes características:

- Prototipado evolutivo.
- Del desarrollo caótico al prototipado evolutivo. El concepto de evolución aguasblancas es planteado como un enfoque de mejora de proceso para organizaciones que tienen un modo de desarrollo caótico.
- Dirigido por casos de uso esenciales y flujos de tareas. El autor parte de la constatación de que el modelado de casos de uso convencional no es adecuado para el desarrollo centrado en el usuario y para el diseño de la IU. Propone el uso de casos de uso esenciales, y de flujos de tareas para modelar los casos de uso de forma más acorde con el desarrollo centrado en el usuario.

El método Wisdom cuenta también con una arquitectura de modelos adecuada, según su autor, al desarrollo de sistemas software interactivos. Cuenta con los siguientes modelos: Modelo de roles de usuario, modelo de negocio o dominio, modelo de casos de uso, modelo de análisis, modelo de interacción, modelo de diseño, modelo de diálogo y modelo de presentación.

La propuesta incluye también una notación (notación Wisdom) que es una extensión a UML para poder modelar los aspectos relevantes del diseño de la IU, y del desarrollo centrado en el usuario en general.

2.4.7.2 Análisis

Wisdom se trata de una propuesta muy completa para la integración de la usabilidad en los aspectos de modelado, utilizando para tal fin extensiones a UML, el cual es el estándar de modelado en el desarrollo OO. Como tal, resulta de gran interés, puesto que el desarrollo OO es un paradigma con gran importancia en la actualidad en la IS. En el aspecto de proceso,

Wisdom se propone por su autor como adecuado para grupos de desarrollo de tamaño pequeño-medio y de complejidad baja-media, y además se propone como vía para empresas pequeñas con un modo de desarrollo caótico, para alcanzar un modelo de proceso más repetible, mediante la adopción de Wisdom. Por tanto, restringe bastante su aplicabilidad al proponerse únicamente para organizaciones que no tienen un proceso definido.

El método se propone como un conjunto completo de prácticas, que se adopta como tal. Por lo tanto es poco flexible para una organización con un proceso establecido que quiere acomodarlo para integrar las prácticas de usabilidad, sin abandonar completamente el proceso software actual debido a las ventajas que le proporciona. Requiere adoptar la filosofía de desarrollo en la parte de IS que se ha usado como base para el método, incluyendo las suposiciones implícitas acerca del desarrollo que incluye. Por ejemplo, el método Wisdom diferencia entre un flujo de trabajo de Análisis y uno de Diseño. Esta distinción está presente en métodos de desarrollo OO previos al Proceso Unificado. Sin embargo, en las interpretaciones más recientes del Proceso Unificado no se realiza tal distinción: Kroll y Kruchten consideran una única disciplina de "análisis y diseño" [Kroll, 03], mientras que Larman no considera una disciplina de análisis y afirma que el Modelo de Análisis del Proceso Unificado no es necesario y apenas se usa [Larman, 02]. Así, la propuesta no es tan atractiva para organizaciones con una filosofía de desarrollo distinta (incluso si utilizan el paradigma OO), como puede serlo para aquellas que sí desarrollen software con este enfoque en concreto.

La parte de la tesis más relevante para el presente trabajo de investigación, la referente al esquema de proceso Wisdom, no aparece en el capítulo VI, Experiencia Práctica y Resultados, el cual podemos considerar el capítulo de validación de los resultados del trabajo de tesis. En dicho capítulo únicamente se tratan los aspectos relacionados con herramientas CASE (*Computer-Aided Software Engineering*), con la generación automática de IUs independientes del dispositivo y de documentos AUIML (*Abstract User-Interface Markup Language*) a partir de modelos Wisdom, con el intercambio de herramientas para mejorar el soporte al modelado de tareas, y con el uso de la notación Wisdom para los patrones de IUs. Como se puede comprobar, todos estos temas están relacionados con los modelos propuestos en Wisdom, pero ninguno de ellos está relacionado con los aspectos de proceso. Esta carencia puede deberse a las dificultades que el autor nombra en el apartado de mejoras futuras. En concreto, se refiere a las dificultades encontradas para poder caracterizar adecuadamente la práctica diaria de los desarrolladores en Portugal, y a las dificultades para recoger datos relacionados con el proceso en las organizaciones que han adoptado el método Wisdom o alguna de sus contribuciones. En cuanto a esta última afirmación, sorprende que el autor no haya incluido más detalles acerca de las empresas que usan Wisdom, porque únicamente se incluye esta referencia vaga, cuando sería interesante contar con más información acerca de la aplicación práctica del método en su vertiente de proceso. Como indicador del mayor énfasis otorgado en esta propuesta a los temas de modelado frente a los de proceso, podemos citar al autor cuando asegura que: "Durante el presente programa de investigación nos hemos concentrado en las contribuciones técnicas que están en la base del método, pero hay grandes oportunidades para la mejora en gestión del proceso" [Nunes, 01].

El enfoque de desarrollo de Wisdom se basa en el del Proceso Unificado, resaltando la iteratividad con el concepto de evolución aguasblancas, por lo que consideramos que se trata de un enfoque verdaderamente iterativo. La propuesta se centra en integrar la usabilidad en procesos de desarrollo orientados a objetos, y, en concreto, con aquellos con un enfoque como el del Proceso Unificado, en el que se basa para describir las actividades del método reflejadas en la Figura 2.15. Por tanto, se trata de una integración detallada con un proceso en particular.

En cuanto al nivel de detalle de la propuesta, el esquema de proceso Wisdom incluye una descripción de las actividades detalladas de las que se compone. El método se presenta como un método de desarrollo de IS ligero para el desarrollo de sistemas software interactivos.

Junto a las actividades relacionadas con el diseño de la IU incluye el resto de actividades del desarrollo (ver Figura 2.15). Por tanto, consideramos que la propuesta abarca el desarrollo del sistema completo.

El método trata la adecuada comprensión del usuario y sus tareas mediante la inclusión de actividades que adoptan las prácticas IPO más relevantes al respecto, en vez del enfoque convencional de la IS, al cual critica. En lo referente al aspecto participativo, de involucración de usuarios en el desarrollo, el autor indica que en el método se combinan las narrativas de casos de uso con flujos de tareas elaborados de forma participativa, según propone el método Bridge. Consideramos que, aunque en la propuesta se pone énfasis en la parte técnica, la inclusión de referencias a las sesiones participatorias es suficiente para considerar que se promueve la participación de usuarios en el desarrollo. Por tanto, consideramos que el método se adhiere a un enfoque centrado en el usuario.

A pesar de las deficiencias identificadas en cuanto a su flexibilidad para poder acomodarse a un abanico amplio de organizaciones, el método Wisdom es muy interesante desde el punto de vista de la IS, por los modelos que ofrece para tratar con temas relacionados con la IU. Así, llena un vacío existente en dicho tema, y responde a una demanda real por parte de desarrolladores que quieren tratar las IUs con modelos de forma similar a otros aspectos del desarrollo.

2.4.8 Plug-in de Experiencia del Usuario para RUP

2.4.8.1 Descripción

El Proceso Unificado [Jacobson, 99] es el proceso que está recibiendo en la actualidad la mayor atención en la IS, puesto que está impulsado por los principales metodologistas del desarrollo orientado a objetos de la década de los 90: James Rumbaugh, Ivar Jacobson y Grady Booch. El Proceso Unificado adopta un enfoque verdaderamente iterativo, el cual se refleja en la denominación de disciplinas para los tipos de actividades que se realizan en el desarrollo, para así evitar la eventual identificación actividad-etapa que constituye la base del ciclo de vida en cascada. De los procesos definidos que tienen un verdadero enfoque iterativo, podemos afirmar que el Proceso Unificado es el que goza de un mayor grado de aplicación en proyectos reales. Aunque la adición del *plug-in* de Experiencia del Usuario no es comparable a las otras propuestas detalladas en secciones anteriores, puesto que representa una integración mucho menor de la usabilidad, se ha incluido en el presente estudio por el papel preponderante del que disfruta el Proceso Unificado entre los modelos de proceso de la IS.

RUP (*Rational Unified Process*) es una particularización del modelo de proceso representado por el Proceso Unificado, y es vendido por la empresa *Rational Software Corporation* como un producto. RUP cuenta con una serie de *plug-ins* que permiten desarrollar facetas específicas del desarrollo necesarias para dominios y tipos de proyectos concretos.

En la versión 2002 de RUP se incluyó un *plug-in* para modelar la experiencia del usuario (UX en sus siglas en inglés) [Rational, 02], basado parcialmente en el trabajo sobre modelado de UX de Jim Conallen.

Según Conallen, el término experiencia del usuario "se utiliza para describir al equipo y las actividades de aquellos especialistas responsables de mantener la IU consistente con los paradigmas actuales y, lo más importante, apropiados para el contexto en el cual se espera que el sistema va a ejecutarse" [Conallen, 03]. El equipo UX es responsable de crear la apariencia y estilo (*look-and-feel*) de la aplicación, determinando las rutas principales de navegación y organizando la estructura del contenido. Este cometido tiene parte arquitectónica y parte estética. La parte estética no se trata explícitamente en el trabajo de Conallen pues es

considerada por el autor un arte y no una ciencia. En cuanto a la parte arquitectural, los artefactos de los que es responsable el equipo de UX son:

- Pantallas y descripción de los contenidos
- Escenarios *Storyboard*
- Rutas de navegación entre pantallas.

Estos modelos o documentos forman el modelo UX. Este modelo captura la visión completa del sistema desde el punto de vista de las pantallas. Las propiedades arquitectónicas significativas y sus relaciones navegacionales son los elementos principales del modelo. Las realizaciones de casos de uso, o colaboraciones UML (*Unified Modeling Language*), capturan los escenarios *storyboard* de la aplicación en términos de las pantallas del sistema y los *storyboards* son propiamente relaciones con el modelo de casos de uso.

2.4.8.2 Análisis

El Proceso Unificado no considera directamente la usabilidad (carece de un enfoque de usabilidad y de soporte para el diseño centrado en el usuario según Gulliksen y Göransson [Gulliksen, 01]), pero tiene como una de sus características básicas el estar dirigido por casos de uso. El modelado de casos de uso tiene similitudes con la técnica IPO de análisis de tareas. Por tanto, podría servir de punto de partida para la integración de la usabilidad en el proceso de desarrollo. Sin embargo, el modelo de casos de uso del Proceso Unificado juega un papel secundario con respecto al diseño de la arquitectura del sistema. El modelo de casos de uso es muy importante en la planificación de la funcionalidad que se va a desarrollar en cada ciclo del desarrollo, pero una vez da comienzo un ciclo, los casos de uso se tratan como versiones preliminares de los elementos internos del diseño. Al etiquetar a los elementos de diseño como "realizaciones de casos de uso" se convierten los casos de uso en artefactos del mundo del diseño de la funcionalidad interna, y por tanto se alejan del entorno del usuario, perdiendo así la mayor parte de las ventajas de las que gozan desde un punto de vista centrado en el usuario.

Por otra parte, el modelado de la experiencia del usuario está basado en las actividades de desarrollo del concepto de un sitio web, que son habituales en el campo del desarrollo web, y que normalmente dirigen el proceso de desarrollo. El desarrollo del concepto del producto es una actividad que podemos considerar habitual en desarrollos centrados en el usuario, puesto que en la IPO se concede mucha importancia al desarrollo de modelos mentales del sistema adecuados [Norman, 90] [Preece, 94]. La definición dada por Conallen para la experiencia del usuario, donde se explicita como de gran importancia el contexto de uso al que está destinado el sistema, indica así mismo que se trata de una incorporación de conceptos típicos de la IPO al proceso de desarrollo general. Sin embargo, a pesar de lo prometedora que resulta esta definición desde una perspectiva IPO, el trabajo de Conallen se dedica únicamente al modelado, no trata acerca de la significación centrada en el usuario que las actividades de modelado deberían tener. Aún así, las técnicas de modelado que propone son técnicas IPO.

A pesar de constituir un avance hacia la integración de la usabilidad en el proceso software, este *plug-in* de RUP no cubre el proceso de desarrollo completo y se limita a unos pocos modelos. No se trata de una propuesta de integración de la usabilidad en el proceso de desarrollo completo, porque no trata de la adecuada comprensión del usuario y sus tareas, ni de la participación de usuarios en actividades de evaluación de la usabilidad, por ejemplo. Sin embargo, es una indicación del creciente interés que se puede observar en el campo del desarrollo para la web hacia la integración en el proceso de una perspectiva centrada en el usuario.

2.4.9 RUPi

2.4.9.1 Descripción

Soares y Furtado [Soares, 03a] [Soares, 03b] definen una extensión al RUP consistente en la adición de conceptos de IPO, que llaman RUPi, de forma análoga al UMLi (*Unified Modeling Language for Interactive Systems*).

Los conceptos de IPO incluidos según las autoras son:

- Factores humanos. Las autoras utilizan este término para referirse a los factores humanos medibles de Shneiderman [Shneiderman, 98]. Se trata de lo que otros autores IPO definen como atributos de usabilidad.
- Guías de diseño de la IU.
- IUs para todos. Diseñar las IU para la población más amplia posible de usuarios. Este objetivo puede alcanzarse, según las autoras, aplicando métodos de aceptabilidad, accesibilidad, usabilidad y de IUs internacionales e interculturales.
- Generación de IUs basada en modelos. Las características de la IU y sus funcionalidades son generados a partir de especificaciones representadas en modelos (modelo conceptual de la interfaz, modelo de tareas, modelo de usuarios, y modelo del contexto de uso).

La necesidad de extender RUP con estos conceptos de la IPO surge, según las autoras, por las siguientes deficiencias identificadas por ellas en RUP:

- No consideración de factores humanos durante el proceso de desarrollo software.
- No consideración de aspectos interculturales y de internacionalización durante el diseño de la IU.
- No consideración de modelos centrados en el usuario durante el diseño de la IU.
- No consideración de la personalización de la IU basada en la tecnología de interacción y en los usuarios.

Las seis buenas prácticas en las que se basa RUPi, según sus autoras, son las siguientes:

- Desarrollar software iterativamente (para poder aplicar aspectos IPO desde el principio del desarrollo)
- Gestionar requisitos (análisis de las tareas de los usuarios, del entorno que les rodea y de los dispositivos que utilizan)
- Usar una arquitectura basada en componentes (arquitectura de “n” capas)
- Modelar visualmente el software (modelos de usuarios, tareas, contexto y dominio)
- Verificar continuamente la calidad del software (RUPi se centra en usabilidad, accesibilidad y aceptabilidad)

- Controlar los cambios en el software (RUPi propone esfuerzos de mantenibilidad para los modelos definidos en la educación de requisitos)

RUPi se basa en las fases de RUP, pero con conceptos IPO en cada fase y evaluación de los mismos en cada hito. El añadido de conceptos IPO es para realizar actividades que producen artefactos que consideran factores humanos, requisitos de usabilidad, temas de accesibilidad y aceptabilidad, evaluación de usabilidad y escenarios gráficos. Estos últimos son generados para representar gráficamente situaciones específicas experimentadas por usuarios mientras interactúan con el sistema.

Respecto a los flujos de trabajo de RUP, RUPi incluye los siguientes cambios:

- Flujo de Requisitos: Añade el modelado de usuarios, de tareas (incluyendo la parte de modelado cognitivo de tareas) del contexto de uso, del dominio, y la definición de guías (no dice si de diseño o de qué) y de requisitos de usabilidad. Para el modelado de tareas propone usar el formalismo MAD, basado en una jerarquía de tareas. Para el modelado del contexto de uso plantea el uso de escenarios gráficos, útiles para anticipar riesgos y abordar temas de accesibilidad y aceptabilidad. Como artefactos nuevos propone el Modelo de usuarios y de tareas (este último orientado por el modelo de casos de uso).
- Flujo de Análisis y Diseño: Añade en este flujo el refinado del modelo de tareas desarrollado en el flujo anterior considerando características de distintos tipos de usuarios y diferentes estilos de interacción. Añade también el diseño conceptual de la IU (estructura navegacional del sistema). El diseñador debe asociar requisitos de usabilidad a las opciones de diseño. El diagrama de clases se define en una arquitectura de n-capas (*n-tier*).
- Flujo de Implementación: La generación del IU físico se mejora mediante la aplicación de guías.
- Flujo de Pruebas: Incluye evaluación de la usabilidad ligada a los requisitos de usabilidad previamente establecidos. Como artefactos nuevos incluye casos de prueba adicionales centrados en la usabilidad.

Quedan por definir en esta ampliación a RUP los posibles cambios a los cinco flujos restantes de dicho proceso.

2.4.9.2 Análisis

En primer lugar, en cuanto a los conceptos IPO que se pretende incluir en RUP, las autoras no detallan por qué han escogido estos conceptos y no otros, pues no se trata de conceptos ampliamente aceptados como definitorios por sí mismos de un enfoque IPO del desarrollo. Mezclan en esa selección de conceptos IPO cosas no comparables: Por ejemplo el "diseño de IUs para todos" es una necesidad para proyectos de desarrollo de software destinado a una población de usuarios caracterizada por una gran variedad, mientras que el uso de guías es una práctica concreta de diseño de IUs, una de las actividades a incluir en el proceso de desarrollo, no un aspecto de IPO de alto nivel. En resumen, el planteamiento resulta un tanto confuso, y está basado en premisas que no están justificadas en el artículo y que no son las comúnmente citadas por autores del campo de la IPO.

En lo referente a las deficiencias identificadas por las autoras en el RUP original, sorprende desde un punto de vista IPO la inclusión de la no consideración de aspectos de internacionalización e interculturales. Tales aspectos son relevantes únicamente en proyectos

específicos, dirigidos a un mercado mundial, lo que no es el caso de una gran cantidad de productos software que se desarrollan. En cuanto a la personalización de la IU, se trata de una posible alternativa de diseño, no una piedra básica de la IPO. Sin embargo, lo nombran al mismo nivel que el no considerar modelos centrados en el usuario.

La modificación propuesta a RUP mezcla la realización de actividades de modelado de usuarios, tareas y entorno con decisiones de diseño de la arquitectura. En concreto, para la arquitectura del sistema software, las autoras proponen la adopción de una arquitectura de *n*-capas (*n-tier*). La justificación ofrecida por las autoras para adoptar tal arquitectura específica es que así la información está más separada. Esto es, usan el principio de ocultación de la información para adoptar tal arquitectura, el cual se trata de un principio perteneciente a la IS. Debido a que las autoras presentan RUPi como un proceso de desarrollo que se basa en la integración de aspectos de IPO, sería de esperar que usaran un argumento de la IPO y no de la IS para seleccionar la arquitectura preferida. Por otra parte, la evaluación de usabilidad la basan exclusivamente en los requisitos de usabilidad, por tanto no tratan de la riqueza de técnicas de evaluación usabilidad existente en el campo de la IPO.

Las autoras defienden que los IUs generados con RUPi son más usables que los de RUP porque incluyen la posibilidad de la personalización, lo cual consideramos que es una afirmación muy discutible, pues la personalización por sí sola no aumenta la usabilidad del sistema si no se acompaña de una adecuada comprensión de los requisitos de los usuarios y sus tareas, por ejemplo, que sí es uno de los pilares básicos del desarrollo centrado en el usuario.

Por todas estas razones, no consideramos que las autoras prueben en el trabajo publicado que su ampliación a RUP convierte a éste a un enfoque verdaderamente centrado en el usuario.

En cuanto al grado de iteratividad, por la descripción dada de las fases de RUP, se puede llegar a la conclusión de que están superponiendo el ciclo de vida en cascada: “En la fase de inicio, el foco está en comprender requisitos genéricos, y en la definición del ámbito del proyecto. En la fase de elaboración, el foco está en los requisitos, aunque algunos esfuerzos se dedican a la producción de un prototipo de arquitectura [...]. En la fase de construcción el foco está el diseño y la implementación, [...]. En la fase de transición el foco está en asegurar que el sistema tiene el nivel correcto de calidad para alcanzar sus objetivos, [...]” [Soares, 03a]. Como se ha indicado más arriba, esta asignación de las etapas de un desarrollo iterativo a etapas del ciclo de vida en cascada, se trata del error más común que hace que no se lleven a cabo desarrollos verdaderamente iterativos [Larman, 02] [Kroll, 03].

RUPi se puede considerar que aborda el desarrollo del sistema completo, pues incluye RUP para los temas de desarrollo no directamente relacionados con la IU o con la usabilidad.

Las autoras no detallan las decisiones sobre las técnicas de usabilidad escogidas para integrar en RUP. La información publicada sobre RUPi no permite llegar a un nivel de detalle suficiente, por lo que categorizamos la propuesta como indicadora únicamente de actividades de alto nivel, ofreciendo, así mismo, tan solo indicaciones generales de cómo se integran los aspectos de usabilidad con el resto de prácticas del desarrollo.

2.4.10 Caracterización de Técnicas de Holzinger

2.4.10.1 Descripción

Holzinger [Holzinger, 05] propone una caracterización de técnicas de usabilidad (a las que denomina métodos de usabilidad) dirigida a desarrolladores software. El autor parte de la base de que los desarrolladores deben no sólo conocer las distintas técnicas de usabilidad, sino

también ser capaces de determinar con rapidez qué técnica es más apropiada para cada situación en un proyecto software.

La caracterización de técnicas propuesta por el autor se muestra en la Tabla 2.3. Como puede comprobarse en la tabla, divide las técnicas entre las de inspección (sin usuarios finales) y las de pruebas (con usuarios finales).

Tabla 2.3 - Comparación de Técnicas de Evaluación de Usabilidad

	Técnicas de Inspección			Técnicas de Pruebas		
	Evaluación Heurística	Recorrido Cognitivo	Análisis de Acción	Pensar en Voz Alta	Observación de Campo	Cuestionarios
Aplicable en Fase	todas	todas	diseño	diseño	pruebas finales	todas
Tiempo Requerido	bajo	medio	alto	alto	medio	bajo
Usuarios Necesarios	ninguno	ninguno	ninguno	3+	20+	30+
Evaluadores Necesarios	3+	3+	1-2	1	1+	1
Equipo Requerido	bajo	bajo	bajo	alto	medio	bajo
Experiencia Requerida	media	alta	alta	media	alta	baja
Intrusiva	no	no	no	sí	sí	no

Cada una de las técnicas caracterizadas es introducida brevemente. Para tres de las técnicas se cita una referencia (Evaluación Heurística, Recorrido Cognitivo y Pensar en Voz Alta), mientras que para las tres restantes no se incluye tal información (Análisis de Acción, Observación de Campo y Cuestionarios).

2.4.10.2 Análisis

El autor presenta las técnicas que incluye en su caracterización como los métodos de usabilidad que se emplean para conseguir que un proyecto software⁴ tenga las cinco características esenciales de usabilidad: Facilidad de aprendizaje, eficiencia, capacidad de recuerdo, baja tasa de errores, y satisfacción. Únicamente incluye técnicas de evaluación de la usabilidad, lo cual sorprende sobremanera, porque existen muchas más técnicas empleadas en el campo de la IPO para conseguir un buen nivel en los atributos de usabilidad. Por otra parte, el autor se contradice con esta decisión, pues él mismo indica al principio del artículo que la usabilidad debe considerarse antes de que se lleve a cabo el prototipado. Todas las técnicas incluidas en su descripción, por el contrario, requieren un prototipo o un sistema funcionando para poder aplicarse. Lamentablemente, el autor no da detalles acerca de los criterios que ha seguido para seleccionar las técnicas de usabilidad a incluir. Respecto a las técnicas seleccionadas, también sorprende que el autor diga, como colofón del artículo, que "una necesidad absoluta es la comprensión de las tareas, cultura y capacidades del usuario; implicar a los usuarios pronto en el diseño; y probar e iterar, con o sin usuarios". Sin embargo, ninguna

⁴ Suponemos que se refiere a un producto software en vez de a un proyecto software.

de las técnicas incluidas en su caracterización sirve directamente al propósito de comprender las tareas, cultura y capacidades del usuario.

Por otra parte, la técnica de "Análisis de Acción" no es una técnica habitual en la IPO. El autor no incluye ninguna referencia que explique tal técnica, por lo que su inclusión en el marco de caracterización no resulta de gran utilidad a los desarrolladores que acuden a este trabajo en busca de información sobre qué técnicas de usabilidad aplicar.

Los valores asignados a cada criterio para cada técnica aparecen en un esquema como la Tabla 2.3, pero no están siempre argumentados en el texto del artículo. Algunos de los valores resultan extraños. Por ejemplo, para la técnica Pensar en Voz Alta, sorprende que considere que el tiempo requerido es alto, frente a un tiempo medio para la Observación de Campo. El autor indica que esta técnica lleva cierto tiempo de aplicar porque es preciso dar instrucciones a los usuarios acerca de cómo realizarla. Para la técnica de Observación de Campo indica que debe llevarse a cabo de forma no obtrusiva. Dicha técnica suele requerir mucho más tiempo de aplicación que la de Pensar en Voz Alta precisamente porque no se quiere molestar al usuario. Así, precisa un tiempo de observación mucho más alto, que sin duda supera el tiempo que se emplea en dar las instrucciones a un participante en una prueba de Pensar en Voz Alta. Otro valor que sorprende para esta técnica es el de la necesidad alta de equipamiento. La aplicación más simple de la técnica de Pensar en Voz Alta no necesita ningún tipo de equipamiento especial, puesto que no es preciso grabar en vídeo la sesión ni registrar tiempos (al ser incompatible con una Medición del Rendimiento por la ralentización que implica para el participante el esfuerzo de verbalizar sus acciones).

Por otra parte, no indica ninguna técnica que favorezca la participación de usuarios en el diseño (al estar todas las técnicas relacionadas con la evaluación), por lo que no podemos considerar que la propuesta se adhiera completamente a un enfoque centrado en el usuario.

El criterio "Aplicabilidad en Fase" es el más relevante para el objetivo de integración de la usabilidad en el proceso de desarrollo. El autor no detalla los posibles valores para cada criterio, pero por los valores que aparecen en la Tabla 2.3 podemos comprobar que responden a un enfoque de ciclo de vida en cascada, puesto que las fases del desarrollo coinciden con tipos de actividades (diseño y pruebas). En un desarrollo iterativo el diseño no es una fase del desarrollo.

Debido a la poca extensión del trabajo (cuatro páginas), a la falta de una argumentación convincente y a la falta de referencias que permitan ampliar la escasa información detallada en el artículo, consideramos que este trabajo resulta de escasa utilidad para el objetivo de integración de las prácticas de usabilidad en el proceso de desarrollo, aunque sí puede servir al propósito de introducir técnicas de evaluación de usabilidad a una audiencia con escasos conocimientos en la materia.

2.5 Resumen de la Revisión del Estado de la Cuestión

Como resumen de la revisión del estado de la cuestión presentada en este capítulo, la Tabla 2.4 recoge las principales características de las propuestas de integración de la usabilidad en el proceso de desarrollo software revisadas, según los criterios descritos en la sección 2.1. Los valores reflejados son los mencionados para cada propuesta en la sección de análisis.

- La columna *Grado de Iteratividad* detalla si la propuesta es verdaderamente iterativa o si, por el contrario, lleva implícito un enfoque del desarrollo basado en el ciclo de vida en cascada.

Tabla 2.4 - Resumen de las Propuestas Revisadas en el Estado de la Cuestión

		GRADO DE ITERATIVIDAD	NIVEL DE DESCRIPCIÓN DEL PROCESO	ALCANCE DEL PROCESO PROPUESTO	GRADO DE INTEGRACIÓN DE ASPECTOS DE USABILIDAD	ADHESIÓN A UN ENFOQUE CENTRADO EN EL USUARIO
Estándar IS	Enmienda 1 al Estándar ISO 12207	N.A.	Actividades de Alto Nivel	Sistema Completo	La integración debe ser planificada	Centrado en el usuario
Métodos IPO	Ciclo de vida en estrella	Iterativo	Actividades de alto nivel	Diseño de la IU	Indicaciones generales	Centrado en el usuario
	Estándar ISO 13407	Iterativo	Actividades de alto nivel	Actividades de diseño centrado en el humano	La integración debe ser planificada	Centrado en el usuario
	Ciclo de vida de la ingeniería de usabilidad	Nominalmente iterativo	Actividades detalladas	Diseño de la IU	Relación con actividades detalladas (método OOSE y métodos de prototipado rápido)	Centrado en el usuario
	Diseño centrado en el uso	Iterativo	Actividades de alto nivel	Sistema completo	Indicaciones generales	Centrado en el usuario
	ISO/TR 18529	Iterativo	Actividades de alto nivel	Actividades de diseño centrado en el humano	Indicaciones generales	Centrado en el usuario
Extensión a Métodos IS	Propuesta de Coutaz	Nominalmente iterativo	Actividades de alto nivel	Evaluación	Indicaciones generales	Parcialmente centrado en el usuario
	MUSE	Nominalmente iterativo	Actividades detalladas	Diseño de la parte interactiva del sistema	Relación con actividades detalladas (JSD)	Centrado en el usuario
	Propuesta de Hakiel	Nominalmente iterativo	Actividades de alto nivel	Sistema completo	Relación con actividades de alto nivel	Parcialmente centrado en el usuario
	LUCID	Nominalmente iterativo	Actividades de alto nivel	Sistema completo	La integración debe ser planificada	Centrado en el usuario
	Propuesta de Costabile	Nominalmente iterativo	Actividades de alto nivel	Sistema completo	Relación con actividades de alto nivel (ciclo de vida en cascada)	Centrado en el usuario
	Propuesta ad-hoc de Anderson et al.	Nominalmente iterativo	Actividades de alto nivel	Sistema completo	Productos de usabilidad se alimentan como entradas a actividades	Centrado en el usuario
	Wisdom	Iterativo	Actividades detalladas	Sistema completo	Relación con actividades detalladas (Proceso Unificado)	Centrado en el usuario
	Plug-in de experiencia del usuario para RUP	Iterativo	Actividades detalladas	Diseño de la IU	Relación con actividades detalladas ⁵ (RUP)	Parcialmente centrado en el usuario
	RUPi	Nominalmente iterativo	Actividades de alto nivel	Sistema completo	Indicaciones generales	Parcialmente centrado en el usuario
Caracterización de Técnicas de Holzinger	Nominalmente iterativo	Actividades de alto nivel	Evaluación	Indicaciones generales	Parcialmente centrado en el usuario	
VALORES IDEALES	Iterativo	Actividades detalladas	Sistema completo	Detallada para un abanico amplio de proyectos	Centrado en el usuario	

⁵ Se trata de un proceso de la IS con un pequeño añadido de usabilidad.

- La columna *Nivel de Descripción del Proceso* detalla si el proceso describe tan solo actividades o etapas de alto nivel, junto con sus dependencias o si, por el contrario, describe actividades de grano fino, sus dependencias a ese nivel, junto con las técnicas a aplicar en cada actividad y los productos producidos.
- La columna *Alcance del Proceso Propuesto* se refiere al objetivo de cada método según lo indican sus autores, en cuanto a si aspira a cubrir el desarrollo del sistema interactivo completo o, por el contrario, se presenta como un método únicamente para una parte del desarrollo, como puede ser el diseño de la IU.
- La columna de *Grado de Integración de Aspectos de Usabilidad* recoge cómo la propuesta trata con la integración de las actividades y técnicas de usabilidad con las del proceso de desarrollo general.
- La columna de *Adhesión a un Enfoque Centrado en el Usuario* indica qué propuestas pueden considerarse como acordes con el enfoque de desarrollo de usabilidad abogado en el campo de la IPO, y son, por tanto, centradas en el usuario (excluyendo la condición de tener un enfoque iterativo, que tiene su propia columna en la tabla).
- En la última fila de la tabla se muestran los valores considerados ideales para cada criterio.

Los estándares de proceso de desarrollo IEEE 1074 e ISO/IEC 12207 no se han incluido en la tabla, puesto que no tratan el problema de la integración de la usabilidad en el proceso de desarrollo. Únicamente se han revisado con el objetivo de conocer cómo tratan la usabilidad en su contenido normativo. A este respecto, podemos afirmar que ninguno de ellos imposibilita la integración de técnicas y actividades de usabilidad en un proceso de desarrollo definido según las restricciones impuestas por cada estándar. Sin embargo, sí podemos observar una mayor dificultad para realizar tal integración en un proceso acorde con el estándar IEEE 1074, que en un proceso definido acorde con el estándar ISO/IEC 12207, debido a la menor rigidez del segundo. En la primera enmienda al estándar ISO/IEC 12207, por el contrario, sí que se tienen en cuenta explícitamente la usabilidad y las actividades que la tratan, por lo que se ha incluido en la tabla resumen del estado de la cuestión. El criterio de iteratividad no es aplicable a este caso, puesto que no se prescribe en el estándar ningún ciclo de vida particular. Esta circunstancia se ha indicado en la tabla con el valor N.A. (No Aplicable).

Según se puede observar en la Tabla 2.4, entre las propuestas que se basan en descripciones de métodos únicamente el ciclo de vida de la ingeniería de usabilidad está descrito de forma más detallada, aunque carece de un verdadero carácter iterativo, y las indicaciones que ofrece para la integración en el proceso de desarrollo más amplio están ligadas al método OOSE o a métodos de prototipado rápido, limitando así la aplicabilidad de la solución propuesta en cuanto a integración. El resto de propuestas del campo de la IPO son propuestas con un verdadero carácter iterativo, pero adolecen de falta de detalle en cuanto a descripción del proceso y de su integración en el proceso de desarrollo general.

En las propuestas de integración propiamente dichas, nos encontramos con una falta de iteratividad real, excepto en los casos del *plug-in* de experiencia del usuario para RUP y el método Wisdom. Ambos casos se tratan de una adición de usabilidad a UP/RUP, limitándose así su aplicabilidad general.

Se puede observar también una mayor tendencia en las propuestas de la IPO a centrarse únicamente en el diseño de la IU, mientras que la mayor parte de las propuestas de extensión a métodos IS adoptan un enfoque más amplio y, por tanto, abordan la integración desde la perspectiva del diseño del sistema completo. En este sentido, el diseño centrado en el uso, a pesar de poder ser considerado como una propuesta desde la IPO, es la más cercana a la IS por su enfoque más amplio del desarrollo, sin centrarse únicamente en el desarrollo de la IU.

Como conclusión de la información mostrada en la tabla, podemos afirmar que **ninguna de las propuestas de integración de la usabilidad en el proceso de desarrollo software reúne los valores que consideramos ideales para los cinco criterios considerados:**

- **Que sea verdaderamente iterativa.**
- **Que describa de forma detallada las actividades y técnicas que se aplican en las mismas.**
- **Que abarque el sistema completo.**
- **Que relacione los aspectos de usabilidad con actividades detalladas de un abanico amplio de proyectos.**
- **Que se adhiera completamente a un enfoque centrado en el usuario.**

Además de estos cinco criterios, considerados básicos para la elaboración de la propuesta realizada en el presente trabajo de investigación, se han identificado otras carencias adicionales en algunas de las aproximaciones consideradas, que también se tendrán en consideración, aunque de forma secundaria. Estas carencias son las siguientes:

- *Basarse en un enfoque particular del desarrollo.* La propuesta MUSE se basa en una concepción del desarrollo transformacional, del tipo entrada-proceso-salida, donde cada actividad del proceso "transforma" la entrada en una salida, la cual no encaja fácilmente con un enfoque iterativo de desarrollo. Por otra parte, el Ciclo de Vida de la Ingeniería de Usabilidad se basa en un enfoque de diseño de descomposición (*top-down*). Así, para poder utilizar una propuesta con un enfoque concreto, la organización debe cambiar sus prácticas de desarrollo para acomodarse a dicho enfoque.
- *Presentar la selección de técnicas a aplicar sin razonar cómo se ha realizado dicha selección.* La mayor parte de propuestas se basan en el principio de autoridad, según el cual los autores fijan con su experiencia de desarrollo previa aquellas prácticas que mejor pueden servir al objetivo de integración de la usabilidad en el proceso de desarrollo. El ingeniero software debe confiar en el prestigio de los autores para considerar la bondad de su propuesta, lo cual no resulta fácil por el desconocimiento que puede tener del campo de la IPO.
- *Basar la propuesta para los casos en los que la construcción de software comienza una vez el producto a construir ha sido identificado, definido y delimitado.* Así, una parte clave del éxito del desarrollo de un producto software queda fuera del ámbito de la propuesta de integración. La IPO ofrece prácticas de usabilidad que pueden resultar de utilidad para esta tarea inicial, por lo que consideramos que debe incluirse en cualquier propuesta de integración que aspire a un cierto nivel de generalidad.

- Un caso particular de este supuesto consiste en *propuestas que están centradas en la automatización de tareas*. En estos casos el objetivo del proyecto está perfectamente delimitado, puesto que se basa en observar cómo se hacen las cosas previamente a la implantación del sistema software, y se estudia cómo automatizar mejor los procesos ya existentes observados. Conforme el desarrollo de software ha ido abriendo nuevas posibilidades, ha ido decreciendo el porcentaje de proyectos que se basan en la automatización, por lo que una propuesta basada en un enfoque de este tipo tampoco puede considerarse como aplicable a un abanico amplio de proyectos.
- *Describir la integración a nivel de actividades pero sin nombrar las técnicas concretas a integrar en cada actividad*. Estas propuestas se centran en la exposición de un planteamiento general, pero no llegan a indicar qué técnicas IPO concretas pueden aplicarse, por lo que su función como guía para un ingeniero software planteando la integración en el proceso resulta bastante limitada.
- *No estar dirigida a una audiencia de ingenieros software*. Algunas de las propuestas requieren de un conocimiento amplio de la IPO para poder comprenderse y, por tanto, no son directamente aplicables por ingenieros software. Puesto que nuestro estudio se centra en el caso de una organización con escasa o nula experiencia previa en IPO, consideramos esta situación como una carencia.

Capítulo 3

Planteamiento del Problema

3.1 Introducción

Una vez realizado el estudio del estado de la cuestión, podemos ya establecer con detalle en este capítulo cómo se va a enfocar el problema, qué hipótesis de trabajo se van a considerar, para qué circunstancias será válida la solución, cómo se plantea ésta, y cómo se va a aplicar para estudiar los resultados que se obtengan.

3.2 Definición del problema

La usabilidad ha cobrado un papel relevante en el mundo del desarrollo de software, pasando de ser un objetivo de segunda clase, que se consideraba en momentos avanzados del desarrollo, a ser un objetivo relevante para las organizaciones que construyen software. Se trata de un atributo de calidad que tiene un impacto decisivo en la satisfacción de los clientes y que un creciente número de organizaciones quieren tratar de forma adecuada.

Por otra parte, existen en el campo de la IPO métodos y técnicas cuya finalidad principal es la obtención de software usable. Estas técnicas IPO se vienen aplicando con éxito para la creación de software usable. Sin embargo, se aplican en el marco de métodos IPO, y no en el proceso de desarrollo habitual. Por otra parte, el proceso de desarrollo sigue, en la mayor parte de organizaciones, los principios de la IS, y se basa, principalmente, en la aplicación de técnicas IS. Excepto en aquellas organizaciones donde se da una relevancia predominante a la IPO, las técnicas IPO tienen un papel secundario al no estar integradas convenientemente en el proceso de desarrollo habitual.

La construcción de software según los métodos y prácticas de la IS no trata de forma adecuada la usabilidad a lo largo del proceso de desarrollo. Así, una organización que quiera construir software usable y que tenga un proceso definido, tiene que realizar un esfuerzo de integración de las prácticas IPO en su proceso. Debido a que dichas prácticas no están encajadas con las prácticas IS, tal integración no resulta trivial y acaba por suponer un esfuerzo considerable. Podemos resumir el problema a resolver en que **se necesitan estrategias que proporcionen a los ingenieros software descripciones de las técnicas y actividades IPO que pueden formar parte de su proceso de desarrollo**. Las principales dificultades para afrontar dicho problema son las siguientes:

- Los procesos de desarrollo del campo de la IPO no están descritos en términos del proceso de desarrollo de la IS. Ambos procesos están desacoplados, puesto que sus puntos de convergencia no están claramente definidos.
- Existen importantes diferencias de terminología entre la IPO y la IS, lo que dificulta el uso de la literatura IPO como fuente de información para el objetivo de integrar la usabilidad, pues requiere un importante esfuerzo de interpretación desde una perspectiva de IS.
- Es necesario hacer comprender a los desarrolladores por qué es necesario tratar la usabilidad desde los momentos tempranos del desarrollo, como defienden los expertos IPO.

En las secciones siguientes se detalla la problemática de cada uno de estos puntos.

3.2.1 Diferencias entre Proceso IPO y Proceso IS

Los procesos de desarrollo del campo de la IPO (denominados procesos centrados en el usuario o humano) no tienen el nivel de detalle habitual en los procesos IS. El término "proceso IPO" puede llevar a engaño, puesto que no existe consenso en IPO sobre actividades y técnicas, existiendo una gran variedad de actividades y técnicas a aplicar en cada actividad entre los distintos autores de la IPO, así como diferencias apreciables en el nivel de detalle con el que se describe el proceso. Algunos autores no definen explícitamente la actividad del proceso en la que aplicar cada técnica, puesto que la visión de proceso de la IPO es, de forma general, más laxa en este sentido que en la IS.

Los procesos IPO no son procesos completos que se ocupen de todos los aspectos del desarrollo. Así, se admite desde el campo de la IPO que el proceso de usabilidad debe integrarse en el proceso de desarrollo general. En algunos casos, en la literatura IPO se dan indicaciones de cómo encajar un método IPO concreto con un método o proceso de desarrollo específico de la IS. En esos casos, la información es válida únicamente para las organizaciones cuyo proceso consolidado coincida con el proceso para el que se dan las indicaciones. Pero, incluso así, los autores IPO tienden a adoptar una perspectiva en la cual el trato con el usuario y/o cliente se realiza en las actividades IPO y la parte que corresponde al proceso de desarrollo de la IS es la de realizar un diseño que se acomode a las especificaciones de cómo debe ser el sistema elaboradas en las actividades IPO.

La visión de dos procesos separados provoca el siguiente conflicto de intereses: La definición de qué software hay que construir (frente a cómo construirlo), lo que constituye el área de los requisitos, es una arena que ambas disciplinas consideran como su ámbito propio. Dada la importancia de esta tarea no es de esperar que ninguna de las disciplinas renuncie a tratar los requisitos. Así pues, la única alternativa viable es la integración de ambas perspectivas. Por tanto, es preciso ofrecer una información mucho más detallada de cómo incorporar técnicas IPO al desarrollo IS, frente a un enfoque simple en el que el producto que se produce como salida de una actividad constituye la entrada a otra actividad (por ejemplo, la salida de una actividad IPO alimentando a una actividad IS). Es decir, no es el único problema la falta de correspondencia entre proceso IPO y proceso IS. Incluso cuando, desde la literatura IPO, se aporta una clarificación sobre cómo integrar con métodos IS concretos, la solución carece de una perspectiva adecuada de la IS, pues se trata de una visión deformadora del ámbito propio de la IS, ya que reduce sensiblemente (si no elimina) todas las aportaciones provenientes de la ingeniería de requisitos.

Para afrontar la integración del proceso IPO en el proceso general de la IS, es preciso expresar el proceso IPO y su enfoque según los esquemas de proceso de la IS. En concreto, para los

temas comunes, es preciso indicar qué actividades ya existentes de la IS deben ser modificadas para acomodar técnicas IPO. Todo ello debe expresarse según una denominación suficientemente genérica de las actividades del proceso, que permita integrar las técnicas IPO en un abanico amplio de procesos IS.

3.2.2 Diferencias de Terminología entre IPO e IS

La aplicación de técnicas IPO permite tratar la usabilidad adecuadamente, pero su mejor ubicación en un proceso de desarrollo consolidado resulta una tarea compleja para un ingeniero software, debido a las diferencias de enfoque y terminología entre las disciplinas de la IPO y la IS.

En cuanto a enfoque, la IPO trabaja con una perspectiva centrada en el usuario, la cual permea todas las actividades de los procesos IPO. Esta preocupación por el usuario, el empeño en que su punto de vista sea considerado en todas las actividades del desarrollo, es preciso imbuirlo en ciertas actividades del proceso IS, pues éstas se verían enriquecidas cumpliendo más satisfactoriamente su objetivo.

Por otro lado, la terminología de proceso de desarrollo también difiere. La diferencia entre diseño de invención (el que tiene como fin conceptualizar y especificar software para satisfacer unas necesidades identificadas) y diseño de descomposición (el que trata la descomposición del software en partes y la definición de las interfaces entre las mismas) no se aborda del mismo modo en la IS y en la IPO. Mientras que en la IS el diseño de invención se considera parte del ámbito de los requisitos [SWEBOOK, 04], en la IPO no se considera tal distinción nítidamente. Así, nos encontramos con definiciones de métodos IPO que únicamente se describen como "procesos de diseño", llevando a error al ingeniero software.

La idea de la usabilidad como un atributo de calidad del cual se ocupa únicamente la actividad de Diseño de la IU contribuye a tal confusión terminológica. Gran parte de los procesos IPO describen como su objetivo el "diseño" de la IU; así, un ingeniero software interpreta a menudo que las técnicas que se incluyen en dicho proceso únicamente se aplican al diseño de la parte visual de la IU y su comportamiento asociado. Por el contrario, en dichos procesos IPO de "diseño" se incluye una parte muy importante de actividades de conceptualización, de diseño de invención.

El mismo término "diseño de la IU" incluye en la terminología IPO el diseño de la interacción, esto es, el diseño de cómo se va a coordinar el intercambio de información entre el usuario y el sistema, incluyendo cómo se van a comportar las funcionalidades del sistema desde el exterior. Esta definición va más allá de la mera apariencia y comportamiento de los elementos visuales de la IU, para abarcar parte de las decisiones de diseño de la parte interna del sistema.

Resumiendo, términos tan relevantes como diseño e IU no tienen la misma semántica en IPO que en IS. Esto mismo ocurre con muchos otros términos secundarios.

Debido a estas diferencias terminológicas, es preciso encajar las técnicas IPO en un marco establecido según la terminología IS, de modo que los desarrolladores comprendan su ubicación en el proceso. No obstante, en algunos casos no existe una terminología propia de IS para describir la finalidad de ciertas técnicas IPO, y no se desea migrar términos IPO que pueden resultar ambiguos para un ingeniero software. En estos casos será necesario acuñar nuevos términos, en consonancia con el estilo de la IS, de tal forma que quede claro para el desarrollador que se trata de elementos de nueva creación en el proceso.

3.2.3 Cuándo Tratar la Usabilidad en el Desarrollo

Como se ha destacado en la sección anterior, en ocasiones los ingenieros software identifican el tratamiento de la usabilidad con una actividad concreta del desarrollo, denominada Diseño de la IU. En dicha actividad se diseña la apariencia de los elementos visuales de la IU y su comportamiento. Los responsables de dicha actividad en su parte gráfica son expertos en diseño gráfico y se rigen por criterios principalmente estéticos. Así, el trabajo de Diseño de la IU pertenece a una disciplina relacionada (pero ajena), y tiene un alto grado de independencia con respecto al trabajo de los ingenieros software. Además, la actividad de Diseño de la IU se suele incluir en el desarrollo en los momentos finales, una vez se ha realizado el diseño de la funcionalidad del sistema y por tanto su componente interna está prácticamente finalizada.

Esta apreciación es errónea, puesto que la usabilidad no puede tratarse a posteriori, una vez que las decisiones principales acerca del funcionamiento del sistema han sido tomadas. La parte visual de la IU es únicamente una parte de los elementos que afectan a la usabilidad, incluso se trata de una parte relativamente menor comparada con el concepto del sistema, la forma en la que interaccione con el usuario, la secuencia de pasos que el usuario realizará, la forma de acceso a la información, etc.

La IPO considera como objeto de su trabajo el conjunto que engloba tanto al ser humano como al sistema informático, realizando juntos una serie de tareas cuya ejecución únicamente por parte del ser humano requeriría de éste un esfuerzo considerable o de imposible consecución. Así, el supersistema considerado (humano + sistema informático) tiene un número mayor de variables a controlar que en el caso de centrarse únicamente en el sistema informático. Tal control requiere de un esfuerzo constante para asegurar que el desarrollo no se desvíe de las necesidades del usuario. El esfuerzo constante implica la realización de actividades IPO a lo largo de todo el proceso de desarrollo.

La identificación de la necesidad de aplicar técnicas IPO a lo largo de todo el desarrollo se puede mostrar realizando una asignación de tales técnicas a tipos de actividades y a momentos en el desarrollo (según una visión de proceso de la IS), de tal forma que se pueda observar que la distribución de técnicas IPO cubre un amplio espectro de tipos de actividades y momentos.

3.3 Hipótesis de Trabajo

El problema que se desea resolver nos lleva a plantear la siguiente hipótesis:

H1. Es posible mejorar la situación actual de aplicación de técnicas IPO en los desarrollos no centrados en el usuario⁶ mediante la integración de técnicas IPO en el proceso de desarrollo IS.

Esta hipótesis general se puede descomponer en las siguientes subhipótesis:

SH1. Los desarrolladores valorarán más el aporte que el tratamiento de la usabilidad realiza a sus tareas del desarrollo si aplican nuestra solución que si aplican el modus operandi habitual⁷.

⁶ Como veremos en la sección siguiente, y ya se ha discutido en el capítulo de Introducción, la solución que aquí se propone está dirigida principalmente a organizaciones que no siguen procesos IPO y que tienen escasa o nula experiencia previa en aplicación de técnicas IPO.

- SH2. La inseguridad e indecisión de los desarrolladores sobre cuándo considerar la usabilidad durante la construcción de software disminuirá con respecto a su modus operandi habitual.
- SH3. El tratamiento de la usabilidad permeará más el desarrollo de software y afectará a más actividades que con su modus operandi habitual.
- SH4. El producto obtenido desarrollando según nuestra solución será más usable que con su modus operandi habitual.

3.4 Condiciones de Aplicación de la Solución

La solución que se plantea en el presente trabajo de investigación es aplicable sobre la base de ciertas condiciones que a continuación se describen.

Consideramos que la organización en la que se piensa utilizar la solución propuesto ya ha identificado como deseable la incorporación de la usabilidad en su proceso de desarrollo. **No está entre los objetivos del marco de integración la inclusión de actividades cuya finalidad sea la justificación de que los beneficios asociados al tratamiento de la usabilidad (mediante la aplicación de técnicas IPO) son superiores al coste de dicho tratamiento.** Esto es, que la solución se presenta como un aporte que será de utilidad principalmente a organizaciones que han decidido emprender la senda de la integración de la usabilidad en su proceso de desarrollo, por haber reconocido ya la importancia del grado de usabilidad del software que producen. Esta finalidad explícita no obsta, de todas formas, a que el marco de integración pueda ser indirectamente de utilidad a organizaciones que no se deciden a dar el paso de la integración, por el esfuerzo que conlleva actualmente.

La solución, así mismo, está planteada para **organizaciones en las que el desarrollo de software se lleva a cabo principalmente según el enfoque y las prácticas de la IS.** Puede resultar de utilidad a organizaciones con un rango variable de experiencia actual en IPO, a saber:

- Para organizaciones con escasa o nula experiencia en usabilidad, el marco de integración propuesto ofrece una selección entre la gran cantidad de técnicas IPO, que incluye aquellas cuya adopción puede resultar menos costosa. Esto no significa que la solución potencie que la labor de aplicación de todas las técnicas IPO vaya a ser realizada por ingenieros software, pero en el caso de tener escasez de personal con formación sólida en IPO, algunas de las técnicas sí que se proponen como aplicables por ingenieros software con una formación programada en dichas técnicas IPO. A pesar de que no se trata del caso ideal desde el punto de vista de la usabilidad, consideramos que se trata de una vía pragmática hacia la integración de la usabilidad en el desarrollo, puesto que, como apuntan Constantine y Lockwood [Constantine, 99], aunque tal vez sería deseable que las decisiones que afectan a la usabilidad del producto fueran tomadas únicamente por expertos en usabilidad, en la práctica los desarrolladores toman una gran cantidad de decisiones en sus labores de desarrollo que dan forma y determinan la usabilidad última del producto software. Así, consideramos que es preferible defender que quienes toman dichas decisiones adquieran una formación en usabilidad compatible con su concepción del desarrollo

⁷ Entendemos por "modus operandi habitual" la forma usual de desarrollo software que cada organización tenga. Esta forma o proceso habitual, entendemos que no deber ser centrada en el usuario sino un proceso tradicional o normal de desarrollo IS.

de software (la de la IS), que abogar por un enfoque todo-o-nada que se base en un cambio radical de escenario, en el que los expertos en usabilidad sustituyan a los ingenieros software en todos los puntos del desarrollo en los que deba tratarse la usabilidad.

- Para organizaciones con un departamento de usabilidad, o con un número apreciable de expertos en IPO en sus equipos de desarrollo, la solución propuesta puede contribuir a una mejor integración de sus esfuerzos con los del resto del equipo de desarrollo. Esto se conseguirá por la mejora en comunicación del enfoque IPO a los desarrolladores con una base predominantemente de la IS, contribuyendo de esta forma a la creación de verdaderos equipos multidisciplinares.

Aunque la solución propuesta resulta útil a las organizaciones que se encuentran en el segundo caso, la máxima utilidad pueden extraerla organizaciones del primer tipo. Por el contrario, en el caso de organizaciones donde el enfoque mayoritario de desarrollo es el de la IPO, la solución propuesta no es de utilidad directa.

El enfoque tomado en el presente trabajo es ofrecer una solución suficientemente general, como para que organizaciones con distintos procesos puedan aplicar la solución propuesta. De todas formas, tenemos que establecer cuáles son las condiciones mínimas que el proceso de una organización debe cumplir para poder considerar que, con la inclusión de las técnicas IPO pertinentes, podrá llegar a ser centrado en el usuario y, por tanto, tratar adecuadamente de la usabilidad. Con este fin se ha realizado un estudio para recopilar las características más representativas de un enfoque centrado en el usuario. Dichas características se dividen entre aquellas que pueden conseguirse mediante la inclusión de técnicas IPO específicas (implicación de usuarios, adecuada comprensión del usuario y de las tareas que realiza, y conocimiento multidisciplinar), y aquellas que son características intrínsecas del proceso (un enfoque iterativo de desarrollo). Así, puesto que las características del primer tipo ya estarían cubiertas por la inclusión de técnicas IPO, **la única característica que debe cumplir cualquier proceso de desarrollo candidato a ser mejorado con la inclusión de prácticas de usabilidad, es estar basado en un enfoque iterativo**. La primera parte del capítulo 6 constituye el estudio sobre las características que debe cumplir cualquier proceso candidato a la integración de las prácticas de usabilidad con ciertas garantías de éxito.

3.5 Esbozo de la Solución

El marco de integración que se detalla en el presente trabajo ofrece una visión acorde con la terminología de la IS de técnicas IPO seleccionadas como más adecuadas para el objetivo de la integración de prácticas de usabilidad en el proceso de desarrollo.

Como primer paso para el establecimiento de dicho marco, se realizó un estudio que permitiera clarificar la variedad existente en el campo de la IPO en cuanto a las actividades que componen el proceso de desarrollo. Cada autor propone un conjunto propio de actividades, con cierta diversidad terminológica que dificulta el acceso de un ingeniero software a la literatura existente. En la primera parte del capítulo 4 de este trabajo se ilustra tal esfuerzo de síntesis en una visión característica del proceso de la IPO en cuanto a las actividades que lo componen, obteniéndose un conjunto de actividades que consideramos las más representativas del campo de la IPO tras considerar las actividades particulares de cada fuente y la semejanza entre las mismas. Pero la heterogeneidad del campo de la IPO no se limita únicamente a las actividades del desarrollo, también se puede observar en las técnicas. Como se ha indicado en la sección 3.2.1, las técnicas IPO no siempre están descritas en la literatura como enmarcadas en una actividad del proceso. Por esta razón, ante la diversidad del campo de la IPO en cuanto a la asignación de técnicas a actividades, fue preciso realizar a

continuación un estudio acerca de las técnicas IPO propuestas en cuanto a su ubicación en actividades concretas. Dicho estudio comenzó con la recopilación de las distintas técnicas propuestas por los autores con su nomenclatura particular, para a continuación clasificar las técnicas similares según las actividades representativas del proceso IPO identificadas previamente, y, finalmente, se eligió un nombre genérico para cada grupo de técnicas identificadas como la misma técnica básica. Con este estudio se completó el mapa de las actividades y técnicas de la IPO, en el que nos basamos para la selección y relación posterior con tipos de actividades IS. El conjunto de técnicas IPO que forman dicho mapa asciende a 94.

Consideramos que un número tan alto de técnicas resulta poco manejable para un ingeniero software, especialmente teniendo en cuenta que pertenecen a un campo ajeno a la IS. Así, el marco de integración que proponemos resultaría demasiado farragoso si incluyera todas las técnicas del campo de la IPO, puesto que tal situación podría afectar a la aplicabilidad de la solución propuesta. Con el fin de facilitar la tarea al usuario del marco de integración resultado del presente trabajo de investigación, caracterizamos cada técnica IPO identificada en la literatura según un conjunto de seis criterios. Dichos criterios resumen las características que pueden resultar de mayor interés para el tipo de organizaciones a las que destinamos nuestra propuesta (descritas en la sección 3.4 más atrás). La caracterización según estos seis criterios cumple dos objetivos. Por un lado, servir de base para la selección de un conjunto de técnicas que sean especialmente adecuadas al propósito de integrar la usabilidad en una organización con escasa o nula experiencia previa en IPO. Por otro lado, servir de guía para que el usuario del marco de integración realice su propia selección entre las técnicas seleccionadas según las características específicas de su organización o los objetivos del proyecto en cuestión. Respecto al primer objetivo, los seis criterios se combinaron en un solo valor para cada técnica denominado "Valoración Total", que nos permitió realizar la selección de aquellas técnicas que pueden mejor contribuir al objetivo de integración de la usabilidad en el proceso. El total de técnicas seleccionadas para formar parte del marco de integración de la usabilidad en el desarrollo asciende a 35. La caracterización de técnicas IPO y la posterior selección constituyen la última parte del capítulo 4.

Una vez seleccionadas las técnicas IPO que se proponen para su posible integración en el proceso, las ubicamos en los tipos de actividades IS del desarrollo en las que resulta más útil su aplicación en cuanto a la mejora de la usabilidad del producto final. Para llevar a cabo tal asignación, en primer lugar se identificó la correspondencia entre las actividades IPO identificadas en el capítulo anterior y los tipos de actividades del desarrollo según la IS. En dicha asignación se estableció una diferenciación entre las actividades IPO cuyos objetivos son coincidentes con algún tipo de actividad IS, y aquellas que no tienen una correspondencia clara con tipos de actividades IS. Estas últimas consideramos que deben ser incorporadas a las prácticas de la IS si se desea la producción de software usable. El conjunto de la asignación del primer tipo de actividades IPO a tipos de actividades IS junto la relación de las actividades nuevas a incluir en el desarrollo constituye el mapa de actividades del desarrollo afectadas por la introducción de la usabilidad. Basándonos en dicho mapa realizamos la asignación de cada técnica IPO al tipo de actividad IS correspondiente, aunque la relación entre actividades IPO y tipos de actividades IS únicamente sirve como guía general. Debido a la mencionada falta de asignación estricta de técnicas a actividades en la literatura IPO, es preciso para cada técnica IPO considerar su ubicación en el tipo de actividad IS donde puede resultar más útil de cara a la usabilidad del producto final. Así para cada técnica IPO, según la actividad IPO en la que se enmarca y según su función tal y como se describe en la literatura IPO, se asignó a un tipo de actividad de las afectadas por la usabilidad. Toda la asignación a tipos de actividades IS, tanto de actividades como técnicas IPO, se detalla en el capítulo 5.

Dado que en un proceso iterativo no todos los ciclos son iguales, hemos considerado necesario incluir en la solución propuesta información sobre la más adecuada ubicación de cada técnica IPO seleccionada en las etapas de un desarrollo iterativo. Tal información se

basa en el hecho de que la literatura IPO describe ciertas técnicas como especialmente adecuadas para momentos específicos del desarrollo. Puesto que tal información es de utilidad a los desarrolladores para la integración de dichas técnicas en su proceso, se organizó de forma que pueda adaptarse a la diversidad de enfoques iterativos que pueden llevarse a cabo para el desarrollo de software. Según la información disponible en la literatura IPO, identificamos dos momentos del desarrollo especialmente importantes desde el punto de vista de la usabilidad. Por una parte, el momento en el que se cuenta con un concepto del producto establecido. Por otra, el momento en que una parte del sistema se encuentra instalada en el entorno del usuario final y en funcionamiento. Estos dos hitos dividen el desarrollo en tres etapas, las cuales denominamos ciclos iniciales, ciclos centrales y ciclos de evolución. Cualquier proceso con un enfoque iterativo puede acomodarse a este esquema genérico de etapas, para poder interpretar la información relativa al momento de desarrollo que ofrecemos en nuestra propuesta. Una vez establecida la base sobre la cual especificar este tipo de información, para cada técnica IPO identificamos su idoneidad para cada etapa, de acuerdo a sus objetivos en el desarrollo. Este estudio sobre los mejores momentos de aplicación de las técnicas IPO en un desarrollo iterativo se detalla en el capítulo 6, a continuación del estudio de los prerrequisitos a cumplir por el proceso.

Finalmente, la caracterización de técnicas IPO realizada según las distintas perspectivas se integró en un marco presentado en forma de tablas. Dicho marco condensa toda la información que puede ser útil a los ingenieros software en el objetivo de integrar las prácticas de usabilidad en el proceso de desarrollo. Con el objeto de facilitar el uso del marco, la información que contiene se estructuró en tres vistas: Según los nombres de las técnicas IPO seleccionadas (por técnicas), según el tipo de actividad en la que cada técnica se enmarca (por actividades), y según el momento del desarrollo en el que cada técnica puede resultar de mayor utilidad (por momentos del desarrollo). Las consideraciones que el usuario del marco debe realizar para seleccionar las técnicas IPO a integrar en su proceso, y para llevar a cabo dicha integración, son complejas y requieren valorar información desde distintas perspectivas. Así, la estructuración del marco en estas tres vistas permite al usuario del mismo realizar una búsqueda de información combinando las tres vistas según los criterios que esté considerando en cada momento. Para la mejor comprensión de la vista por actividades, se incluyó como introducción a dicha vista una descripción de los tipos de actividades IS considerados. De igual forma, como introducción a la vista por momentos, se incluyó una descripción del esquema genérico de etapas en un desarrollo iterativo empleado para describir dicha vista. El capítulo 7 detalla toda esta información, la cual se denomina marco de integración de la usabilidad en el proceso de desarrollo software, y que constituye el resultado del presente trabajo de investigación. Dicho capítulo también incluye una breve descripción de la herramienta web creada a partir de la solución propuesta, con el fin de posibilitar a los ingenieros software interesados un mejor acceso a la información contenida en el marco de integración.

3.6 Aplicación y Evaluación de la Solución Propuesta

La adecuada validación de cualquier propuesta metodológica cuenta con importantes obstáculos, debido a que las variables que influyen en el desarrollo de cualquier sistema software de una mínima complejidad son numerosas y difíciles de controlar. Así, la repetición de proyectos de desarrollo en un entorno controlado entraña una gran dificultad, y la comprobación de la validez de la solución en distintas situaciones resulta todavía más complicada.

Teniendo en cuenta dichas dificultades, nos vamos a centrar en la evaluación de la viabilidad de la solución propuesta en los siguientes casos:

1. La viabilidad se comprobará en un caso real de desarrollo de software en un entorno industrial. La solución propuesta se aplicará por parte de una compañía asentada, en un proyecto de desarrollo de un sistema que la compañía tiene intención de vender a distintos clientes de la compañía. Consideramos de vital importancia la aplicación real de cualquier solución metodológica del campo de la IS, puesto que si queda en un mero ejercicio teórico, corre un gran riesgo de acabar por no contribuir en ninguna medida a la práctica real de desarrollo software.
2. Con la finalidad de comprobar la aplicación por parte de equipos formados por desarrolladores con menor experiencia, y en proyectos de desarrollo de sistemas a medida, se va a estudiar la aplicación de la solución propuesta por parte de alumnos de la Facultad de Informática con experiencia variable de desarrollo de software. Esta aplicación se realizará por medio de dos casos de estudio. El primero de ellos consistirá en un desarrollo por parte de un alumno de postgrado, con experiencia media en desarrollo de software, mientras que el segundo caso de estudio consistirá en la aplicación de la solución propuesta por parte de alumnos de grado en el último curso que desarrollarán distintos sistemas a medida en equipo. En este último caso los equipos de desarrollo estarán formados por desarrolladores noveles.

Para cada uno de estos casos se va a estudiar el grado de cumplimiento de las hipótesis planteadas. Para ello se van a utilizar las siguientes herramientas:

- **Encuesta para los desarrolladores acerca de cómo han integrado la usabilidad en su desarrollo.** Se ha confeccionado un cuestionario personalizado para cada uno de los casos estudiados, y dichos cuestionarios se incluyen en los Anexos B, C y E. Las preguntas incluidas en los cuestionarios permiten estudiar el grado de cumplimiento de las tres primeras hipótesis, y piden a los desarrolladores que expresen lo siguiente:
 - SH1. Valoración de si el esfuerzo que conlleva aplicar las técnicas IPO compensa a la vista de los resultados obtenidos
 - SH2. Valoración del tipo de actividades del desarrollo afectadas por el tratamiento de la usabilidad.
 - SH3. Técnicas IPO concretas que han aplicado en las etapas de su proceso de desarrollo.
- **Encuesta realizada a usuarios representativos acerca de la usabilidad percibida del sistema** en comparación con la usabilidad de un sistema desarrollado previamente con el mismo objetivo, pero sin la integración de técnicas de usabilidad según la solución propuesta. La usabilidad es un atributo de calidad que depende del contexto en el que vaya a utilizarse el sistema, de usuarios específicos realizando tareas específicas, por lo que únicamente se puede probar una mejora de usabilidad en un entorno concreto entre dos sistemas diseñados para funcionar en dicho entorno. Así, en las tres aplicaciones de la solución propuesta que se van a estudiar se va a comparar la percepción de la usabilidad por los usuarios del sistema desarrollado con la percepción de la usabilidad de un producto equivalente. El cuestionario empleado en los casos de estudio se incluye en el Anexo D. De esta forma podremos valorar el grado de cumplimiento de la cuarta hipótesis en los casos estudiados:
 - SH4. El cuestionario incluye tanto preguntas que permiten valorar la percepción que tiene el usuario acerca de cuán usable es el sistema, como una pregunta planteada directamente acerca de la mejora percibida con respecto al producto equivalente anterior.

Capítulo 4

Actividades y

Técnicas en un

Proceso

Centrado en el

Usuario

4.1 Introducción

En este capítulo se detalla el estudio realizado sobre las actividades IPO presentes en un proceso centrado en el usuario, y las técnicas que se recomienda aplicar en cada actividad. Para aquellas técnicas que no son ubicadas directamente en ninguna actividad concreta en la literatura, hemos deducido qué actividad IPO es la más apropiada para su aplicación. El objeto del estudio es establecer las actividades y técnicas características del campo de la IPO, con el objeto de una posible integración en cualquier proceso de desarrollo, y seleccionar las más adecuadas para dicho objetivo de integración.

Mientras la IS ha tenido como uno de sus objetivos la formalización de un proceso de desarrollo explícito, para lograr un desarrollo de software sistemático y disciplinado, la IPO no tiene una visión detallada del proceso al mismo nivel. Cada autor del campo de la IPO tiene una visión particular del proceso, y de las actividades que forman parte de cualquier desarrollo en el que se persiga alcanzar un cierto nivel de usabilidad en el producto software. El estudio detallado en este capítulo se ha realizado para obtener el conjunto de actividades más típicas entre las propuestas por los distintos autores. Estas actividades van a relacionarse en capítulos posteriores con las actividades típicas de un proceso de desarrollo de la IS. De todas formas, el estudio de actividades de usabilidad es valioso por sí mismo como mapa de los conceptos establecidos en la IPO en torno al proceso de desarrollo, el cual puede servir como base para la comunicación entre los desarrolladores formados en IS y el equipo de usabilidad.

Con respecto a las técnicas IPO, la literatura de esta disciplina detalla un gran número de técnicas, puesto que hay una gran diversidad entre autores respecto a terminología y respecto

a técnicas propuestas. Para poder trabajar con un número de técnicas razonable, hemos elaborado una taxonomía, para lo cual se ha realizado un estudio en la literatura de la IPO para recopilar las técnicas recomendadas y asignarlas a las actividades en un proceso centrado en el usuario anteriormente obtenidas. Así mismo, se ha llevado a cabo una caracterización de las técnicas, para tener un criterio que guíe la selección de aquellas técnicas que se proponen para su posible inclusión en el proceso de desarrollo software.

En primer lugar, se detalla en la sección 4.2 el estudio de las actividades características de un proceso centrado en el usuario. A continuación, la sección 4.3 refleja la taxonomía de técnicas obtenida. En la sección 4.4 se caracteriza cada técnica según determinados parámetros relevantes para nuestro objetivo, y en la sección 4.5 se seleccionan las técnicas más apropiadas para el objetivo de inclusión en el proceso de desarrollo software.

4.2 Estudio de Actividades de Usabilidad en la Literatura IPO

Para poder presentar un esquema representativo de los procesos IPO, se ha realizado un estudio de las actividades de desarrollo propuestas en la literatura IPO. En primer lugar, se presentan las fuentes consideradas para este estudio y, a continuación, se detallan los resultados del estudio, con un cuadro resumen y una descripción de cada una de las actividades identificadas.

4.2.1 Fuentes Consideradas

Para este estudio de actividades de usabilidad en la literatura IPO se han considerado únicamente libros y estándares. Nuestro objetivo es estudiar lo que se considera como conocimiento comúnmente aceptado en el campo de la IPO y, por tanto, los libros y estándares son más adecuados para este fin que las publicaciones de investigación en revistas y congresos.

Entre la gran cantidad de libros y estándares publicados pertenecientes a la IPO, hemos escogido los volúmenes más relevantes y los más citados. A continuación, se detalla una breve descripción de cada obra escogida:

- [Hix, 93] – "Developing User Interfaces: Ensuring Usability Through Product and Process" Deborah Hix, H. Rex Hartson.

Esta obra presenta un enfoque muy práctico y aplicado al tema del diseño de la IU. Uno de sus objetivos es servir de libro de texto para cursos de desarrollo de IUs con un fuerte componente de usabilidad.

- [Nielsen, 93] – "Usability Engineering". Jakob Nielsen.

La ingeniería de usabilidad es una subdisciplina de la IPO que se centra en el establecimiento de objetivos de usabilidad medibles, y su evaluación por medio de test de usabilidad en cada ciclo del desarrollo. Este libro ha sido durante mucho tiempo la principal referencia del tema. Nielsen ofrece un enfoque cercano al de la ingeniería para la construcción de software usable, acercando de este modo la usabilidad a la IS.

- [Preece, 94] – "Human-Computer Interaction". Jenny Preece, Yvonne Rogers, Helen Sharp, David Benyon, Simon Holland, Tom Carey.

Esta obra trata toda la variedad de temas relacionados con el campo de la IPO. Tiene un objetivo enciclopédico, con cierta predominancia de los componentes teóricos. Ha sido uno de los principales libros de texto para cursos generales de IPO hasta el momento de la realización de este trabajo de tesis.

- [Wixon, 97] – "The Usability Engineering Framework for Product Design and Evaluation". Dennis Wixon, Chauncey Wilson. En *Handbook of Human-Computer Interaction*, 2nd edition.

La obra en la que aparece este artículo describe la diversidad del campo de la IPO, tanto en la investigación como en la práctica. El artículo de Wixon y Wilson ofrece una buena descripción general de la subdisciplina de la ingeniería de usabilidad. Los autores pertenecen al grupo de usabilidad en la compañía DEC que creó este enfoque (según se acredita en [Gould, 88]).

- [Shneiderman, 98] – "Designing the User Interface". Ben Shneiderman.

Ben Shneiderman es uno de los autores más reconocidos en el campo de la IPO. Recibió el premio de la asociación ACM-SIGCHI "*CHI Lifetime Achievement Award*" en el año 2001. Las tres ediciones de este libro han supuesto referencias fundamentales en el campo del diseño de IUs, debido a su equilibrado balance entre los aspectos de la interacción teóricos y los centrados en el desarrollo.

- [Constantine, 99] – "Software for Use". Larry L. Constantine, Lucy A.D. Lockwood.

Larry Constantine es ampliamente conocido en la IS. Ha desplazado su interés desde el diseño estructurado en los años 80 al desarrollo de software usable en los últimos 10 años. La experiencia de Constantine y Lockwood como consultores de usabilidad, destilada en forma de un método, se describe en esta obra, la cual tiene un carácter eminentemente práctico.

- [ISO13407, 99] – "ISO 13407. Human-Centred Design Processes for Interactive Systems". International Organization for Standardization.

Este estándar pretende ayudar a los responsables de la gestión de procesos de diseño hardware y software a identificar y planificar las actividades de diseño centrado en el humano en su momento adecuado y de forma efectiva. Los estándares ISO son la referencia principal del campo de la IPO, y han sido adoptados ampliamente por la industria de desarrollo de sistemas informáticos (incluyendo el diseño de elementos hardware).

- [Mayhew, 99] - "The Usability Engineering Lifecycle: A Practitioners Handbook for User Interface Design". Deborah J. Mayhew.

Este texto describe uno de los procesos IPO más completos desde el punto de vista de la IS, en cuanto a que define para cada actividad la precedencia con otras actividades, roles que participan, técnicas a aplicar y productos que se generan. Se trata de una de las referencias más citadas actualmente entre las que describen métodos IPO.

4.2.2 Resultados del Estudio

Hemos analizado la propuesta de cada fuente en cuanto a las actividades de usabilidad para extraer las actividades comunes, o aquellas que están al mismo nivel de abstracción y son comunes a varias fuentes. En la Tabla 4.1 se muestra el resultado de este estudio. Dado que el

objetivo es comparar las distintas propuestas, hemos agrupado en cada fila las actividades que se refieren al mismo concepto. Cada fila está etiquetada (en la primera columna de la tabla) bien con el término más genérico, o bien con aquel más utilizado por los diferentes autores del estudio. En la tabla se ha incluido una columna por cada fuente considerada, conteniendo las actividades que dicha fuente propone, de acuerdo con la siguiente notación:

- Cuando un autor presenta varias tareas dentro de la misma actividad, en la tabla aparece el nombre completo para la actividad. Por ejemplo, [Hix, 93] propone Análisis de Sistemas / Tareas / Funcional / de Usuarios y ha sido clasificada en la actividad de Especificación del Contexto de Uso.
- Cuando un autor describe una actividad genérica que incluye la actividad que estamos considerando como una subtarea, entonces la subtarea específica se indica al pie de la tabla con el número correspondiente entre paréntesis.
- Cuando un autor propone varias actividades que encajan con la actividad descrita, se incluyen en forma de lista precedidas por un asterisco (*). Por ejemplo, [Wixon, 97] propone dos actividades que encajan con la Especificación del Contexto de Uso: Especificar y categorizar a los usuarios, y Llevar a cabo un análisis de tareas.
- Aquellas actividades que no son mencionadas por el autor aparecen en la tabla con un guión (-).

En la Tabla 4.1, se puede observar que hay una gran coincidencia entre las fuentes consideradas en cuanto a las siguientes actividades: Especificaciones de Usabilidad, Prototipado y Evaluación de la Usabilidad. La Especificación del Contexto de Uso, sea como una actividad única o como la combinación de las actividades de Análisis de Tareas y de Usuarios, también es común entre las distintas fuentes.

Donde hemos encontrado un mayor grado de discrepancia es en las actividades de diseño (aparte del Prototipado), como son el Desarrollo del Concepto del Producto y el Diseño de la Interacción. Mientras algunos autores no tratan el asunto de cómo llevar a cabo el diseño, más allá de etiquetarlo como centrado en el usuario o abogar por el diseño iterativo, otros autores como Constantine y Lockwood [Constantine, 99] entran más a fondo en el diseño como actividad. En concreto, Constantine y Lockwood, critican la tendencia dominante en el campo de la ingeniería de usabilidad, la cual se centra casi exclusivamente en la evaluación de la usabilidad, descuidando la oportunidad de intentar realizar un buen diseño desde el principio. Ésta puede ser la razón por la que el tema del diseño de la interacción recibe menos atención en el resto de autores considerados. Con la actividad relativa al Concepto del Producto sucede algo parecido, pues únicamente dos fuentes ([Shneiderman, 98] y [Preece, 94]) lo consideran claramente como actividad en el desarrollo, y otras dos más ([Hix, 93] y [Mayhew, 99]) lo consideran de forma tangencial en la descripción de un diseño conceptual que precede a un diseño detallado. Sin embargo, tanto la creación de un concepto del producto adecuado como el diseño de la interacción tienen un papel importante en los principales textos de usabilidad consultados, por tanto su no inclusión como actividad en el proceso de desarrollo puede deberse a que corresponden a una parte del desarrollo más reciente en el campo de la IPO, comparada con, por ejemplo, la evaluación de usabilidad, que se trata de una parte ampliamente explorada.

Tabla 4.1 Actividades de Usabilidad por Fuente

ACTIVIDAD	Hix, 93	Nielsen, 93	Preece, 94	Wixon, 97	Shneiderman, 98	Constantine, 99	ISO13407, 99	Mayhew, 99
ESPECIFICACIÓN DEL CONTEXTO DE USO	Análisis de Sistemas / Tareas / Funcional / de Usuarios	Conocer al usuario	Análisis Funcional / Análisis de Tareas	*Especificar y categorizar a los usuarios *Llevar a cabo un análisis de tareas	Llevar a cabo una investigación y análisis de necesidades	Modelado de Tareas	Comprender y especificar el contexto de uso	*Perfil de Usuarios *Análisis Contextual de Tareas
ESPECIFICACIONES DE USABILIDAD	Requisitos / Especificaciones de Usabilidad	Establecimiento de objetivos	Especificación de Requisitos	*Definir objetivos de usabilidad cuantitativos *Establecer niveles deseados de usabilidad para cada objetivo	Diseñar conceptos y un prototipo de las pantallas clave (1)	-	Especificar los requisitos de usuario y organizacionales	Establecimiento de Objetivos de Usabilidad
DESARROLLO DEL CONCEPTO DEL PRODUCTO	Diseño Conceptual	-	Diseño Conceptual / Diseño Formal	-	Desarrollar el concepto del Producto	-	-	Diseño del Modelo Conceptual
PROTOTIPADO	Prototipado Rápido	Prototipado	Prototipado	-	Diseñar conceptos y un prototipo de las pantallas clave	-	Producir soluciones de diseño (2)	Prototipado de los Estándares de Diseño de Pantalla
DISEÑO DE LA INTERACCIÓN	Diseño y representación del diseño	Diseño Iterativo	Diseño Conceptual / Diseño Formal	-	Llevar a cabo diseño y refinamiento iterativo	Modelado del Contenido de la Interfaz	Producir soluciones de diseño	Diseño Detallado de la Interfaz de Usuario
EVALUACIÓN DE LA USABILIDAD	Evaluación de la Usabilidad	Evaluación de la Interfaz	Evaluación	Probar el producto contra los objetivos de usabilidad	Llevar a cabo diseño y refinamiento iterativo (3)	Inspección de Usabilidad	Evaluar la usabilidad contra los requisitos	Evaluación Iterativa del Diseño Detallado de la Interfaz de Usuario

Subtareas específicas que corresponden a la actividad descrita dentro de una actividad más amplia:

- (1): Crear objetivos de usabilidad específicos basados en las necesidades de los usuarios
- (2): Hacer las decisiones de diseño más concretas, utilizando simulaciones, modelos, maquetas, etc.
- (3): Llevar a cabo test de usabilidad a escala completa

La escasa formalidad en los procesos IPO conduce a que las actividades relacionadas con la creación del concepto del producto y con el diseño de la interacción hayan sido ignoradas o tratadas levemente por una gran cantidad de autores.

Por tanto, podemos ceñirnos a las siguientes actividades como representativas de un proceso centrado en el usuario: Especificación del Contexto de Uso, Especificaciones de Usabilidad, Prototipado, y Evaluación de la Usabilidad; mientras que vamos a aceptar con reservas las actividades de Desarrollo del Concepto del Producto y Diseño de la Interacción, más por consonancia con su tratamiento como puntos importantes en la literatura IPO, que por su presencia como actividades en la mayoría de las fuentes consideradas.

En la Figura 4.1 se muestran estas actividades de usabilidad, organizadas según la clasificación tradicional de actividades de desarrollo software en análisis, diseño y evaluación. Esta clasificación no es propia directamente de la IPO, sino una organización tradicional de actividades de construcción de software. Por esta razón se representan estos grupos en la figura mediante cajas con línea discontinua y en color gris.

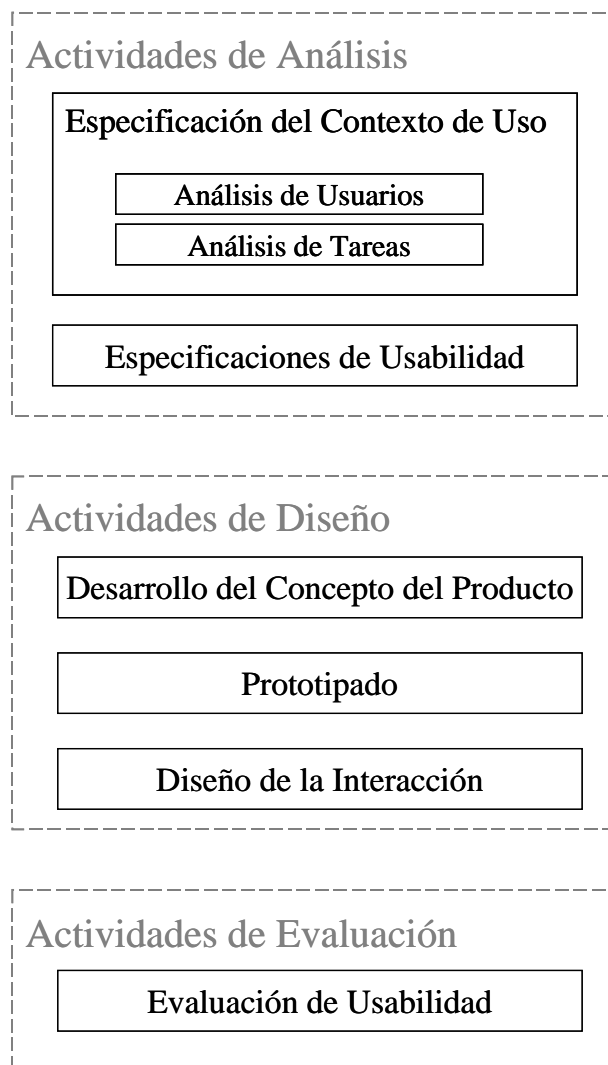


Figura 4.1 - Actividades de Usabilidad

Nótese que la actividad de Especificación del Contexto de Uso se descompone en Análisis de Usuarios y Análisis de Tareas. Algunos autores, como [Wixon, 97] y [Mayhew, 99], hacen distinción entre estas dos actividades, mientras [Hix, 93] nombra como parte de la actividad

los dos tipos de labores, aun cuando todos ellos reconocen que Análisis de Usuarios y de Tareas están estrechamente relacionados. Debido a esta estrecha relación y a que "Especificación del Contexto de Uso" es un término ampliamente aceptado en la IPO (pues es la terminología del estándar ISO 13407 [ISO13407, 99]), hemos decidido utilizarlo para designar esta actividad. De todas formas, para no perder la información que algunos autores proporcionan al distinguir entre las labores de Análisis de Usuarios y de Tareas, hemos incluido ambas como subactividades de la Especificación del Contexto de Uso. Se trata de la única actividad en la que consideramos subactividades, porque en este caso está documentada en la literatura la estrecha relación que existe entre ambas subactividades y, por otra parte, en la mayor parte de las fuentes consultadas se considera una única actividad.

En las siguientes secciones se detalla cada una de las actividades recogidas.

4.2.3 Especificación del Contexto de Uso

El objetivo de esta actividad es comprender y registrar las características del contexto previsto de uso, en cuanto éstas puedan ser relevantes para la usabilidad del producto software final, de cara a su consideración en las tareas de diseño.

El contexto de uso es un término muy amplio que se compone de varios aspectos interrelacionados. Tal y como se define en [ISO13407, 99] consta de los siguientes componentes:

- Las características de los usuarios a los que está dirigido el software. La identificación de estas características se conoce como Análisis de Usuarios.
- Las tareas que los usuarios van a realizar. El Análisis de Tareas se ocupa de este asunto.
- El entorno en el que los usuarios van a utilizar el sistema, incluyendo el hardware, software y materiales que se van a utilizar.

Se van a describir a continuación las primeras dos subactividades, Análisis de Usuarios y Análisis de Tareas, con un mayor nivel de detalle, mientras que la tercera se va a considerar como parte de las labores de análisis que se llevan a cabo en el Análisis de Usuarios.

4.2.3.1 Análisis de Usuarios

El Análisis de Usuarios considera e identifica para los usuarios previstos sus conocimientos, necesidades y características, que sean relevantes en su interacción con el sistema. Las características a identificar incluyen el conocimiento del dominio, destreza, experiencia, formación, características físicas, hábitos, preferencias y aptitudes. Para cierto tipo de sistemas también pueden ser relevantes características como la edad, discapacidades, daltonismo, etc. Todas estas características se estudian con el fin de poder adaptar el sistema a desarrollar a sus futuros usuarios.

También es de interés para el estudio realizado en las tareas de análisis de usuarios el tipo de hardware y software que utilizan, así como la experiencia que tienen con sistemas informáticos en general, y con aquellos que son similares al sistema a desarrollar en particular. De hecho, incluso el tipo de sistema al que la población de usuarios está acostumbrada puede ser relevante, con el fin de asegurar que el sistema se diseña según las expectativas que los usuarios puedan tener, las cuales se suelen basar en sus experiencias previas con sistemas informáticos.

El entorno físico también es importante aunque, estrictamente hablando, no se trata de una característica del usuario. Se deben considerar elementos como los altos niveles de ruido, la

baja luminosidad, la temperatura, y otras características similares referentes al lugar de trabajo del usuario. El software se tiene que diseñar, entonces, de tal forma que supere estas limitaciones del entorno de uso previsto en la medida de lo posible, aunque el sistema resultante acabe por ser menos eficiente en otros entornos.

El entorno social y/o laboral también puede ser un tema a considerar, cuando la estructura organizacional y las prácticas de trabajo son relevantes para el diseño a realizar. Por ejemplo, puede ser necesario considerar que el usuario será interrumpido frecuentemente en su trabajo por compañeros o por clientes. Otra posibilidad a tener en cuenta en ciertos proyectos se refiere al factor fatiga, que puede aparecer en el usuario debido a las características específicas del tipo de trabajo que realiza.

El estudio de los usuarios puede llevar a la necesidad de considerar distintos grupos de usuarios objetivo. La población de usuarios no tiene porqué ser un grupo único y homogéneo. En muchos proyectos es necesario considerar la estratificación de los usuarios en una serie de grupos relevantes. En estos casos el análisis de usuarios se complica en cierta medida, puesto que la selección de los grupos de usuarios a considerar requiere un estudio en sí mismo, y el análisis de usuarios se tiene que realizar entonces sobre cada uno de los grupos de usuarios identificados.

4.2.3.2 Análisis de Tareas

El Análisis de Tareas tiene como finalidad "obtener descripciones de lo que las personas hacen, representar estas descripciones, predecir dificultades y evaluar los sistemas contra requisitos de usabilidad o funcionales" [Preece, 94]. En resumen, se ocupa de lo que las personas hacen para llevar a cabo los asuntos de los que se ocupan.

El Análisis de Tareas está íntimamente relacionado con la Educación y Especificación de Requisitos, pero tiene la característica clave de centrar este tipo de actividades en los objetivos últimos del uso del sistema por parte del usuario. La diferencia entre una tarea y una funcionalidad es que la tarea tiene significado en sí misma para el usuario, mientras que la funcionalidad tiene sentido para el sistema software. El usuario considera necesario o deseable realizar las tareas. Por tanto, el término tarea implica una intención o propósito que no tiene porqué estar presente en el concepto de funcionalidad que ofrece un sistema. Por ejemplo, una funcionalidad para generar ficheros intermedios (necesarios para el buen funcionamiento del sistema según está concebido), no corresponde a un propósito del usuario, sino que es algo que el sistema requiere para su ejecución. En ocasiones funcionalidad y tarea pueden coincidir.

La descripción de la tarea debe incluir el rol que el usuario desempeña en la ejecución global de la tarea, no únicamente en términos de las funcionalidades provistas por un producto o sistema. El Análisis de Tareas puede considerarse como orientado a funcionalidades, pero suplementa los enfoques de los métodos orientados a funcionalidades de la IS con la consideración de las intenciones del usuario cuando realiza una tarea. La principal dificultad para un ingeniero software que quiera llevar a cabo la actividad de Análisis de Tareas estriba en su semejanza con la descomposición funcional, que puede llevar a utilizar los criterios habituales en tal tipo de descomposición, en lugar de usar los propios del Análisis de Tareas. El Análisis de Tareas debe estar siempre centrado en los objetivos del usuario.

En torno a la actividad de Análisis de Tareas observamos cierta confusión en el campo de la IPO. Algunos autores como Preece et al. [Preece, 94] o Mayhew [Mayhew, 99] utilizan el término para referirse a la actividad de analizar las tareas que se realizan en la actualidad, mientras que Hix y Hartson [Hix, 93] destacan lo variada que es la literatura IPO al respecto, y eligen el término para referirse al diseño de las tareas que el sistema ofrecerá. En este trabajo de investigación no vamos a optar por ninguna de las dos visiones, pues ambas están

implícitas en algunas técnicas IPO que consideramos valiosas para la producción de software usable y, por tanto, forman parte del aporte de usabilidad que se pretende integrar en el proceso de desarrollo. El Análisis de Tareas es muy importante para el diseño de las funcionalidades que el sistema ofrecerá. Cuando se lleva a cabo adecuadamente, está en la base de un verdadero proceso de desarrollo centrado en el usuario, puesto que los objetivos del usuario deben ser tenidos en cuenta y colocados en un lugar prominente a lo largo del proceso de desarrollo.

4.2.4 Especificaciones de Usabilidad

Las Especificaciones de Usabilidad son objetivos de usabilidad cuantitativos, que se utilizan como guía para determinar cuándo un sistema alcanza el nivel de usabilidad adecuado. Pueden considerarse un subconjunto de los requisitos no-funcionales de un sistema. Se basan en dos componentes, por un lado la eficiencia del usuario, y por otro su satisfacción. La eficiencia se interpreta como el nivel de eficiencia deseado del sistema más amplio que forman el usuario junto con el sistema software, trabajando de forma conjunta para conseguir ciertos objetivos del usuario. En este sentido, las Especificaciones de Usabilidad pueden denominarse también valores de referencia o *benchmarks* de usabilidad. El uso del término *benchmark*, habitualmente traducido en la IS como banco de pruebas, es acorde con el uso habitual en la IPO, puesto que las Especificaciones de Usabilidad son una definición de qué valores se esperan para la eficiencia y satisfacción del usuario realizando tareas específicas. Es decir, las Especificaciones de Usabilidad se definen sobre tareas específicas definidas a partir de las descripciones de tareas generales obtenidas en las actividades de Especificación del Contexto de Uso. Así, el conocimiento reunido en dichas actividades sirve como base para esta actividad. Las Especificaciones de Usabilidad se definen según las características de la población de usuarios obtenidas en el Análisis de Usuarios, y según los objetivos y tareas identificados en el Análisis de Tareas.

El conjunto de las Especificaciones de Usabilidad representan el criterio de aceptación del sistema desde el punto de vista de su usabilidad. Se evalúan al final de cada ciclo del desarrollo, estableciéndose cuánto progreso se ha hecho hacia el objetivo marcado de usabilidad. De esta forma, sirven como criterio para determinar cuándo se deja de iterar.

Aunque los atributos de usabilidad no son directamente medibles, las Especificaciones de Usabilidad deben serlo. Por tanto, los atributos de usabilidad se descomponen en subatributos y se particularizan para tareas específicas, basándose en los resultados de la Especificación del Contexto de Uso. Así, se pueden medir indirectamente los valores de los atributos de usabilidad. Cada especificación de usabilidad está ligada a un atributo de usabilidad concreto, pero referido a un aspecto particular del atributo en cuestión.

Algunos autores ([Nielsen, 93], [Wixon, 97] y [Mayhew, 99]) prefieren el término "objetivos de usabilidad", puesto que estas especificaciones se establecen como un objetivo a cumplir en el diseño del sistema. Los objetivos de usabilidad dirigen el diseño como parte de la información compartida por el equipo de desarrollo completo, sirviendo como posible criterio de decisión entre distintas alternativas de diseño que pueden presentarse. Su papel no es únicamente como caso de prueba que se evalúa para comprobar si se ha cumplido o no. Entre ambos términos hemos preferido "Especificaciones de Usabilidad", porque se acerca más a la terminología de la IS para indicar características que el sistema a implementar debe cumplir.

4.2.5 Desarrollo del Concepto del Producto

La base de esta actividad yace en los modelos mentales ([Norman, 90], [Preece, 94]). El concepto de modelo mental pertenece al campo de la psicología cognitiva, y se ha manifestado en las teorías psicológicas de la IPO en multitud de formas [Preece, 94].

Los usuarios siempre desarrollan un modelo en su mente acerca de cómo funciona el sistema, independientemente de que en el desarrollo se haya llevado a cabo una actividad explícita de desarrollo del concepto del producto. Este modelo mental del usuario es normalmente un modelo imperfecto, esto es, no se corresponde exactamente con la forma de funcionamiento del sistema, con la lógica que guía los procesos que lleva a cabo para responder a cada acción del usuario, y con la forma en la que están distribuidas las opciones y funcionalidades en la IU. Lo ideal para los diseñadores del sistema es que el usuario pudiera ser capaz de crear el modelo mental en su cabeza rápida y fácilmente, y que se correspondiera con la imagen del sistema.

El modelo de diseño es el que tienen en la cabeza los diseñadores, sea implícito o explícito, y en el que se basan para desarrollar el sistema. Si el sistema no sigue las expectativas del usuario, éste lo encontrará difícil de usar y desconcertante, y cometerá más errores usándolo. También, si el concepto del producto es ambiguo, inconsistente, oscuro, o no se transmite bien al usuario, éste se hará un modelo mental incorrecto, y esta situación llevará a un uso pobre del sistema. En la Figura 4.2 (tomada de [Preece, 94]) se muestra cómo se relacionan el modelo de diseño, la imagen real del sistema y el modelo mental que se forma en la cabeza del usuario. El usuario desarrolla un modelo mental parcial (comparado con el modelo completo de diseño que los diseñadores poseen), y, por tanto, sólo puede aprovechar una parte de las posibilidades que ofrece el sistema según ha sido concebido por sus diseñadores.

El concepto del producto al que se refiere el nombre de esta actividad es el modelo de diseño de la Figura 4.2. El término no está definido en las fuentes consultadas con la nitidez con la que están definidos otros conceptos IPO como, por ejemplo, las Especificaciones de Usabilidad. Se trata de un término elusivo, porque a la hora de describirlo distintos autores lo interpretan de distinta forma. En cualquier caso, hay coincidencia en las fuentes que lo mencionan en cuanto a la gran importancia de ayudar a que los usuarios desarrollen modelos mentales productivos, con el objeto de mejorar la curva de aprendizaje. Sin embargo, no hay indicaciones precisas acerca de cómo conseguir este objetivo, aparte de establecer un modelo de diseño consistente y acorde con las expectativas del usuario.

La IS no suele preocuparse del concepto del producto que el usuario conforma en su mente. Varias fuentes IPO consideradas en este estudio citan técnicas que se aplicarían en la actividad de Desarrollo del Concepto del Producto, puesto que tienen un carácter exploratorio y pretenden obtener una visión común por parte del equipo de diseño del sistema que se va a desarrollar. Puesto que se trata de un tipo de tareas que no se llevan a cabo habitualmente como parte de los procesos de desarrollo de la IS, y engloban una serie de técnicas que son de interés para nuestro propósito, hemos decidido incluir esta actividad de usabilidad, a pesar de concitar un menor consenso que otras de las que aparecen en la Figura 4.1.

En esta actividad se consideran las reglas generales que gobernarán el funcionamiento del sistema, sus espacios de interacción principales y cómo se trabajará con el mismo. En el caso de que se decida emplear algún tipo de metáfora (la cual relaciona el concepto del producto con elementos existentes en la realidad), se considera en esta actividad. La finalidad del uso de metáforas es que el modelo del diseño sea más fácilmente aprehensible para el usuario.

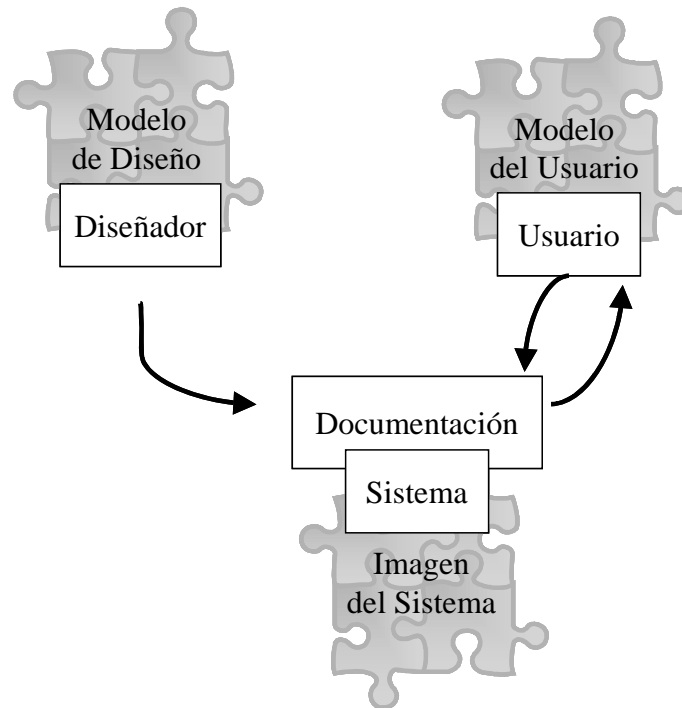


Figura 4.2 - Relación entre el Modelo de Diseño, la Imagen del Sistema y el Modelo del Usuario
[Preece, 94] (tomado de [Norman, 90])

Cuando un sistema no tiene una definición clara y explícita de su concepto del producto, es muy probable que sea inconsistente, y el usuario no será capaz de comprender la lógica del sistema, porque no existirá tal lógica. Los buenos diseñadores siempre tienen en mente un cierto concepto del producto, pero con eso no basta. Es necesario preocuparse de hacerlo explícito y acorde a las expectativas del usuario. Resaltar su importancia en el proceso de desarrollo ayuda a dar forma al sistema de tal forma que comunique explícitamente el concepto del producto al usuario. Uno de los objetivos principales de esta actividad es conseguir una visión del producto común dentro del equipo de diseño. Tal objetivo resulta cercano a una de las preocupaciones típicas de la IS, el modelado, el cual ayuda a la comunicación en el seno del equipo y evita las ambigüedades. En el caso del concepto del producto de la IPO, se incluye también al usuario entre los objetivos de comunicación. La búsqueda de la coherencia y la lógica interna en un diseño es un objetivo innato en cualquier tipo de ingeniería, y el aporte de la IPO se centra en el matiz de enfocar dicha coherencia a las expectativas y conocimiento previo del usuario.

4.2.6 Prototipado

El estándar ISO 13407 [ISO13407, 99] de IPO define un prototipo como "una representación de todo o parte de un producto o sistema que, aunque limitado de algún modo, puede utilizarse con fines de evaluación". Los prototipos permiten a los diseñadores comunicarse de forma más efectiva con los usuarios, y reducen la necesidad y el coste que conlleva rehacer un sistema ya implementado cuando los problemas se identifican tarde en el desarrollo. Es necesario construir prototipos porque las especificaciones técnicas y los modelos abstractos no suelen ser una buena vía de comunicación cuando se quiere involucrar a usuarios en el proceso de desarrollo.

El Prototipado está íntimamente ligado al desarrollo iterativo. Para que los prototipos sean efectivos, deberían tener un coste mínimo en términos de recursos y tiempo. La diferencia entre el tipo de Prototipado propuesto por la IPO y el empleado habitualmente en los

desarrollos de la IS se encuentra, de nuevo, en el enfoque. Los prototipos son útiles desde un punto de vista de usabilidad cuando reflejan principalmente la interacción usuario-sistema, de forma que pueden transmitir cómo va a funcionar el sistema desde el punto de vista del usuario. Así, los prototipos se utilizan para probar ideas de diseño con usuarios y para recoger sus impresiones [Preece, 94]. Ciertos prototipos IPO sirven para probar ideas de diseño en las etapas más tempranas del desarrollo, por lo que son muy poco costosos y tienen una baja fidelidad con el producto final. El coste y la fidelidad son los puntos principales de diferenciación entre los prototipos IPO y los prototipos IS.

El diseño, sea cual sea el nivel de abstracción al que se aborda, debería comunicarse por medio de un prototipo. Desde el concepto del producto al diseño detallado completo, se pueden elaborar prototipos de fidelidad creciente al producto final, con el fin de utilizarlos en las actividades de evaluación de la usabilidad. El enfoque de Prototipado en el desarrollo de sistemas interactivos implica la elaboración de, al menos, una versión inicial del sistema que ilustra las características principales del futuro sistema. Cuando se utiliza en etapas tempranas del proceso de desarrollo, un prototipo alienta la participación e implicación del usuario, y permite a los desarrolladores observar el comportamiento de los usuarios y su reacción ante el prototipo [Hix, 93].

Un objetivo importante de las actividades de Especificación del Contexto de Uso, consiste en asegurar que el sistema propuesto tiene toda la funcionalidad necesaria para llevar a cabo las tareas que el usuario quiere realizar. El Prototipado proporciona un medio de asegurar este objetivo, puesto que puede servir para educir información de los usuarios sobre [Preece, 94]:

- Las funcionalidades necesarias del sistema.
- Secuencias de operaciones.
- Necesidades de soporte al usuario.
- Representaciones necesarias.
- Aspecto y sensación (*look and feel*) de la IU.

Con el actual auge de los enfoques iterativos en la IS, los prototipos son cada vez más comunes en el desarrollo de software. La IPO ofrece la posibilidad de ligar tales prototipos a las raíces de un enfoque centrado en el usuario. El uso de ciertos prototipos IPO, menos costosos, permite realizar un número mayor de iteraciones, aumentando de esta forma las posibilidades de contrastar la visión del equipo de desarrollo con la visión del cliente y de los usuarios.

4.2.7 Diseño de la Interacción

El Diseño de la Interacción es la actividad de los procesos IPO que está menos definida, puesto que varía considerablemente entre las distintas fuentes consultadas. Hix y Hartson [Hix, 93], por un lado, indican que el diseño es una actividad compleja y que no hay fórmulas aplicables de forma general que garanticen el éxito y, por otro lado, aseguran que el diseño como proceso es una de las actividades menos comprendidas. Un consejo común a varias fuentes consiste en mantener a lo largo de todo el proceso de diseño un enfoque centrado en el usuario y, con este fin, el equipo de diseño debe basar su trabajo en los productos elaborados en actividades previas de usabilidad.

El Diseño de la Interacción y el Diseño de la IU están estrechamente relacionados, y las diferencias de terminología entre la IS y la IPO, destacadas en secciones anteriores, han hecho que se confundan ambos conceptos. La mayoría de autores emplean el término Diseño de la IU ([Nielsen, 93], [Hix, 93], [Mayhew, 99] y [Shneiderman, 98]), mientras que [Preece, 94] utiliza el término Diseño de la Interacción, y otros usan el término diseño de forma general ([Wixon, 97] e [ISO13407, 99]). Finalmente, [Constantine, 99] se refiere al diseño del diálogo

o al diseño visual. Hemos preferido el término "Diseño de la Interacción" para enfatizar en un contexto de IS que no se trata únicamente del diseño de los elementos visibles de la IU.

El Diseño de la Interacción se encarga de la definición de los entornos de interacción y su comportamiento. Al incluir el comportamiento implica coordinar la interacción entre el usuario y el sistema, lo cual lleva a decisiones de diseño que afectan a la estructura interna del sistema. También incluye el diseño de los elementos visuales que forman la interfaz gráfica de usuario, cuando la interfaz es de dicho estilo.

4.2.8 Evaluación de Usabilidad

La usabilidad es un concepto complejo, debido a la naturaleza compleja de los seres humanos. Sin llevar a cabo algún tipo de evaluación es imposible saber si el sistema satisface las necesidades de los usuarios y si encaja adecuadamente en el contexto físico, social y organizacional en el que va a ser usado [Preece, 94]. No importa cuánto se enfatice la realización de actividades de usabilidad en el proceso de desarrollo, no se puede predecir con exactitud el nivel de usabilidad del sistema por adelantado. Por esta razón es necesario realizar actividades de Evaluación de Usabilidad a lo largo de todo el desarrollo, especialmente al final de cada ciclo iterativo, para conocer qué nivel de usabilidad ha alcanzado el producto, y cuánta mejora será necesaria para cumplir los objetivos de usabilidad establecidos.

Se necesita realizar Evaluación de Usabilidad en todas las etapas del desarrollo, aunque el grado de formalidad necesaria varía, según los productos de diseño disponibles para evaluación en cada momento, y según las limitaciones de tiempo y recursos.

La evaluación se puede usar según el estándar ISO 13407 [ISO13407, 99] para:

- Proporcionar retroalimentación que sirva para la mejora del diseño.
- Evaluar si los objetivos del usuario y organizacionales se han alcanzado.
- Monitorizar el uso a largo plazo del producto o sistema.

Una diferencia entre la Evaluación de Usabilidad y la Validación y Verificación tradicionales de la IS es que la Evaluación de Usabilidad temprana tiene como finalidad dirigir el diseño. Este enfoque se conoce como evaluación formativa [Nielsen, 93], puesto que ayuda a dar forma al diseño. El objetivo de la evaluación formativa es aprender qué aspectos concretos del sistema son buenos y cuáles no lo son, y de qué forma puede mejorarse el diseño. Esta aproximación es la opuesta a la evaluación sumativa que aspira únicamente a dar un valor que refleje la calidad de un sistema cuando está terminado o casi terminado.

La Evaluación de Usabilidad es una parte fundamental del enfoque iterativo del desarrollo, porque las actividades de evaluación pueden producir soluciones de diseño para su aplicación en el próximo ciclo de desarrollo o, al menos, un mayor conocimiento sobre la naturaleza del problema de interacción detectado. Por tanto, la Evaluación de Usabilidad no es una prueba con un resultado apto/no-apt, sino que es parte inherente del proceso de desarrollo.

4.3 Estudio de Técnicas de Usabilidad en la Literatura IPO

Para abordar el estudio de las técnicas de usabilidad presentes en las fuentes consultadas, se sigue en esta sección un esquema similar al de la sección 4.2 referente a las actividades. El objetivo de este estudio es obtener un conjunto de técnicas representativas de las propuestas

en el campo de la IPO, las cuales serán las candidatas a ser incorporadas en el proceso de desarrollo software. Dado que existe un número de técnicas mucho mayor que de actividades, y debido a la dispersión observada en la literatura existente, en primer lugar se realiza un estudio recopilatorio para obtener una taxonomía de técnicas, organizadas según las actividades de usabilidad en las cuales se aplican.

Puesto que esta investigación se realiza en el ámbito de la IS, mantenemos los conceptos de actividad y técnica según se entienden en la terminología de proceso software de la IS. Actividad se refiere a una tarea a realizar, mientras que técnica alude a la herramienta conceptual mediante la cual se realiza la tarea (la forma en la que se lleva a cabo). En otras palabras, en la IS se hace una distinción entre qué hay que hacer (actividad) y cómo se hace (técnica). Sin embargo, estos dos conceptos no están delimitados con tanta claridad en la mayoría de las fuentes IPO consideradas, puesto que el qué y el cómo se confunden en más de una ocasión. Por esta razón, algunos conceptos que aparecían en la sección anterior como actividades también aparecen en esta sección como técnicas. Sin embargo, el enfoque difiere entre ambas situaciones, dependiendo de si son estudiadas con una perspectiva de qué o de cómo. Por ejemplo, se ha incluido en la sección anterior la actividad "Requisitos/Especificaciones de Usabilidad" mencionada en [Hix, 93]. Desde un punto de vista de actividad, hemos registrado que, en algún momento durante el desarrollo, se dedican esfuerzos a elaborar las Especificaciones de Usabilidad. Con una perspectiva de técnica, en esta sección, analizamos qué formato concreto proponen los autores y cómo se rellena para producir dichas Especificaciones de Usabilidad. Esto es, qué método o técnica propone el autor para llevar a cabo esta actividad.

En primer lugar, se presentan las fuentes consideradas y, a continuación, se detallan los pasos seguidos para realizar la labor de recopilación y clasificación. El resultado de dicha clasificación de técnicas se describe posteriormente, para finalizar con la caracterización de técnicas que permite realizar una selección entre las mismas.

4.3.1 Fuentes Consideradas

Las fuentes consultadas para este estudio son las mismas que se han detallado en la sección anterior, a excepción de [Wixon, 97] e [ISO13407, 99]. Estas dos fuentes no se han considerado aquí por estar centradas en las actividades de desarrollo, y no incluir apenas detalles sobre las técnicas individuales a aplicar en cada actividad. Para el resto de fuentes, en el Anexo A se detallan las técnicas recogidas en cada fuente considerada.

Además de las técnicas mencionadas en estas fuentes, se ha considerado la técnica **Personas**, por tratarse de una técnica de uso habitual por parte de expertos en usabilidad, a pesar de lo reciente de su aparición (la primera cita en la literatura IPO data de 1999 [Cooper, 99]). Su uso se encuentra especialmente extendido en el desarrollo para la web, aunque puede usarse para el diseño de software de cualquier tipo [Cooper, 99]. Una muestra del impacto que tiene esta técnica actualmente la podemos encontrar en su uso por el portal MSN de Microsoft (con el nombre "MSN Personas") en su estrategia de marketing para conseguir anunciantes, indicando que se preocupan por quiénes son sus usuarios [MSN, 03]. En los siguientes sitios web de prestigio que tratan el tema de la usabilidad también se nombra la técnica como técnica que contribuye al desarrollo de software usable:

- *UsabilityFirst* <http://www.usabilityfirst.com/> [DiamondBullet, 04]

Compendio de los principales conceptos de usabilidad, ofrecido por la compañía Diamond Bullet. Esta firma ofrece servicios de consultoría en usabilidad. El sitio se creó en 2002 y está actualizado en 2004. Contiene más de 1000 términos en su

glosario de usabilidad. Nótese que la técnica aparece en dicho glosario como "persona" en vez de "personas".

- *Dey Alexander: Web usability and user-centred design consulting and resources* <http://deyalexander.com/> [Alexander, 04]

Sitio web mantenido por Dey Alexander, consultor en usabilidad y accesibilidad en la web. Contiene una excelente colección de recursos relacionados con la usabilidad en el campo del desarrollo web, formada por numerosos enlaces a otras fuentes de información en la web para cada tema tratado, junto con una breve explicación. Incluye la técnica Personas en su relación de métodos y técnicas de diseño.

- *SAP Design Guild* <http://www.sapdesignguild.org/> [SAP, 04]

SAP ha adoptado una política en los últimos años consistente en hacer de la usabilidad un objetivo estratégico de la compañía. Este completo sitio web lo ofrece la compañía como prueba de su compromiso con la usabilidad, con la finalidad de compartir su conocimiento con la comunidad de diseño de IUs en general. La técnica Personas es mencionada en las páginas que discuten la creación de portales web.

Sobre el origen de la técnica Personas, nótese que Cooper presenta la técnica como una creación suya en [Cooper, 03b], aunque admite que construcciones similares han sido utilizados anteriormente por otros diseñadores de la interacción o profesionales del marketing. Según Cooper la primera vez que se formuló la técnica tal y como él la propone es en [Cooper, 99]. Tanto [Lombardi, 04] como [Mcmullin, 03] opinan que el uso de usuarios arquetípicos precede a la propuesta de llamarlo "Personas" por parte de Cooper. El término "Persona" tiene su origen en el teatro griego clásico, pues es la palabra que denominaba las máscaras que se utilizaban para representar sus papeles, y es un término que Carl Jung introdujo en la Psicología en su teoría de la personalidad [Jung, 1955]. Cuando usamos el término "Personas" en este trabajo de investigación nos referimos a la técnica IPO enunciada por Alan Cooper.

4.3.2 Proceso de Recopilación y Clasificación

Al comenzar el presente trabajo de investigación contábamos con encontrar en la literatura IPO una clasificación de técnicas según actividades en los términos que se manejan en la IS. Dicha clasificación sería el punto de partida del trabajo de investigación en cuanto a la integración de dichas técnicas y actividades con las propias de la IS. En la realización del estudio que hemos presentado en la sección 4.2 y el estudio que presentamos en esta sección 4.3, hemos podido comprobar que la información IPO no está estructurada como puede esperar un ingeniero software. Por esta razón, ha sido necesario llevar a cabo un trabajo de recopilación y clasificación de la información disponible sobre técnicas de usabilidad en las fuentes consideradas. La diversidad entre autores, en lo referente a actividades de construcción de software según la IPO, ha hecho que el estudio de actividades obligara a comparar entre sí las actividades de cada autor para obtener las similitudes y poder sacar un conjunto genérico de actividades. De todas formas, mientras el resultado de dicho estudio de actividades ha producido un conjunto de seis actividades, en el caso de las técnicas nos encontramos con un número mucho mayor de elementos a comparar.

En principio, la asignación de técnicas a actividades parece que permitiría reducir la complejidad del esfuerzo de clasificación. Sin embargo, algunos autores recomiendan la utilización de ciertas técnicas, sin dar detalles acerca de su ubicación en ninguna actividad concreta del desarrollo. Esta situación no permite contar siempre con la ayuda que una asignación de técnicas a actividades puede proporcionar. En la Figura 4.3, se representa de

forma aproximada la información sobre técnicas y actividades presente en la literatura consultada de forma gráfica. En esta figura, las distintas T_n representan algunas de las técnicas referenciadas en la literatura consultada, y las cajas denotadas A_n representan cuatro de las actividades obtenidas en el estudio de la sección 4.2. Como pretende representar la figura, se parte de una situación caótica con un gran número de técnicas no ubicadas y con distintos nombres para la misma técnica básica.

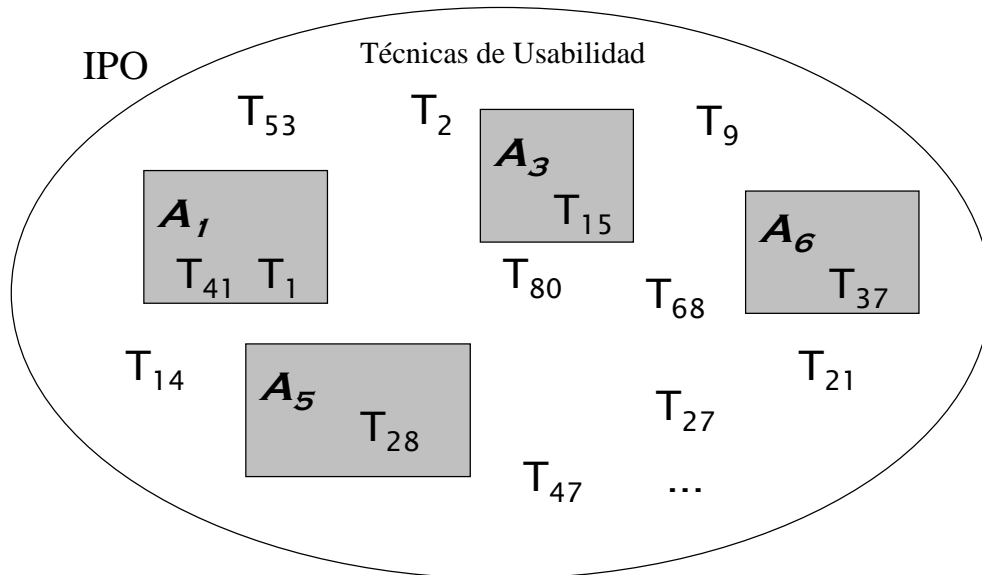


Figura 4.3 - Muestra del Conjunto de Técnicas de Usabilidad en la Literatura IPO

El primer paso para realizar el estudio ha consistido en la recopilación de todas las técnicas nombradas en las fuentes consultadas. Debido a la disparidad terminológica entre distintos autores, hemos realizado a continuación una agrupación de aquellas técnicas que tienen los mismos objetivos y características similares, esto es, que responden a la misma idea. Una vez agrupadas en torno a dicha idea de técnica básica, hemos asignado cada grupo de técnicas a la actividad del desarrollo centrado en el usuario en la que mejor encajan. Algunas de las fuentes ofrecían directamente una ubicación de la técnica en una actividad, pero la mayoría no. Para las técnicas que no estaban ubicadas, ha sido necesario valorar los objetivos de cada técnica y las consideraciones que el autor incluye sobre su aplicación, para poder asignarla a una actividad concreta. La Figura 4.4 representa, de forma gráfica, cómo cierto número de grupos de técnicas similares quedan asignadas a la actividad A_1 .

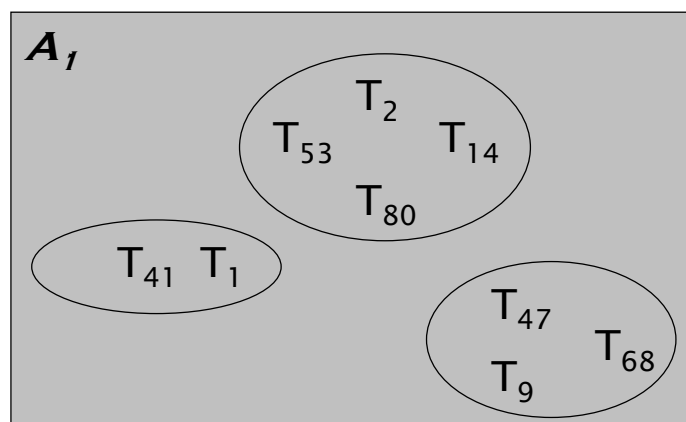


Figura 4.4 - Agrupación de Técnicas correspondientes a una Actividad

Para cada grupo de técnicas identificadas como la misma técnica básica ha sido necesario, para el objetivo de comunicar el resultado final de este trabajo a los desarrolladores,

identificar cuál es el nombre más representativo en el campo de la IPO, o bien el que mejor describe ese conjunto de técnicas desde el punto de vista de la IS. El nombre genérico escogido para cada técnica es el que usaremos en posteriores capítulos, y el que se comunicará a los desarrolladores en los resultados de este trabajo de investigación. Este último paso del proceso de clasificación está representado en la Figura 4.5.

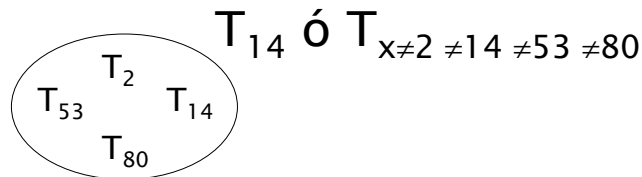


Figura 4.5 - Elección del Nombre Genérico para cada Grupo de Técnicas

4.3.3 Clasificación de las Técnicas

El resultado del trabajo de clasificación se recoge en varias tablas: Tres tablas detallan las técnicas relacionadas con actividades de análisis (Tabla 4.2, Tabla 4.3 y Tabla 4.4), dos tablas para las relacionadas con actividades de diseño (Tabla 4.5 y Tabla 4.6), y cuatro tablas adicionales para las técnicas relacionadas con la evaluación (Tabla 4.7, Tabla 4.8, Tabla 4.9 y Tabla 4.10). Cada tabla tiene una columna por cada fuente consultada. La columna más a la izquierda especifica las actividades en la que se han enmarcado las técnicas. La segunda columna indica los nombres genéricos escogidos para cada técnica. Cada fila corresponde a una técnica. Se ha considerado, así mismo, la posibilidad de variantes de una técnica, de forma que tanto la técnica genérica como cada una de las variantes tiene su propia fila (por ejemplo, en la Tabla 4.4 se muestra la técnica de *Card Sorting* junto con cuatro de sus variantes).

4.3.3.1 Técnicas Relacionadas con Actividades de Análisis

La Tabla 4.2, la Tabla 4.3 y la Tabla 4.4 recogen las técnicas relacionadas con las actividades de análisis. Como se muestra en la Tabla 4.2, mientras algunas técnicas están directamente relacionadas con el Análisis de Usuarios o de Tareas, otras técnicas relacionadas con la Especificación del Contexto de Uso no están restringidas únicamente a uno de los tipos de análisis, sino que afectan a la Especificación del Contexto de Uso en general, y por ello se muestran en la tabla en primer lugar.

En la Tabla 4.3 se recogen las técnicas destinadas al establecimiento de las Especificaciones de Usabilidad. Nótese que dichas técnicas están también estrechamente relacionadas con las actividades de Evaluación de Usabilidad, puesto que las Especificaciones de Usabilidad describen los objetivos contra los que se evalúa la usabilidad del sistema.

La Tabla 4.4 recoge la técnica de *Card Sorting*, junto a una serie de variantes de la misma. Esta técnica está relacionada con el análisis, con las tareas de educación de información de los usuarios, pero las fuentes no la ubican directamente en ninguna de las actividades concretas de análisis. El caso de la variante *Task Sorting* de esta técnica es distinto, puesto que [Mayhew, 99] sí presenta la técnica de *Task Sorting* como ubicada en la actividad de Análisis de Tareas. Por esta razón *Task Sorting* aparece tanto en la Tabla 4.4 (por ser variante de *Card Sorting*) como en la Tabla 4.2 (por aplicarse en la actividad de Análisis de Tareas).

Tabla 4.2 - Técnicas Relacionadas con la Especificación del Contexto de Uso

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99	
	Análisis Competitivo		Análisis Competitivo					
	Análisis de Impacto Financiero		Análisis de Impacto Financiero					
	Investigación Contextual	Investigación Contextual		Investigación Contextual			Entrevistas Contextuales	
	Diagramas de Afinidad						Diagramas de Afinidad	
	Observación Etnográfica			Etnografía	Observación Etnográfica			
	JEM ⁸					JEM		
	Análisis de Usuarios	Perfiles de Usuario	Perfiles de Usuario	Características de Usuario Individuales		Perfiles de Uso	Modelo Estructurado de Roles	Cuestionarios de Perfiles de Usuario
		Mapa de Roles de Usuario					Mapa de Roles de Usuario	
		Modelo Operacional					Modelo Operacional	Capacidades y Restricciones de Plataforma
		Personas	<i>Esta reciente técnica no se menciona en ninguna de las fuentes, pero se incluye por las razones detalladas más arriba en la sección 4.3.1.</i>					
	Análisis de Tareas	Casos de Uso Esenciales					Casos de Uso Esenciales	
HTA ⁹				HTA				
Familia de Modelos GOMS ¹⁰		GOMS		GOMS	GOMS			
		NGOMSL			NGOMSL	NGOMSL		
Modelo de Interfaz Objeto-Acción					Modelo de Interfaz Objeto-Acción			
Escenarios de Tareas							Escenarios de Tareas	
Task Sorting ¹¹						Task Sorting		

⁸ JEM: Modelado Esencial Conjunto (*Joint Essential Modeling*)

⁹ HTA: Análisis Jerárquico de Tareas (*Hierarchical Task Analysis*)

¹⁰ GOMS: Metas, Operadores, Métodos y Reglas de Selección (*Goals, Operations, Methods and Selection Rules*)

¹¹ La técnica *Task Sorting* se trata de una variante de la técnica *Card Sorting* descrita en la Tabla 4.4

Tabla 4.3 - Técnicas Relacionadas con las Especificaciones de Usabilidad

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99
Especificaciones de Usabilidad	Especificaciones de Usabilidad	Especificaciones de Usabilidad	Objetivos de Usabilidad	Especificaciones de Usabilidad			Objetivos de Usabilidad
	Objetivos de Rendimiento	Medidas Objetivas					Objetivos de Rendimiento
	Objetivos de Satisfacción	Medidas Subjetivas					Objetivos de Satisfacción
	Línea de Objetivos de Usabilidad		Línea de Objetivos de Usabilidad				
	Objetivos de Preferencia						Objetivos de Preferencia
	Objetivos Cualitativos						Objetivos Cualitativos

Tabla 4.4 - Técnicas Relacionadas con Análisis pero no con una Actividad Específica

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99
<i>Sin actividad específica</i>	<i>Card Sorting</i>		<i>Card Sorting</i>			<i>Card Sorting</i>	
	Agrupación por Afinidad					Agrupación por Afinidad	
	Ordenación por Criterio					Ordenación por Criterio	
	Voto por Umbral					Voto por Umbral	
	<i>Task Sorting</i>						<i>Task Sorting</i>

4.3.3.2 Técnicas Relacionadas con Actividades de Diseño

En la Tabla 4.5 se muestran las técnicas relacionadas con las actividades de diseño identificadas en el desarrollo centrado en el usuario de la IPO: Desarrollo del Concepto del Producto, Prototipado y Diseño de la Interacción.

Entre las técnicas relacionadas con el diseño encontramos una serie de técnicas que se refieren al modo de tomar decisiones de diseño, a la captura de la lógica tras las decisiones de diseño, o al diseño del subsistema de ayuda. Estas técnicas no están relacionadas en concreto con ninguna de las actividades de diseño identificadas en la sección 4.2.2, y están recogidas en la Tabla 4.6.

Tabla 4.5 - Técnicas Relacionadas con Actividades de Diseño

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99
Desarrollo del Concepto del Producto	Escenarios y <i>Storyboards</i>			Escenarios, <i>Storyboards</i> e Instantáneas	Escenarios	Escenarios y <i>Storyboards</i>	
	Tormenta de Ideas Visual			Tormenta de Ideas Visual			
Prototipado	Prototipado	Prototipado	Prototipado				
	Prototipos Escenario		Escenarios				
	Prototipos Activos			Animación de Requisitos		Prototipos Activos	Maquetas de Alta Fidelidad
	Prototipos de Papel			Prototipado de Baja Fidelidad		Prototipos Pasivos	Maquetas de Baja Fidelidad
	Prototipos Guiados			Prototipos Guiados			
	Prototipos Mago de Oz			Prototipos Mago de Oz			
Diseño de la Interacción	Representaciones de Pantallas	Escenarios y representaciones de pantallas					
	Guía de Estilo del Producto						Guía de Estilo del Producto
	Gramáticas				Gramáticas		
	UAN ¹²	UAN			UAN		
	TAG ¹³				TAG		
	Árboles de Menús				Árboles de Menús		
	Diagramas de Transición de Estados de la Interfaz	Diagrama de Transición de Estados de la Interfaz			Diagramas de Transición		
	Diagramas de Estados de Harel				Diagramas de Estados de Harel		
	Modelo del Contenido de la Interfaz					Modelo del Contenido de la Interfaz	
Mapa de Navegación					Mapa de Navegación entre Contextos		

4.3.3.3 Técnicas Relacionadas con Actividades de Evaluación

Las técnicas de Evaluación de Usabilidad se han organizado en cuatro tablas. En las tres primeras se recogen las técnicas que se agrupan en los tres principales tipos de Evaluación de Usabilidad: Evaluación por Expertos (Tabla 4.7), Test de Usabilidad (Tabla 4.8), y Estudios de Seguimiento de Sistemas Instalados (Tabla 4.9). En la Tabla 4.10 se detallan otras técnicas de Evaluación de Usabilidad que no encajan claramente dentro de ninguna de estas tres categorías.

¹² UAN: Notación de Acciones del Usuario (*User Action Notation*)

¹³ TAG: Gramáticas Tarea-Acción (*Task-Action Grammars*)

Tabla 4.6 - Técnicas Relacionadas con el Diseño pero no con una Actividad Específica

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99
Sin actividad específica	Diseño Integrador					Diseño Integrador (<i>Both-And Design</i>)	
	Diseño Paralelo		Diseño Paralelo				
	Análisis de Impacto	Análisis Coste / Importancia	Análisis de Impacto	Análisis de Impacto			
	Organización de la Ayuda según Casos de Uso					Organización de la Ayuda según Casos de Uso	
	IBIS ¹⁴ y PHI ¹⁵			IBIS y PHI			
	Análisis de Espacios de Diseño			Análisis de Espacios de Diseño			
	Análisis de Afirmaciones			Análisis de Afirmaciones			

Entre las técnicas de Evaluación por Expertos, recogidas en la Tabla 4.7, nótese que para Constantine y Lockwood [Constantine, 99], las Inspecciones de Conformidad cubren tanto las Inspecciones de Conformidad con Estándares como la Revisión de Guías.

Tabla 4.7 - Técnicas Relacionadas con la Evaluación por Expertos

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99	
Evaluación por Expertos	Evaluación Heurística	Evaluación Heurística	Evaluación Heurística	Evaluación Heurística	Evaluación Heurística	Evaluación Heurística	Evaluación Heurística	
	Inspecciones	de Conformidad con Estándares			Inspección de Estándares		Inspecciones de Conformidad	Inspecciones de Estándares
		Revisión de Guías				Revisión de Guías		Revisión de Guías
		de Consistencia			Inspección de Consistencia	Inspección de Consistencia	Inspección de Consistencia	Inspecciones de Consistencia
		Colaborativas					Inspecciones de Usabilidad Colaborativas	
	Recorrido (Walk-through)	Cognitivo			Recorrido Cognitivo	Recorrido Cognitivo	Recorrido Cognitivo	Recorridos Cognitivos
		Pluralístico		Recorrido Pluralístico	Recorrido Pluralístico		Recorrido de Usabilidad Pluralístico	Recorridos Pluralísticos

¹⁴ IBIS: Sistemas de Información Basados en Cuestiones (*Issue-Based Information Systems*)

¹⁵ PHI: Jerarquía Procedural de Cuestiones (*Procedural Hierarchy of Issues*)

La Tabla 4.8 recoge las técnicas que pueden aplicarse en la realización de Test de Usabilidad. En esta tabla se puede observar que se trata de las técnicas sobre las que existe mayor información en la IPO, puesto que la realización de test de usabilidad es una actividad clave en el desarrollo centrado en el usuario.

Tabla 4.8 - Técnicas Relacionadas con los Test de Usabilidad

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99	
Test de Usabilidad	Pensar en Voz Alta	Toma del Protocolo Verbal Concurrente	Pensar en Voz Alta	Protocolo de Pensar en Voz Alta		Háblame (pensar en voz alta)	Test Formales de Usabilidad (en las etapas iniciales)	
		Interacción Constructiva		Interacción Constructiva				
		Test Retrospectivo	Toma Retrospectiva del Protocolo Verbal	Test Retrospectivo	Protocolo Post-Evento		Reflexión Diferida	
		Toma de Incidentes Críticos	Toma de Incidentes Críticos					
		Método de Entrenamiento		Método de Entrenamiento				
	Medición del Rendimiento			Tareas de Referencia		Métricas de Rendimiento	Test Formales de Usabilidad (en etapas avanzadas)	
	Información Post-Test					Información Post-Test		
	Test de Usabilidad en Laboratorio	Test en Laboratorio	Laboratorios de Usabilidad		Test de Usabilidad y Laboratorios	Test en Laboratorio		
	Test de Campo	Test de Campo				Test de Campo		
	Grabación Vídeo	Grabación Vídeo	Grabación Vídeo	Grabación Vídeo				
	Grabación Audio	Grabación Audio		Protocolo Verbal				
	Registro del Uso	Instrumentación Interna de la Interfaz	Registro del Uso	Registro Software	Registro Continuo del Rendimiento del Usuario			
		Registro de Pulsaciones en el Tiempo		Registro de Pulsaciones en el Tiempo				
		Registro de la Interacción		Registro de la Interacción				
	Evaluación por Control Remoto						Evaluación por Control Remoto	
Test Remoto por Videoconferencia						Test Remoto por Videoconferencia		

La Tabla 4.9 detalla las técnicas de Evaluación de Usabilidad que pueden emplearse para el seguimiento de sistemas ya instalados. Nótese que una serie de técnicas de Registro del Uso aparecen tanto en esta tabla como en la Tabla 4.8, debido a que pueden emplearse tanto para un Test de Usabilidad tradicional como para obtener datos reales de interacción en el uso de un sistema instalado.

Tabla 4.9 - Técnicas Relacionadas con los Estudios de Seguimiento de Sistemas Instalados

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99
Estudios de Seguimiento de Sistemas Instalados	Observación Directa		Observación	Observación Directa			
	Observación Aleatoria						Observación Aleatoria
	Cuestionarios y Encuestas		Cuestionarios	Cuestionarios y Encuestas	Encuestas		
	Entrevistas		Entrevistas	Entrevistas	Entrevistas		
	Entrevistas Estructuradas	Entrevistas Estructuradas		Entrevistas Estructuradas			
	Entrevistas Flexibles			Entrevistas Flexibles			
	Focus Groups		Focus Groups		Discusiones Focus Groups		
	Registro del Uso	Instrumentación Interna de la Interfaz	Registro del Uso Real	Registro Software	Registro Continuo del Rendimiento del Usuario		Evaluación Remota Instrumentada
	Registro de Pulsaciones en el Tiempo			Registro de Pulsaciones en el Tiempo			
	Registro de la Interacción			Registro de la Interacción			
	Monitores Software de Activación Aleatoria						Monitores Software
	Retroalimentación del Usuario		Retroalimentación del Usuario		Buzón de Sugerencias o de Reporte de Errores en Línea		
	Servicios de Atención al Usuario en Línea				Operadores en Línea o Telefónicos		
	Foros				Newsgroups y Bulletin Boards		
	Revistas y Conferencias para Usuarios				Revistas y Conferencias para Usuarios		
	Evaluación Remota Semi-Instrumentada						Evaluación Remota Semi-Instrumentada

Finalmente, en la Tabla 4.10 se recogen otras técnicas de Evaluación de Usabilidad, las cuales no pertenecen a ninguna de las tres categorías principales de técnicas de Evaluación de Usabilidad.

Tabla 4.10 - Técnicas de Evaluación de la Usabilidad que no Pertenecen a Ninguno de las Tres Categorías Principales

ACTIVIDAD	TÉCNICA	Hix, 93	Nielsen, 93	Preece, 94	Shneiderman, 98	Constantine, 99	Mayhew, 99	
<i>Sin actividad específica</i>	Test Experimentales			Experimentos Tradicionales	Experimentos Controlados Orientados Psicológicamente			
	Métricas Predictivas	Procedurales					Métricas Predictivas Procedurales	
		Modelo a Nivel de Pulsaciones			Modelo a Nivel de Pulsaciones			
		Estructurales					Métricas Predictivas Estructurales	
		Semánticas					Métricas Predictivas Semánticas	
	Evaluación Cooperativa			Evaluación Cooperativa				

4.4 Caracterización de Técnicas

Tenemos un total de 94 técnicas (incluyendo las variantes de cada técnica general) obtenidas en el estudio detallado en la sección anterior. Este número de técnicas resulta, definitivamente, poco manejable para un ingeniero software, especialmente si se considera que pertenecen a un campo ajeno a la IS. Una oferta tan amplia de técnicas puede llevar a un problema de saturación, con el riesgo de provocar la renuncia al objetivo de integrar actividades y técnicas de usabilidad en el proceso de desarrollo. Con el fin de facilitar el trabajo al ingeniero software, vamos a detallar para cada técnica del estudio su grado de utilidad para el objetivo general de integración desde una perspectiva de la IS, según unos criterios específicos. Es decir, en cierta manera vamos a priorizar el universo actual de 94 técnicas de modo que el ingeniero software pueda fácilmente elegir aplicar las primeras en la asignación de prioridad que proponemos.

4.4.1 Criterios de Caracterización

Para poder establecer la prioridad entre técnicas de un forma ordenada, vamos a valorar cada técnica según un conjunto de criterios que vamos a describir a continuación. El valor asignado a cada criterio para cada técnica se va a recoger en una serie de tablas.

- **Participación de los Usuarios:** Uno de los puntos básicos de un proceso centrado en el usuario, como se indicó en secciones anteriores, es la implicación activa de los futuros usuarios del sistema. Algunas de las técnicas de usabilidad están destinadas específicamente a favorecer dicha implicación, y vamos a destacarlas en nuestro estudio con un valor positivo para este criterio. En las tablas se indica en la columna "P.U." con los valores *sí* o *no*, si la técnica está destinada a la implicación activa del usuario.

- **Necesidad de Formación:** Este criterio se refiere a cuánta formación necesita un ingeniero software medio para poder aplicar la técnica con mínimas garantías de éxito. Una necesidad de formación con valor *muy alto* indica que se requiere una experiencia en usabilidad extensa para poder aplicarse, esto es, que se requiere personal experto en la materia para su aplicación. El valor *alto* indica que su aplicación requeriría al menos un perfil combinado IS-IPO, en el sentido de que el ingeniero software que pudiera aplicarla sería aquel que hubiera recibido una formación extensa en usabilidad. Un valor *medio* indica que, si bien se requiere una formación de cierta importancia, puede ser aplicada por ingenieros software medios con dicha formación. El valor *bajo* indica que únicamente es necesaria formación básica para poder ser aplicada por ingenieros software medios.
- **Aplicabilidad General:** Este criterio refleja la generalidad de la técnica, esto es, cuánto de aplicable es a un abanico amplio de tipos de proyectos de construcción de software. Una valor de aplicabilidad *alto* indica que su uso puede resultar de utilidad prácticamente en todo tipo de proyectos. Si se tiene un valor *medio* para este criterio, quiere decir que la técnica es aplicable en ciertos tipos de proyectos, pero en otros tipos no. Finalmente, un valor *bajo* en este criterio indica que únicamente en tipos de proyectos específicos, los cuales forman un porcentaje escaso entre todos los tipos de proyectos, son adecuados para su aplicación. La columna correspondiente a este criterio en las tablas está etiquetada como "Aplicabilidad".
- **Cercanía a la IS:** Este criterio refleja si los principios en los que se basa la técnica coinciden con principios y enfoques presentes habitualmente en la IS. Un valor *alto* en este criterio indica que la técnica puede ser fácilmente comprendida por ingenieros software, puesto que se basa en habilidades y enfoques típicos de la IS. Un valor *medio* indica que, si bien la técnica se basa en principios que no pertenecen a la IS, no está tan alejada de la misma como para considerarla completamente ajena al campo. Un valor *bajo* en este criterio, por el contrario, indica que la técnica requiere un enfoque de desarrollo y unas habilidades que son ajenas a los que suele tener una persona con formación en IS.
- **Grado de Aportación de Usabilidad Frente al Esfuerzo:** Este criterio se refiere a cuánta mejora en la usabilidad del producto final puede aportar el uso de la técnica, comparada con el esfuerzo que supone su aplicación. Puesto que los recursos para usabilidad pueden ser escasos en muchos proyectos de desarrollo de software, consideramos que la información de tipo coste-beneficio puede resultar de gran utilidad para la selección de las técnicas a aplicar. Los valores posibles para este criterio son *alto*, *medio* y *bajo*, reflejando la importancia de la aportación de usabilidad que puede resultar de la aplicación de la técnica, en relación al esfuerzo que tiene que invertir el equipo de desarrollo en su aplicación. La columna correspondiente a este criterio en las tablas está etiquetada "Aportación/Esfuerzo".
- **Representatividad:** Este criterio refleja cuán comúnmente se aplica la técnica en el campo de la IPO. Como indicador de este criterio vamos a utilizar el número de autores del estudio que aconsejan la aplicación de la técnica, por tanto el valor de este atributo es numérico y está en el rango 1-6. En el caso de técnicas para las cuales se han definido variantes, se considera el valor agregado de todos los autores que, o bien describen la técnica general, o bien describen alguna de sus variantes. Por ejemplo, se puede observar en la Tabla 4.9 que la técnica de Observación Directa es nombrada por [Nielsen, 93] y [Preece, 94], y también en su variante de Observación Aleatoria por [Mayhew, 99]. Así, su representatividad tiene un valor de tres.

Es necesario precisar que los valores que se indican para cada uno de estos criterios son orientativos. Se han asignado basándose en una perspectiva de IS, y pretenden facilitar la

tarea de integración al permitir reducir el número de técnicas a considerar por parte del ingeniero software. Puesto que el objetivo es realizar una selección entre las técnicas, se ha resumido el valor combinado de cada criterio en un solo valor denominado **Valoración Total**, el cual puede tomar los siguientes valores: *Muy útil*, *útil*, y *poco útil*. Los criterios para asignar cada valor son los siguientes:

- **Muy útil:** Aquellas técnicas que son especialmente útiles para nuestro propósito tienen las siguientes características:
 - Su grado de aportación de usabilidad frente al esfuerzo invertido es alto, puesto que nos interesan especialmente las técnicas que producen mayores réditos de usabilidad con el menor esfuerzo.
 - No requieren un nivel de formación alto, pues así serán más fáciles de introducir en organizaciones con poca experiencia previa en técnicas IPO.
 - Tienen un margen de aplicabilidad medio o alto, porque interesa que tengan un ámbito de aplicación suficientemente amplio.
 - O bien tienen un grado de cercanía a la IS alto, o bien son técnicas que se aplican comúnmente en el campo de la IPO (representatividad mayor o igual que tres). En el primer caso su introducción resulta más sencilla en una organización con poca experiencia IPO, mientras que en el segundo se trata de técnicas cuyo valor probado motiva su recomendación por diversas fuentes de las empleadas en el estudio.
- **Útil:** Mientras las técnicas que cumplen las condiciones anteriores son las más destacadas para el objetivo que perseguimos, otras técnicas también pueden resultar de utilidad. Con esta valoración hemos destacado las técnicas que cumplen alguna de las siguientes condiciones:
 - El grado de aportación de usabilidad frente al esfuerzo es alto, puesto que consideramos que las técnicas con dicho valor son útiles independientemente del nivel en otros criterios
 - El grado de aportación frente a esfuerzo es medio y además se cumple alguna de las siguientes condiciones:
 - La aplicabilidad es media o alta, puesto que la solución a presentar a los ingenieros software se desea que tenga un cierto grado de generalidad, y que sea lo suficientemente limitada en el número total de técnicas como para resultar útil como herramienta.
 - Es una técnica que fomenta la participación de los usuarios y cumple las condiciones que se detallan a continuación. Puesto que la participación de usuarios es una característica central de los procesos centrados en el usuario y no es común en la IS, hemos relajado las condiciones de este tipo de técnicas para ser consideradas útiles. Así, aunque la aplicabilidad de la técnica sea baja, la consideramos útil si su representatividad es mayor o igual que tres, puesto que entonces se trata de una técnica habitual en los desarrollos centrados en el usuario.
- **Poco útil:** Las técnicas que no cumplen los requisitos para pertenecer a ninguna de las categorías anteriores se han etiquetado como poco útiles en la valoración total. A pesar de que pueden resultar de utilidad en desarrollos concretos, consideramos que su posible utilidad en casos limitados no compensa su inclusión en el esquema general de integración, puesto que la exhaustividad en el reflejo de todo el campo de la IPO puede actuar como elemento disuasorio para los ingenieros software, debido a una innecesaria complejidad de la propuesta.

Los valores de los criterios para cada una de las técnicas del estudio se muestran en varias tablas que se describen en las siguientes secciones.

4.4.2 Caracterización de Técnicas Relacionadas con Análisis

En la Tabla 4.11 se recogen los valores para las técnicas relacionadas con análisis, según se detalla a continuación:

- **Investigación Contextual** y **Observación Etnográfica** son muy similares, en cuanto a que requieren una formación alta, no son aplicables en todo tipo de proyectos (valor medio en aplicabilidad general) por requerir una disponibilidad alta en la organización cliente, y están en un nivel medio de cercanía a la IS. En cuanto al grado aportación/esfuerzo, la Investigación Contextual establece un diálogo con los usuarios que puede resultar muy útil, en cuanto aporta información que se puede emplear en el diseño de un sistema más usable, y requiere emplear menos tiempo que cuando se realiza una Observación Etnográfica, en la que se observa de forma pasiva. Por esta razón, la Observación Etnográfica tiene un valor medio de aportación de usabilidad, mientras que la Investigación Contextual tiene un valor alto.
- **Diagramas de Afinidad** y **Card Sorting** son también técnicas similares en cuanto a que ambas son sencillas de aplicar, por lo que su necesidad de formación es baja. Son útiles en una gran variedad de proyectos (aplicabilidad general alta), tienen un nivel medio de cercanía a la IS, pues son técnicas participativas que no suelen realizarse por ingenieros software. Ambas tienen un grado de aporte frente al esfuerzo alto, puesto que su coste de aplicación es muy bajo, y, sin embargo, sí ofrecen la posibilidad de una importante mejora en la usabilidad del producto final mediante su aplicación.
- Las técnicas de modelado de tareas **Casos de Uso Esenciales** y **HTA** tienen una necesidad de formación media, puesto que, a pesar de tratarse de notaciones relativamente simples, requieren una forma de pensar en el problema a modelar que requiere cierto entrenamiento para poder realizarse con perspectivas de éxito. Los Casos de Uso Esenciales son aplicables a todo tipo de proyectos, puesto que las tareas del usuario se deben considerar en todo proyecto (aplicabilidad alta), debido a que se trata de una de las bases del enfoque centrado en el usuario, pero la técnica HTA se centra en copiar la forma de realizar las tareas que se realiza actualmente, por lo que requiere que exista un sistema actual (aplicabilidad media). Puesto que las tareas de modelado son comunes en la IS, tienen un valor alto de cercanía a la IS. En cuanto al grado de aportación frente al esfuerzo, los Casos de Uso Esenciales tienen cierta dificultad en su aplicación y refinamiento, pero la mejora que pueden aportar a la usabilidad del sistema es alta, por lo que su valor es alto. HTA requiere algo más de esfuerzo de aplicación, porque tiene un proceso detallado de aplicación, y aunque su contribución a la mejora de la usabilidad del producto final es importante, el conjunto de ambos factores hace que su valor para este criterio sea medio.
- Las técnicas de modelado de usuarios (**Perfiles de Usuario**, **Mapa de Roles de Usuario** y **Modelo Operacional**) son cercanas a las IS, puesto que son tareas de especificación y modelado. En cuanto a la necesidad de formación, mientras es baja para el Mapa de Roles de Usuario, puesto que se trata de una técnica muy sencilla, los Perfiles de Usuario y el Modelo Operacional requieren un conocimiento alto de los principios de usabilidad, para poder comprender qué información puede ser o no ser relevante para la usabilidad del sistema final, por tanto el valor para este criterio es alto para ambas técnicas. Los Perfiles de Usuario son necesarios en cualquier proyecto, puesto que el conocimiento de los futuros usuarios del sistema es una de las

Tabla 4.11 - Valoración de Técnicas Relacionadas con Análisis

Técnica	P.U.	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Representatividad	Val. Total
<i>Card Sorting</i>	sí	bajo	alto	medio	alto	3	Muy útil
Casos de Uso Esenciales	no	medio	alto	alto	alto	1	Muy útil
Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	Muy útil
Personas	no	medio	medio	medio	alto	3	Muy útil
Análisis Competitivo	no	medio	alto	medio	alto	1	Útil
Diagramas de Afinidad	sí	bajo	alto	medio	alto	1	Útil
Escenarios de Tareas	sí	medio	medio	medio	alto	1	Útil
HTA	no	medio	medio	alto	medio	1	Útil
Investigación Contextual	sí	alto	medio	medio	alto	3	Útil
JEM	sí	medio	medio	alto	medio	1	Útil
Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	Útil
Observación Etnográfica	no	alto	medio	medio	medio	2	Útil
Perfiles de Usuario	no	alto	alto	alto	alto	5	Útil
Análisis de Impacto Financiero	no	alto	medio	alto	bajo	1	Poco útil
Familia de Modelos GOMS	no	muy alto	bajo	bajo	bajo	3	Poco útil
Modelo de Interfaz Objeto-Acción	no	alto	bajo	medio	bajo	1	Poco útil
Modelo Operacional	no	alto	bajo	alto	medio	2	Poco útil

bases del enfoque centrado en el usuario, por esta razón su aplicabilidad general es alta. Los Mapas de Roles de Usuario sólo son aplicables cuando el número de roles distintos es alto y se pueden establecer relaciones entre ellos, por lo que su aplicabilidad general es media. El Modelo Operacional sólo es necesario como técnica aparte cuando el sistema a desarrollar tiene características del entorno de uso que son especialmente relevantes para la usabilidad del sistema final, por lo que su uso está circunscrito a proyectos con características especiales y, así, su valor de aplicabilidad general es bajo. Puesto que los Perfiles de Usuario son básicos para la usabilidad del sistema, aunque requieren cierto grado de esfuerzo su grado aportación/esfuerzo es alto. Por el contrario, el Mapa de Roles de Usuario no es costoso de aplicar, pero el aporte en usabilidad es reducido, por lo que su valor para este criterio es medio. Finalmente, el Modelo Operacional requiere cierto esfuerzo de aplicación, pero en los casos en los que es aplicable contribuye de un modo apreciable a la usabilidad, por lo que su valor es medio.

- Las técnicas formales para el análisis de tareas y las acciones relacionadas (**Modelos GOMS e Interfaz Objeto-Acción**) tienen en común el requerir una formación extensa (el valor es muy alto para GOMS, y alto para el Modelo de Interfaz Objeto-Acción por ser algo más sencillo al no incluir aspectos cognitivos). Su aplicabilidad general es baja porque para sistemas con una IU de complejidad media-alta no son practicables, ya que el esfuerzo de formalización de todas las posibles acciones sería ingente. Los modelos tipo GOMS no son cercanos a la IS porque incluyen aspectos cognitivos, los cuales no están entre las preocupaciones clásicas de la IS, por tanto tienen un valor bajo en este criterio. El Modelo de Interfaz Objeto-Acción es más cercano al modelado habitualmente realizado en la IS, pero aún así no es tan cercano puesto que está enfocado a interfaces directamente manipulables, por lo que su valor es medio. Como cualquier técnica formal, estas tres técnicas tienen un esfuerzo de aplicación muy alto, mientras que las mejoras en la usabilidad del sistema final no son tan grandes, por lo que su nivel aportación/esfuerzo es bajo.
- La técnica **JEM** requiere cierta formación al tratarse de un proceso detallado para modelar, por lo que su necesidad de formación es media. Es aplicable en casos específicos pero bastante generales, puesto que es apropiada para cuando se quiere conseguir que todas las partes involucradas sientan que su voz cuenta. Así, su aplicabilidad general es media. Es cercana a la IS (valor alto) porque se basa en la técnica JAD (*Joint Application Design*), una técnica de la IS relacionada con el proceso de requisitos. Tiene un coste considerable de aplicación por tratarse de reuniones multidisciplinarias, pero el aporte en usabilidad es importante, quedando la relación aportación/esfuerzo en un grado medio.
- Los **Escenarios de Tareas** y la técnica **Personas** se aplican cuando el rango de usuarios y tareas reales es amplio, y es útil abstraer posibles usuarios (Personas) y/o sus tareas (Escenarios de Tareas), que aunque no correspondan con ningún usuario real sean representativas de cómo es un usuario medio y cómo es una tarea típica de las que realiza. Así, su aplicabilidad general es media, porque no en todos los proyectos es necesario abordar el desarrollo de esta forma. Ambas son cercanas a la IS en cuanto a su objetivo de abstraer, pero la forma de elaborar Personas es más similar a lo que se hace en los estudios de mercado que en los procesos de la IS, y los Escenarios de Tareas tampoco responden a un planteamiento clásico de la IS, por lo que ambas técnicas tienen un valor medio para este criterio. Ambas técnicas requieren una formación de cierta consideración, por lo que su necesidad en este sentido es media. La técnica Personas requiere un esfuerzo considerable de aplicación, pero su aporte en usabilidad puede ser muy importante, y por tanto tiene un grado de aportación/esfuerzo alto, mientras que los Escenarios de Tareas no ofrecen tanto

retorno en usabilidad, por lo que su valor para este criterio es medio. En cuanto a la técnica Personas, su valor de Representatividad sería cero puesto que no está nombrada por ninguna de las fuentes consideradas, como se indicó más arriba, sin embargo, debido al amplio uso que tiene en la actualidad, hemos considerado oportuno otorgar a la técnica un valor tres de representatividad.

- El **Análisis Competitivo** requiere cierta formación porque es necesario aprender qué hay que observar en un producto competidor, y cómo obtener sus puntos fuertes y débiles en usabilidad, por lo que su necesidad de formación es media. Es aplicable a todo tipo de proyectos, incluso en sistemas innovadores, pues el análisis de productos con los que puede haber alguna similitud puede ser útil para imitar aquello que funciona bien. Así, su aplicabilidad general es alta. La evaluación de sistemas es algo conocido en la IS, pero el analizar productos competidores no es tan habitual, por lo que el valor de cercanía a la IS es medio. Aunque en algunos casos el esfuerzo en la aplicación de esta técnica puede ser considerable, los beneficios en la usabilidad del sistema final cubren con creces el coste, por lo que el grado aportación/esfuerzo es alto.
- El **Análisis de Impacto Financiero** requiere una formación en usabilidad considerable, por lo que su valor en este criterio es alto. Es aplicable siempre que la eficiencia en uso sea un atributo de importancia considerable, pero en otros casos no es tan útil, por lo que su aplicabilidad general es media. Los estudios de retorno a la inversión no son ajenos a la IS, pero al centrarse en usabilidad se manejan variables no habituales en la IS, por lo que su cercanía a la IS es media. No aporta casi a la usabilidad del sistema final, puesto que la principal finalidad de la técnica es la decisión de cuánto esfuerzo en usabilidad es necesario incluir en el proyecto (grado aportación/esfuerzo bajo).
- Establecer las **Especificaciones de Usabilidad** requiere una formación y experiencia en usabilidad considerables, por lo que implica de conocimiento de qué objetivos son alcanzables con un coste razonable, por esta razón la técnica tiene un valor alto de necesidad de formación. Esta técnica es aplicable a aquellos problemas en los que sea factible conocer al principio del desarrollo las tareas que se quieren soportar, en tanto el sistema no se base en tareas de un usuario en un entorno de oficina, la técnica no es tan válida. Por tanto, en el criterio de aplicabilidad general tiene un valor medio. Las Especificaciones de Usabilidad, en cuanto requisitos no-funcionales, son cercanas a la IS, pero incluyen al usuario (humano) en los objetivos a establecer. Así, los conocimientos requeridos para el establecimiento de las especificaciones son muy distintos a los que maneja habitualmente un ingeniero software, por tanto, al tener estas dos vertientes contradictorias consideramos que el valor de cercanía a la IS es medio. Finalmente, el grado de aportación/esfuerzo es alto en cuanto son una herramienta muy útil para conocer el nivel de usabilidad que se desea alcanzar, y poder establecer cuán lejos se está de dicho nivel, a pesar de que el esfuerzo que supone establecerlas sí puede ser alto.

4.4.3 Caracterización de Técnicas Relacionadas con Diseño

Entre las técnicas relacionadas con diseño no se ha incluido el Prototipado como técnica general, ni sus variantes de Prototipos Escenario y Prototipos Activos, puesto que, a pesar de estar descritos en la literatura IPO, son conocidos como Maquetas o Prototipos Ejecutables en la IS. Tampoco se han incluido los Prototipos Guiados, pues son las "demos" habituales de la IS. Sí se han incluido las variantes de prototipado siguientes: Prototipos de Papel y del Mago de Oz, puesto que son específicas de la IPO y en la IS no se aplican habitualmente. Para las técnicas que capturan la lógica del diseño (IBIS, PHI, Análisis de Espacios de Diseño y

Análisis de Afirmaciones) no se puede considerar su aportación a la usabilidad de forma particular, ya que su objetivo es capturar cualquier tipo de decisión de diseño. Por tanto, no se han incluido.

En la Tabla 4.12 se recogen los valores para las técnicas relacionadas con diseño, según se detalla a continuación:

- Las técnicas de **Escenarios** y **Storyboards** no son cercanas a la IS (valor bajo), pues requieren imaginar el futuro sistema y su contexto a un nivel detallado para un caso concreto, lo cual no suele realizarse en la IS. Así mismo, esto hace que la necesidad de formación sea media. Los Escenarios y *Storyboards* son aplicables cuando el sistema tiene un carácter innovativo, o bien cuando las partes involucradas (principalmente el cliente) no sabe expresar con precisión qué es lo que quiere. Aunque un gran número de proyectos tendrán alguna de estas características, muchos otros no, por tanto el valor de aplicabilidad general es medio. En los casos en que es aplicable sí que produce una mejora en la usabilidad del producto final, que podemos calificar de alta en relación con el esfuerzo invertido.
- La técnica de **Tormenta de Ideas Visual** no es una técnica compleja, por lo que la necesidad de formación es baja, es aplicable a todo tipo de problemas para las reuniones iniciales que dan forma al proyecto (aplicabilidad general alta), y no es cercana a la IS porque no se suele trabajar de este modo. Se trata de una técnica barata en cuanto a esfuerzo de aplicación, que consigue una contribución importante en cuanto a mejora de la usabilidad del producto final (grado de aportación frente a esfuerzo alto).
- La técnica de **Guía de Estilo del Producto** podemos considerarla como de una cercanía de nivel medio a la IS, puesto que por una parte responde a la necesidad de especificar las reglas por las que se va a guiar el diseño (objetivo que no resulta ajeno a la IS), mientras que por otra parte, dichas reglas se ocupan de los elementos que forman parte de la IU, cuyo diseño no es parte de la IS según el SWEBOK [SWEBOK, 04]. Por esta razón, consideramos que tiene un valor medio en cuanto a cercanía a la IS. La elaboración de una Guía de Estilo del Producto requiere una experiencia amplia en temas de usabilidad, por lo que consideramos que la necesidad de formación es alta. Una Guía de Estilo del Producto sólo se justifica en sistemas de cierta complejidad, especialmente cuando se trata de una familia de productos, por lo que su aplicabilidad general es media. Se trata de una técnica muy costosa en esfuerzo, y lo obtenido en cuanto a mejora de usabilidad es importante, pero no tanto como lo obtenido con otras técnicas, por lo que su grado de aportación/esfuerzo es medio.
- Los dos tipos de **Prototipos (de Papel, y Mago de Oz)** son técnicas sencillas, que no requieren excesiva formación (nivel bajo). Los Prototipos de Papel son aplicables en todo tipo de proyectos (nivel de aplicabilidad alto), mientras que los Mago de Oz son útiles únicamente para proyectos en los que el comportamiento previsto del sistema puede ser realizado fácilmente por una persona, frente al coste de programar dicho comportamiento, por lo que su aplicabilidad general es baja. Los Prototipos de Papel están cercanos a la IS (nivel alto), pues son análogos a algunas maquetas que se realizan para servir a modo de "demo" para el cliente. Por el contrario, cuando estos prototipos se utilizan para realizar test de usabilidad, como es el caso de los Mago de Oz, ya no resultan tan cercanos a la IS (nivel medio). Los Prototipos de Papel requieren muy poco esfuerzo de elaboración, y son una técnica que contribuye de forma muy importante a la usabilidad del producto final, por lo que su grado de aportación frente a esfuerzo es alto. Por el contrario, los Prototipos Mago de Oz requieren un esfuerzo considerable, pues el objetivo es evitar que el usuario sospeche

Tabla 4.12 - Valoración de Técnicas Relacionadas con Diseño

Técnica	P.U.	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Representatividad	Val. Total
Árboles de Menús	no	bajo	medio	alto	alto	1	Muy útil
Escenarios y <i>storyboards</i>	sí	medio	medio	bajo	alto	3	Muy útil
Prototipos de Papel	sí	bajo	alto	alto	alto	3	Muy útil
Análisis de Impacto	no	medio	medio	alto	medio	3	Útil
Diagramas de Transición de Estados de la Interfaz	no	bajo	alto	alto	medio	2	Útil
Guía de Estilo del Producto	no	alto	medio	medio	medio	1	Útil
Mapa de Navegación	no	medio	alto	alto	medio	1	Útil
Modelo del Contenido de la Interfaz	no	medio	alto	medio	medio	1	Útil
Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	Útil
Tormenta de Ideas Visual	sí	bajo	alto	bajo	alto	1	Útil
Diagramas de Estados de Harel	no	medio	bajo	alto	bajo	1	Poco útil
Diseño Integrador	no	medio	medio	alto	bajo	1	Poco útil
Diseño Paralelo	no	bajo	medio	alto	bajo	1	Poco útil
Gramáticas	no	medio	bajo	alto	medio	1	Poco útil
Prototipos del Mago de Oz	sí	bajo	bajo	medio	medio	1	Poco útil
Representaciones de Pantallas	no	medio	bajo	alto	bajo	1	Poco útil
TAG	no	muy alto	bajo	medio	bajo	1	Poco útil
UAN	no	muy alto	bajo	medio	bajo	2	Poco útil

que no está trabajando con un sistema real, así su valor para este criterio (aportación/esfuerzo) es medio.

- La técnica de **Representaciones de Pantallas** se trata de una especificación de los contenidos de la parte visible de la IU, lo cual es un tipo de actividad muy cercano a los típicos de la IS (valor alto). Tiene una necesidad de formación media, puesto que el diseño de la IU no es algo habitual entre los ingenieros software y la aplicación de esta técnica requeriría una formación extensa en usabilidad. Su aplicabilidad general es baja, puesto que en la mayoría de los proyectos de desarrollo software no vale la pena especificar la IU, ya que el esfuerzo de especificar es el mismo que el de crear directamente el producto final. Únicamente es recomendable cuando se necesita especificar con mucho detalle al equipo de diseño de la IU cuál es su papel, porque un equipo multidisciplinar e integrado no sea posible (por lejanía geográfica, por ejemplo), por tanto, tiene una aplicabilidad general baja. El esfuerzo de especificar mediante Representaciones de Pantallas es alto, y la mejora en usabilidad no es especialmente destacable por lo que el grado de aportación frente al esfuerzo es bajo.
- La técnica de **Gramáticas** es aplicable únicamente a IUs basadas en línea de comandos, las cuales conforman un porcentaje muy bajo de las IUs que se diseñan en la actualidad, por lo que el valor de aplicabilidad general es bajo. La técnica es relativamente compleja, por lo que requiere un grado medio de formación. El tipo de técnica es cercano a las técnicas de la IS (valor alto), por su alto grado de formalidad. Finalmente, su grado de aportación frente a esfuerzo es medio, puesto que tiene un coste elevado y las mejoras en usabilidad son destacables, pero no tanto como sucede con otras técnicas.
- Las técnicas **UAN** y **TAG** son ambas muy complejas, por lo que tienen una necesidad de formación muy alta. En cuanto a cercanía a la IS, podemos considerar que tienen un valor medio, puesto que, a pesar de su alto grado de formalidad, tratan con conceptos mentales que no son comúnmente abordados por la IS. Su aplicabilidad general es baja, al igual que se mencionó más arriba para la familia de modelos GOMS, puesto que únicamente en proyectos donde interesa tener todos los comportamientos formalmente especificados se van a aplicar estas técnicas, debido a su alto coste. La aplicación de las técnicas produce ciertas mejoras en usabilidad, pero el esfuerzo que requieren es muy considerable, por lo que el grado de aportación frente a esfuerzo es bajo.
- Los **Árboles de Menús** constituyen una técnica de especificación relativamente sencilla (su necesidad de formación es baja), que tiene una aplicabilidad general alta, pues son muy comunes las IUs con menús. Resulta cercana a la IS (valor alto), puesto que es similar a otras técnicas de modelado pertenecientes a la IS. Debido a la sencillez de la técnica su aplicación no resulta costosa, y los beneficios de usabilidad sí pueden resultar de importancia, por lo que su grado de aportación frente a esfuerzo es alto.
- El resto de técnicas de modelado de la interacción (**Diagramas de Transición de Estados de la Interfaz**, **Diagramas de Estados de Harel**, **Modelo del Contenido de la Interfaz**, y **Mapa de Navegación**) tienen una necesidad de formación media debido a su relativa complejidad, excepto los Diagramas de Transición que son más simples y tienen una necesidad baja. Su aplicabilidad general es alta, pues se basan en describir la transición en una IU gráfica, el tipo de IU más común en la actualidad. Únicamente los Diagramas de Estados de Harel tienen una aplicabilidad baja, puesto que están especialmente destinados a reflejar temas de concurrencia y sincronización, los cuales no suelen ser problemas destacados en la mayoría de los casos.

Consideramos que se trata de técnicas cercanas a la IS (valor alto) pues, tal y como se ha comentado para los Árboles de Menús, el modelado es una actividad típica de la IS. La excepción está en el Modelo del Contenido de la Interfaz, que por tratarse de modelos que utilizan medios no habituales en la IS (como notas Post-It), tiene un valor medio de cercanía a la IS. El grado de aportación frente al esfuerzo es bajo para los Diagramas de Estados de Harel, puesto que se trata de la técnica más compleja de aplicar, y la mejora en usabilidad que puede ofrecer no es muy alta. El resto de técnicas tienen un valor medio, puesto que son algo costosas, pero sí ofrecen una mejora apreciable de la usabilidad del producto final.

- Las técnicas de **Diseño Integrador** y **Diseño Paralelo** son de aplicación para proyectos en los que se requiere un cierto grado de creatividad, cuando es necesario probar distintos diseños y encontrar la mejor solución. Se trata de una situación común a muchos proyectos, pero no a todos, por lo tanto su aplicabilidad general es media. Contrastar distintos diseños es una práctica que surge habitualmente en la IS, por tanto la cercanía a la misma es alta. En cuanto a la necesidad de formación, el Diseño Paralelo no se trata de una técnica elaborada, por lo que su nivel es bajo, mientras que el Diseño Integrador sí conlleva un proceso específico que hay que aprender (nivel medio). Ambas estrategias de diseño consumen bastantes recursos, puesto que se tiene más de un equipo de diseño trabajando por separado hacia el mismo fin, y aunque pueden contribuir a la usabilidad del producto final, el grado de aportación frente a esfuerzo es bajo, puesto que no se trata tanto de las mejoras en usabilidad que se consiguen como la obtención de una decisión de diseño adecuada.
- El **Análisis de Impacto** resulta similar al manejo de técnicas de estimación de la IS, por lo que el nivel de cercanía a la misma es alto. La técnica requiere cierta formación sin ser excesiva, por lo que el valor para este criterio es medio. La técnica puede aplicarse cuando es posible calcular impacto y costes de cada par problema/solución, lo cual no puede hacerse en todos los casos (aplicabilidad general media). El esfuerzo necesario para realizar los cálculos del Análisis de Impacto es alto, y los beneficios de usabilidad están a un nivel similar, por lo que el grado aportación/esfuerzo es medio.
- La técnica de **Organización de los Casos de Uso según la Ayuda** requiere cierta formación (necesidad de formación media), y puede aplicarse en sistemas en los que se pueden extraer un número limitado de casos de uso representativos (aplicabilidad general media). Al basarse esta técnica en los casos de uso, tiene una cierta cercanía con la IS (nivel alto) y, aunque no resulta excesivamente costosa su aplicación, los beneficios están al mismo nivel (grado de aportación/esfuerzo medio).

4.4.4 Caracterización de Técnicas Relacionadas con Evaluación

Respecto a las técnicas relacionadas con la Evaluación de Usabilidad, no se han incluido la Evaluación por Control Remoto ni el Test Remoto por Videoconferencia, puesto que se trata de técnicas muy similares al Test en Laboratorio en cuanto a los criterios que se están considerando. Así mismo, la Grabación Audio y la Grabación Vídeo se han considerado de forma conjunta por la misma razón. Las Entrevistas son similares a los Cuestionarios y Encuestas en cuanto a las características aquí observadas, por lo que se ha integrado con los mismos.

Los valores para cada técnica relacionada con la evaluación se detallan en la Tabla 4.13, según se explica a continuación:

Tabla 4.13 - Valoración de las Técnicas Relacionadas con Evaluación

Técnica	P.U.	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Representatividad	Val. Total
Inspecciones	no	medio	alto	medio	alto	4	Muy útil
Pensar en Voz Alta	sí	medio	alto	bajo	alto	5	Muy útil
Retroalimentación del Usuario	sí	bajo	alto	alto	alto	3	Muy útil
Cuestionarios, Entrevistas y Enc.	sí	medio	alto	medio	medio	3	Útil
Evaluación Heurística	no	alto	alto	bajo	alto	6	Útil
Información Post-Test	sí	medio	alto	medio	alto	1	Útil
Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	Útil
Medición del Rendimiento	sí	medio	medio	medio	medio	3	Útil
Recorrido Cognitivo	no	alto	medio	medio	medio	4	Útil
Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	Útil
Registro del Uso	no	alto	medio	alto	medio	5	Útil
Test de Usabilidad en Lab.	sí	medio	medio	medio	medio	4	Útil
Evaluación Cooperativa	sí	bajo	bajo	bajo	medio	1	Poco útil
<i>Focus Groups</i>	sí	alto	bajo	medio	medio	2	Poco útil
Grabación Audio / Vídeo	sí	medio	alto	bajo	bajo	3	Poco útil
Métricas Predictivas	no	muy alto	bajo	alto	bajo	2	Poco útil
Observación	no	alto	medio	bajo	bajo	3	Poco útil
Test de Campo	sí	medio	bajo	medio	medio	2	Poco útil
Test Experimentales	sí	muy alto	bajo	bajo	bajo	2	Poco útil

- La **Evaluación Heurística**, al ser el tipo de evaluación por expertos menos estructurada, es la que requiere un mayor grado de experiencia por parte de los evaluadores, por lo que la necesidad de formación es alta. Esta es la razón por la cual la cercanía a la IS tiene un valor bajo, porque no es común el uso de técnicas poco estructuradas. Por el contrario, esta característica le da un valor de aplicabilidad alto por poder ser de utilidad en todo tipo de desarrollos. En cuanto al grado de aportación de usabilidad frente al esfuerzo el valor es alto, puesto que el esfuerzo no es excesivo y las mejoras en usabilidad que puede aportar sí que son importantes.
- Para realizar correctamente **Inspecciones** se requieren habilidades de identificación de inconsistencias y otros errores de usabilidad, por lo que es necesario un entrenamiento previo (formación nivel medio). Las inspecciones son aplicables a todo tipo de proyectos (aplicabilidad alta), aunque las **Inspecciones Colaborativas**¹⁶ requieren una predisposición de todas las partes involucradas, por lo que para este tipo concreto de inspecciones el nivel de aplicabilidad es medio. Las inspecciones de código son una técnica habitual de la IS. Así pues, en cuanto a la mecánica de la técnica son cercanas a la IS, pero no lo son en cuanto al tipo de elementos a observar, por consiguiente, en conjunto, el nivel de cercanía es medio. La aportación de usabilidad de las Inspecciones es importante. En el caso de las Inspecciones Colaborativas el esfuerzo es alto por lo costoso y lento de las reuniones de inspección, por lo que su grado de aportación frente al esfuerzo es medio. En el resto de Inspecciones el esfuerzo es menor, por lo cual el valor para el criterio aportación/esfuerzo es alto.
- El **Recorrido Cognitivo** es una técnica que requiere tener un conocimiento amplio de los aspectos cognitivos por lo que la necesidad de formación es alta. Se centra en el rendimiento de un usuario experto en condiciones óptimas de uso, por lo que su aplicabilidad se limita a las situaciones donde dicho tipo de usuarios es relevante (aplicabilidad media). Al estar centrada en aspectos cognitivos, no resulta especialmente cercana a la IS, aunque los recorridos sí son una técnica común en la IS, por lo que el valor para el criterio de cercanía a la IS es medio. La aportación de usabilidad es importante, pero el coste de aplicación es alto, por lo que el grado de aportación frente al esfuerzo es medio.
- El **Recorrido Pluralístico**, al igual que el anterior, es costoso (por tratarse de una técnica de grupo) y también el rendimiento que se obtiene es considerable, por lo que su grado de aportación frente a esfuerzo es medio. Es igualmente cercano a la IS por tratarse de un recorrido, pero con un enfoque distinto, participativo, que no es habitual en la IS (cercanía media a la IS). Su aplicabilidad es más amplia, pero requiere que las partes involucradas estén abiertas a este tipo de técnicas, por lo que el nivel para este criterio es medio. Finalmente, las necesidades de formación son bajas, debido a que está pensada para ser aplicada por usuarios, y la organización de las sesiones de recorrido no requiere unos conocimientos extensos.
- La técnica de **Pensar en Voz Alta** no es cercana a la IS (nivel bajo), pero las necesidades de formación no son excesivas (nivel medio). Tiene una aplicabilidad alta, puesto que es útil para un abanico amplio de proyectos, y su aportación de

¹⁶ En general, no se han detallado las variantes de las técnicas (cuando éstas existen). Sólo se detallan, como en el caso de las Inspecciones Colaborativas, en aquellos casos en los que el valor de algún criterio para alguna variante difiere del valor de la técnica general.

usabilidad es bastante alta, mientras que el esfuerzo de aplicación es bajo (grado aportación/esfuerzo alto).

- En cuanto a la **Medición del Rendimiento, Test de Usabilidad en Laboratorio y Test de Campo**, son técnicas con una necesidad de formación media, puesto que tienen su mecánica que hay que aprender y requieren cierto entrenamiento, pero no de forma excesiva. Su aplicabilidad es media en el caso de la Medición del Rendimiento y de los Test de Usabilidad en Laboratorio, debido a que se centran en la medición del rendimiento, el cual puede no ser especialmente relevante en ciertos proyectos; mientras que para el Test de Campo se requiere acceso a los usuarios finales en el entorno previsto de operación, y la instalación del equipo necesario, lo cual no siempre puede llevarse a cabo (nivel bajo). Su cercanía a la IS es media, porque sin tratarse de técnicas completamente ajenas, sí que suponen un complemento a lo que habitualmente se realiza en el marco de la IS. El grado de esfuerzo es de cierta entidad en la Medición del Rendimiento y en el Test de Campo, y en el caso de los Test en Laboratorio es bastante alto por el coste que supone montar una instalación de estas características; mientras que el aporte de usabilidad en las tres técnicas es igualmente importante. Por tanto, tienen un grado medio de aportación frente a esfuerzo.
- La técnica de **Información Post-Test** requiere cierta formación para comprender el propósito y los formatos o formularios a utilizar (nivel medio). Es aplicable a todo tipo de proyectos, puesto que vale para cualquier tipo de test de usabilidad, tanto de pensar en voz alta como de medición de rendimiento (aplicabilidad general alta). No resulta ajena a la IS, pero no es cercana porque los test de usabilidad con usuarios no son algo común en la IS (cercanía a la IS media). En cuanto al esfuerzo de aplicación es bajo, y el aporte de usabilidad es bastante alto, por lo que el grado de aportación frente a esfuerzo es alto.
- La **Grabación Audio/Vídeo** tiene como principal desventaja el tiempo que consume el análisis de las grabaciones. Así, aunque puede aportar detalles que permitan mejorar la usabilidad, el grado de aportación frente a esfuerzo es bajo por dicho coste en tiempo. No es similar a ninguna técnica de IS, por lo que el valor para la cercanía a la IS es bajo. Su aplicabilidad es alta, pues en principio puede aplicarse siempre que se realice un test de usabilidad. Finalmente, requiere cierta formación (nivel medio), principalmente para lo referente al análisis de las grabaciones.
- El **Registro del Uso** entronca con el uso de monitores software en la IS para medir la eficiencia, por lo que el nivel de cercanía a la IS es alto. De todas formas, se trata de una técnica relativamente compleja, por lo que requiere un nivel de formación alto. No es aplicable a todos los casos, en tanto únicamente es practicable cuando la organización cliente está dispuesta a que se recojan datos sobre el uso del sistema por parte de los usuarios finales (aplicabilidad media). El esfuerzo de establecimiento de los monitores software es alto, aunque aporta información valiosa de posibles puntos problemáticos de usabilidad, debido a que permite conocer en detalle el uso real de las distintas funcionalidades. Por esta razón el grado de aportación/esfuerzo es medio.
- La **Observación** requiere una formación alta porque es preciso estar muy centrado en cuanto a qué observar, y no es aplicable en todo tipo de proyectos (valor medio en aplicabilidad general) porque puede realizarse únicamente cuando se tiene la posibilidad de observar en la organización de los usuarios. No se trata de una técnica cercana a la IS, porque no se suele evaluar el producto instalado de esta forma (nivel de cercanía a la IS bajo). Finalmente, el grado de aportación/esfuerzo es bajo porque se necesita emplear mucho tiempo para observar un número suficiente de problemas de usabilidad.

- Los **Cuestionarios, Entrevistas y Encuestas** tienen una aplicabilidad general alta, puesto que son útiles para todo tipo de proyectos. La necesidad de formación es importante sin resultar excesiva (nivel medio). Sí se manejan en cierta medida cuestionarios en la IS (tests beta), por lo que hay cierta cercanía a la IS (nivel medio). El esfuerzo de elaboración, distribución y análisis de los cuestionarios es considerable, sin embargo sí que pueden reflejar un número importante de problemas de usabilidad, por lo que el grado de aportación frente al esfuerzo es medio.
- Los **Focus Groups** son utilizados por las compañías de software comercial por lo que no es ajeno a la IS, aunque tampoco es una técnica común (nivel de cercanía a la IS medio). Manejar una reunión tipo *Focus Groups* no resulta sencillo, puesto que se trata de llevar una agenda pero dando la impresión a los participantes de que la discusión fluye libremente. Así, se requiere un entrenamiento y experiencia altos para aplicar esta técnica (necesidad de formación alta). La técnica se puede aplicar únicamente cuando se cuenta con una base de usuarios interesados a los que poder reunir, lo cual no ocurre habitualmente (aplicabilidad general baja). Finalmente, aunque la aportación de usabilidad no es desdeñable, el esfuerzo de aplicación es considerable, por lo que el grado de aportación frente a esfuerzo es medio.
- La **Retroalimentación del Usuario** puede ser de muchos tipos, pero las quejas del cliente/usuario en general son algo común para los ingenieros software, por lo que podemos considerar a esta técnica con un nivel alto de cercanía a la IS. La complejidad de la técnica puede variar, pero en sus formas más comunes la necesidad de formación es baja. En todo tipo de proyectos se puede contar con la retroalimentación del usuario, por lo que su aplicabilidad general es alta. Finalmente, sí que resulta de gran utilidad para el descubrimiento de problemas de usabilidad, y su coste es bajo, por lo que el nivel aportación/esfuerzo es alto.
- Tanto los **Test Experimentales** como las **Métricas Predictivas** son técnicas que requieren una formación extensa para su aplicación (necesidad de formación muy alta). Su aplicabilidad general es baja, puesto que únicamente en sistemas donde la usabilidad es crítica valdrá la pena realizar test como los realizados en Psicología, o será rentable el esfuerzo del uso de Métricas Predictivas. Ambas técnicas son muy costosas de aplicar, y las aportaciones de usabilidad no son tan valiosas, por lo que el grado de aportación frente al esfuerzo es en ambos casos bajo. En cuanto a la cercanía a la IS, los Test Experimentales no lo son (nivel bajo), puesto que en la disciplina únicamente se utilizan en el entorno investigador. Las Métricas Predictivas, sin embargo, sí son cercanas a los métodos formales de la IS, por lo que el nivel de cercanía a la IS es alto.
- La **Evaluación Cooperativa** se describe como una técnica que puede ser usada por diseñadores y usuarios sin conocimientos IPO, por lo que su necesidad de formación es baja. La aplicabilidad general es baja, puesto que requiere una base de usuarios que puedan tomar el rol decisor que esta técnica requiere, lo cual no suele ser habitual. Así mismo, no es habitual en la IS que los usuarios decidan qué técnica emplear y qué evaluar, por lo que el nivel de cercanía a la IS es bajo. La aportación de usabilidad es variable, depende del grupo de participantes, por lo que asignamos un valor medio al grado aportación/esfuerzo.

4.5 Selección de Técnicas

Tras caracterizar cada técnica de usabilidad identificada en la literatura, vamos a basarnos en la valoración total de utilidad para realizar una selección de técnicas. Las técnicas

seleccionadas van a formar parte del marco de integración que constituye la solución propuesta en el presente trabajo de investigación.

El marco de integración no se plantea como una solución que tenga que adoptarse como un todo, por el contrario, se trata de una propuesta que incluirá un abanico de técnicas que en algunos casos pueden resultar de utilidad. Por otra parte, un objetivo ya mencionado anteriormente es mantener la complejidad del marco en un nivel razonable, con el fin de potenciar su comprensibilidad y aplicabilidad desde la perspectiva de la IS. Así, hemos decidido seleccionar tanto aquellas técnicas cuya valoración total es de "muy útil" como aquellas para la que dicha valoración total es "útil", puesto que con el conjunto de técnicas resultante (treinta y cinco) se cumplen los objetivos.

Para que la aplicación de las 35 técnicas seleccionadas pueda realizarse con garantías de éxito, los desarrolladores deben formarse en el uso de las técnicas. Para este fin, los desarrolladores necesitarán acceder a textos de referencia para cada técnica en los que se explique con detalle su aplicación. Así, aun cuando tenemos referencias de textos seleccionados de la literatura IPO donde se recomienda cada técnica (ver tablas de la sección 4.3.3), vamos a proporcionar para cada técnica de usabilidad una referencia básica. Dicha referencia servirá de base para los desarrolladores que quieran aplicar una técnica como parte de sus actividades de desarrollo, ofreciéndoles un punto de partida para conocer los puntos básicos de la técnica y su modo de aplicación. Mientras que para el estudio de las secciones 4.2 y 4.3 se han considerado fuentes de la literatura IPO generalistas, que dan una visión general del proceso centrado en el usuario y las técnicas a aplicar en cada actividad, para la búsqueda de referencias básicas no hemos seguido el mismo criterio. Para proporcionar a los desarrolladores una referencia lo más completa posible para cada técnica, es necesario acudir a textos más específicos, en ocasiones centrados en un conjunto reducido de técnicas o actividades. Por esta razón, no hemos considerado ninguna restricción a priori sobre el tipo de texto a facilitar como referencia, aparte de que sea la fuente disponible con un tratamiento más a fondo de la técnica y que esté suficientemente referenciada en el campo.

La Tabla 4.14 presenta un resumen de la selección de técnicas realizada en este capítulo, incluyendo la referencia básica para cada una de ellas, a la que un desarrollador puede acudir para conseguir información detallada sobre sus características y aplicación. Las técnicas que aparecen en primer lugar, sobre fondo blanco, son aquellas que hemos considerado muy útiles, mientras que las que aparecen a continuación, sobre fondo gris, son aquellas consideradas útiles.

Tabla 4.14 - Técnicas de Usabilidad Recomendadas para su Incorporación al Proceso de Desarrollo

Actividad IPO		Técnica	Características Más Destacadas de la Técnica	Referencia Básica
Especificación del Contexto de Uso	Análisis de Usuarios	Personas	Útil cuando hay varios tipos de usuarios, esta técnica requiere un cierto esfuerzo de aprendizaje y aplicación. El retorno en usabilidad que se obtiene de su aplicación es considerable, sin embargo.	[Cooper, 03a]
	Análisis de Tareas	Casos de Uso Esenciales	Cercana a la IS, puesto que se trata de unos casos de uso a un nivel mayor de abstracción. Aplicable a un rango amplio de proyectos, requiere cierto aprendizaje, pero puede aportar una mejora sustancial en usabilidad.	[Constantine, 99]
Especificaciones de Usabilidad		Especificaciones de Usabilidad	Requiere un gran aprendizaje, pues es difícil de dominar. Aplicable cuando se conocen con precisión por anticipado las tareas a las que se va a dar soporte, mayoritariamente en entornos de oficina. Cercana a la IS por ser parte de los requisitos no-funcionales. Gran aportación a la usabilidad del sistema.	[Hix, 93]
Análisis (actividad sin especificar)		Card Sorting	Técnica participativa con necesidad de formación baja, y sencilla de aplicar. Permite conocer el mapa mental del usuario acerca del dominio de aplicación. Proporciona una gran mejora de la usabilidad frente al esfuerzo invertido.	[Robertson, 01]
Desarrollo del Concepto del Producto		Escenarios y Storyboards	Apropiada para sistemas con un alto carácter innovativo, o cuando el cliente tiene dificultades para expresar claramente qué necesita. Sencilla de aplicar, aunque requiere de una formación de cierta extensión por no tratarse de un tipo de técnica habitual en la IS. Cuando su aplicación es pertinente, la mejora en usabilidad conseguida es importante.	[Carroll, 97]
Prototipado		Prototipos de Papel	Técnica sencilla, aplicable a todo tipo de proyectos, que no requiere una extensa formación. El concepto de prototipado es cercano a la IS, a pesar de centrarse en prototipos con menor fidelidad al producto final que los prototipos habituales de la IS. El aporte a la usabilidad del producto final es muy importante.	[Snyder, 03]
Diseño de la Interacción		Árboles de Menús	Técnica de especificación relativamente sencilla, aplicable en los sistemas que cuentan con algún tipo de menú. Es de un tipo de modelado cercano al de la IS. Su aplicación produce unos resultados en la usabilidad del producto importantes frente al esfuerzo invertido.	[Shneiderman, 98]
Evaluación por Expertos		Inspecciones	Técnica que requiere una formación previa de cierta importancia, que se puede aplicar a todo tipo de proyectos. Tipo de técnica común en la IS, aunque no en lo referente a la usabilidad. Puede realizar un aporte importante a la usabilidad del sistema final.	[Nielsen, 94]
Test de Usabilidad		Pensar en Voz Alta	Útil para un abanico amplio de proyectos. Requiere cierta formación por no tratarse de un tipo de técnica habitual en la IS. La aportación a la usabilidad que se logra por medio de la enumeración en voz alta de las intenciones y problemas del usuario es muy importante. El esfuerzo invertido en este tipo de test no es, por contra, excesivo.	[Nielsen, 93]
Estudios de Seg. de Sistemas Instalados		Retroalimentación del Usuario	La consideración de las quejas/sugerencias del usuario no resulta ajena a la IS. El coste de esta técnica es bajo, pues requiere únicamente el establecimiento de un modo de reporte de incidencias que permita identificar los problemas de usabilidad. En cuanto a la aportación a la usabilidad del producto final, resulta destacable frente al coste de aplicación.	[Shneiderman, 98]

Actividad IPO		Técnica	Características Más Destacadas de la Técnica	Referencia Básica
Especificación del Contexto de Uso	Análisis de Usuarios	Mapa de Roles de Usuario	Técnica relativamente sencilla, con un tipo de modelado como el habitual en la IS. Aplicable cuando el número de tipos de usuario es alto y se pueden establecer relaciones entre ellos. Aporta la profundidad en el estudio del usuario necesaria para poder hacer aportes a la usabilidad del sistema.	[Constantine, 99]
		Perfiles de Usuario	Técnica aplicable a un rango amplio de proyectos, que requiere cierto esfuerzo de aprendizaje. Se trata de una técnica básica en cualquier desarrollo preocupado por la usabilidad, por lo que su aporte en este sentido es muy importante.	[Mayhew, 99]
	Análisis de Tareas	Escenarios de Tareas	Útil cuando no están bien definidas las tareas que los usuarios medios van a realizar con el sistema. Necesaria una formación específica para su aplicación. Ayudan a centrarse en la usabilidad del sistema, aunque su aporte no es tan alto como el de otras técnicas.	[Mayhew, 99]
		HTA	Se basa en la copia de cómo se realizan las tareas habitualmente, por lo que es apropiada a proyectos de informatización de tareas. Cercana a la IS por basarse en gran medida en el modelado. Tiene un proceso detallado de aplicación que hace la técnica más costosa de aplicar, con una contribución de usabilidad cierta pero que no despunta frente a otras técnicas.	[Annett, 04]
	Análisis Competitivo	Esta técnica puede aplicarse a todo tipo de proyectos. Requiere cierto grado de formación para poder ser aplicada con éxito. Su aplicación contribuye muy positivamente a la usabilidad del sistema desarrollado.	[Nielsen, 93]	
	Diagramas de Afinidad	Técnica de carácter participativo que requiere muy poca formación y es sencilla de aplicar. Su aplicación puede ofrecer una importante mejora de usabilidad en el producto final.	[Beyer, 98]	
	Investigación Contextual	Requiere una disponibilidad alta en la organización cliente, pues requiere un usuario representativo para aplicar la técnica. La formación para poderse aplicar adecuadamente es considerable, y la aportación de usabilidad que puede realizar es alta.	[Beyer, 98]	
	JEM	Técnica detallada de modelado, que requiere una formación de cierta entidad para poder aplicarse. Útil para conseguir que todas las partes afectadas se involucren en el proyecto. Basada en la técnica IS denominada JAD (<i>Joint Application Design</i>). Su coste de aplicación es alto por requerir reuniones multidisciplinarias, pero su posible aportación en términos de usabilidad es importante.	[Constantine, 99]	
	Observación Etnográfica	Adecuada cuando existe un alto grado de disponibilidad en la organización cliente. Requiere una formación extensa para poderse aplicar correctamente, pues requiere desarrollar las aptitudes de observación sin intervenir. Resulta relativamente compleja de aplicar, por lo que su aportación de usabilidad en comparación con el esfuerzo invertido es media.	[Wixon, 96]	
	Desarrollo del Concepto del Producto	Tormenta de Ideas Visual	Se trata de una técnica sencilla de aplicar, que permite una contribución participativa a la generación de ideas inicial de un proyecto. La aportación en usabilidad que puede producir es relevante.	[Preece, 94]

Actividad IPO	Técnica	Características Más Destacadas de la Técnica	Referencia Básica
Diseño de la Interacción	Diagramas de Transición de Estados de la Interfaz	Puesto que los Diagramas de Transición de Estados son un modelo habitual de la IS, la formación requerida para la aplicación de esta técnica es baja. Es aplicable a productos con IUs gráficas. La aportación en usabilidad puede resultar importante, si bien su aplicación puede requerir un esfuerzo igualmente importante.	[Wasserman, 85]
	Guía de Estilo del Producto	La complejidad de aplicación de esta técnica se justifica únicamente en sistemas de cierta envergadura. La formación necesaria para aplicarla es amplia. La aportación en usabilidad puede resultar muy importante, si bien requiere un empleo de recursos considerables para toda la gestión que envuelve el uso de esta técnica.	[Mayhew, 99]
	Mapa de Navegación	Apropiada cuando la interfaz se compone de distintos contextos o ventanas entre los cuales navega el usuario. Requiere una formación de cierta extensión, y su aplicación resulta compleja. Así, las posibles mejoras en usabilidad son importantes, aunque requiere un esfuerzo igualmente importante.	[Constantine, 99]
	Modelo del Contenido de la Interfaz	Debido a la diferencia con técnicas habituales de la IS tiene una necesidad de una formación de cierta entidad. Es aplicable para el diseño de IUs gráficas. Requiere cierto esfuerzo su aplicación, pero los resultados en mejora de la usabilidad son acordes con el esfuerzo invertido.	[Constantine, 99]
	Análisis de Impacto	Aplicable cuando se tienen datos con los que poder estimar con cierta precisión el impacto y coste de las soluciones a los problemas de usabilidad identificados. Tales datos son complejos de reunir y analizar, aunque la mejora en usabilidad derivada de su aplicación puede ser importante.	[Hix, 93]
Diseño (actividad sin especificar)	Organización de la Ayuda según Casos de Uso	Apropiada para proyectos con cierta complejidad y con un número limitado de casos de uso básicos. Tiene una cierta cercanía con la IS debido al empleo de la técnica de casos de uso, pero aún así requiere una formación de cierta entidad para su aplicación. El aporte de usabilidad no es de los más destacados, pero su aplicación no es costosa.	[Constantine, 99]
	Evaluación Heurística	La aplicación de esta técnica requiere una formación en usabilidad alta, debido a la experiencia necesaria para aplicar la técnica de forma adecuada. Puede ser aplicada a todo tipo de proyectos, y puede influir de forma importante en la usabilidad del sistema final si se plantea como complemento a técnicas de test de usabilidad con usuarios. Puede servir para hacer algunos ciclos iterativos más cortos, pues su aplicación por expertos no es excesivamente costosa en tiempo.	[Nielsen, 93]
Evaluación por Expertos	Inspecciones Colaborativas	Este tipo de inspecciones de carácter participativo requiere una predisposición de todas las partes involucradas en el desarrollo a este tipo de prácticas. No requiere una formación extensa, pero es relativamente costosa su aplicación. Aún así, las mejoras obtenidas en usabilidad son destacables.	[Constantine, 99]
	Recorrido Cognitivo	La aplicación de esta técnica requiere un grado de experiencia importante, por lo que demanda una inversión considerable en formación. Adecuada cuando se quiere optimizar el rendimiento de un usuario experto en condiciones óptimas de uso. Su aplicación es costosa, y la mejora en usabilidad que se consigue es considerable, pero únicamente en el aspecto antes mencionado.	[Lewis, 97]

Actividad IPO	Técnica	Características Más Destacadas de la Técnica	Referencia Básica
	Recorrido Pluralístico	Se trata de una variante de <i>walkthrough</i> de carácter participativo. Al igual que con otras técnicas de este tipo, requiere una predisposición a aceptar la validez de este tipo de técnicas por parte de todos los implicados en el desarrollo. Las necesidades de formación son bajas, aunque el esfuerzo de aplicación es alto por tratarse de una técnica de grupo. Aún así, el rendimiento obtenido en mejora de la usabilidad es considerable.	[Bias, 94]
	Información Post-Test	Requiere una formación de cierta entidad, y se aplica a todo tipo de proyectos que permitan realizar un test de usabilidad. El esfuerzo de aplicación es bajo, y el aporte de usabilidad es importante.	[Constantine, 99]
Test de Usabilidad	Medición del Rendimiento	Adecuada para proyectos en los que es relevante el rendimiento alcanzado por los usuarios. La mecánica de preparación y realización de este tipo de tests requiere de un aprendizaje de cierta entidad. Requiere un esfuerzo importante su aplicación, y la aportación en usabilidad es igualmente importante	[Dumas, 99]
	Test de Usabilidad en Laboratorio	Adecuada para proyectos en los que el rendimiento alcanzado por los usuarios tiene una especial relevancia. Es necesario realizar una formación de cierta extensión para poder aplicarla, y su coste es elevado por el equipamiento especial que requiere. El aporte de usabilidad que se consigue mediante su aplicación es importante, sin ser excesivamente alto.	[Dumas, 99]
Estudios de Seguimiento de Sistemas Instalados	Cuestionarios, Entrevistas y Encuestas	Resultan de utilidad para todo tipo de proyectos, requieren una formación de cierta entidad, mientras que el esfuerzo de aplicación es considerable por lo trabajoso del proceso de elaboración de cuestionarios, selección de participantes y análisis de los resultados recogidos. La aportación de usabilidad es bastante importante, acorde con el esfuerzo que requiere su aplicación.	[Mayhew, 99]
	Registro del Uso	Se trata de una técnica similar a los monitores software. Es aplicable en los casos en los que existe la posibilidad de registrar las acciones de los usuarios reales y tener acceso a dicha información. Su complejidad hace que requiera una formación extensa y que su aplicación sea costosa. La información que puede aportar es valiosa para la mejora de la usabilidad del sistema.	[Shneiderman, 98]

Capítulo 5

Asignación de

Técnicas IPO

5.1 Introducción

En el capítulo anterior se han estudiado las actividades del desarrollo presentes en la literatura IPO, como marco en el que encajar las técnicas de IPO individuales, entre las cuales se ha realizado una selección. Para su integración en el proceso de desarrollo general, es preciso utilizar la terminología de actividades de la IS, puesto que es la que los desarrolladores identifican y conocen. De esta forma, los desarrolladores podrán conocer cómo encajan en las actividades que ellos realizan las técnicas que se aplican para lograr software usable.

Con dicha finalidad, en este capítulo vamos a realizar una asignación entre cada actividad IPO y el tipo de actividad correspondiente en un desarrollo general. A continuación, basándonos en dicha asignación, vamos a estudiar para las técnicas seleccionadas en qué tipo de actividad de IS encaja cada técnica concreta.

5.2 Relación entre Actividades IPO y Actividades IS

El conjunto de actividades de un proceso centrado en el usuario identificadas en secciones anteriores están descritas según la terminología IPO. Sin embargo, los desarrolladores no están familiarizados con esta terminología. Así pues, es preciso identificar cómo encajan las actividades IPO con las actividades que se llevan a cabo en cualquier proceso de desarrollo tradicional.

Mientras en algunos casos las actividades de usabilidad se integran en actividades existentes, como es el caso de las Especificaciones de Usabilidad que se integran en la Especificación de Requisitos, en otras ocasiones es necesario añadir actividades adicionales que no suelen llevarse a cabo en un desarrollo no centrado en el usuario, como por ejemplo el Diseño de la Interacción. Vamos a tomar el SWEBOK (*Software Engineering Body of Knowledge*) [SWEBOK, 04] como base, siempre que sea posible, para la definición de la terminología de actividades en un proceso de desarrollo tradicional. Para las actividades que no son comunes en los procesos de desarrollo de la IS vamos a utilizar la terminología IPO, siempre que no lleve a confusión desde una perspectiva de IS.

Para describir la relación existente entre las actividades IPO y las actividades del desarrollo general, vamos a tratarlas por tipos genéricos de actividades: Análisis, Diseño y Evaluación.

5.2.1 Actividades IPO Relacionadas con Análisis

Las dos actividades IPO relacionadas con análisis son la Especificación del Contexto de Uso y las Especificaciones de Usabilidad. La primera de ellas se subdivide en Análisis de Usuarios y Análisis de Tareas. Tal y como se han descrito estas actividades en el capítulo 4, podemos observar que sus objetivos son cercanos a los de la ingeniería de requisitos, tal y como es llevada a cabo habitualmente en los procesos de desarrollo de la IS. En efecto, tareas clave para la usabilidad del sistema final, como son el análisis de los usuarios y sus tareas, la observación de los usuarios en su entorno habitual, la identificación de sus necesidades y la concepción del producto software que mejor puede dar soporte a dichas necesidades, son actividades que vienen realizándose en la IS dentro de esta subdisciplina (ingeniería de requisitos). La IPO puede aportar su perspectiva centrada en el usuario, para que dichas actividades se realicen en el desarrollo de software con mejores resultados en cuanto a la usabilidad del producto final, enfatizando dicha perspectiva en todo el desarrollo.

Según se describe en el SWEBOK, los tipos de actividades relativas a requisitos son la Educción, Análisis, Especificación y Validación. Analizando las actividades IPO y las técnicas que proponen para dichas actividades, resulta muy difícil establecer una delimitación clara entre aquellas que corresponden a la Educción y aquellas que corresponden al Análisis de Requisitos. En un gran número de técnicas de usabilidad analizadas (por ejemplo, JEM y HTA) la recogida de información de los usuarios (Educción) y la organización y modelado de dicha información (Análisis) están integradas en la misma técnica como distintos pasos de la misma. De forma adicional, en las técnicas de tipo participativo, la educación y modelado se realiza en el mismo momento en el tiempo, puesto que los usuarios modelan junto con los desarrolladores, reflejando en dicho proceso su conocimiento sobre el dominio. Así pues, para nuestro objetivo de asignación de técnicas IPO a actividades, no vamos a realizar la delimitación típica en IS entre las actividades de Educción y Análisis de Requisitos. Por el contrario, debido a que no existe una diferenciación entre ambas actividades en la literatura IPO consultada, vamos a considerar un único grupo de actividades "Educción y Análisis de Requisitos". Respecto a los otros dos tipos de actividades relacionados con requisitos, Especificación y Validación, sí tienen una frontera más definida que permite trabajar con ellos por separado.

La actividad IPO de Especificación del Contexto de Uso incluye tanto técnicas para educir los requisitos como técnicas que permiten reflejar lo observado en modelos que puedan servir de referencia en el diseño de la interacción del sistema. Las actividades de Análisis de Usuarios y de Análisis de Tareas encajan en esta categoría de actividades relacionadas con los requisitos. Por lo tanto, la actividad de Especificación del Contexto de Uso se corresponde con la Educción y Análisis de Requisitos en un proceso de desarrollo tradicional. Dentro de la Educción y Análisis habrá que considerar las actividades específicas de Análisis de Usuarios y de Tareas, puesto que son de especial importancia para la usabilidad del sistema software a desarrollar.

En cuanto a las Especificaciones de Usabilidad, puesto que su cometido es el establecimiento de unos objetivos de usabilidad que debe cumplir el sistema a desarrollar, encajan en la definición que el SWEBOK da del tipo de información que aparece en el Documento de Definición del Sistema (*System Definition Document*), uno de los tres tipos de documentos de los que se ocupa la Especificación de Requisitos. Según el SWEBOK, el Documento de Definición del Sistema "define los requisitos de alto nivel del sistema desde la perspectiva del dominio" [SWEBOK, 04]. Las Especificaciones de Usabilidad están redactadas siempre desde el punto de vista del uso que se va a dar al sistema, por lo que pertenecen al dominio del problema. Así pues, la actividad IPO de Especificaciones de Usabilidad encaja con la Especificación de Requisitos.

En la Figura 5.1 se muestra la relación entre las actividades IPO relacionadas con análisis y los correspondientes tipos de actividades IS.

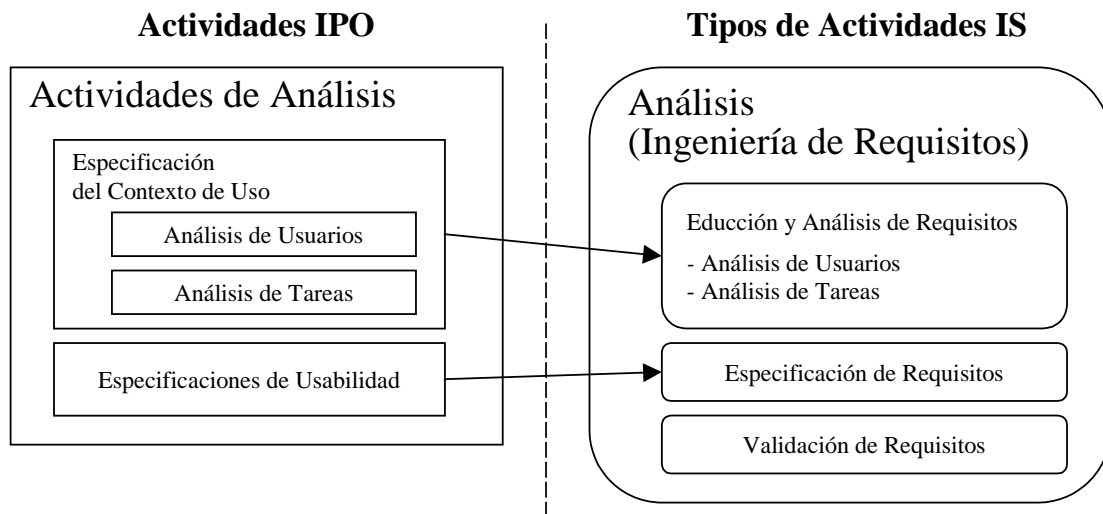


Figura 5.1 - Correspondencia entre Actividades IPO Relacionadas con Análisis y Tipos de Actividades IS

5.2.2 Actividades IPO Relacionadas con Diseño

En el estudio de las actividades que la literatura IPO describe en relación con el diseño, nos encontramos con algunas que son consideradas por la IS como no pertenecientes al diseño, puesto que la IS maneja una definición de diseño más restrictiva.

La definición general de diseño que se maneja en la IS, y que se cita expresamente en el SWEBOK, es la que da el Glosario Estándar de Terminología editado por la IEEE [IEEGlossary, 90]. Según dicha definición, diseño es: "el proceso de definir la arquitectura, componentes, interfaces y otras características de un sistema o componente y el resultado de dicho proceso" [IEEGlossary, 90]. La definición de alto nivel de en qué consiste el sistema, lo que podemos denominar conceptualización del producto, no tiene cabida en esta definición de diseño.

Por otra parte, la técnica de Prototipado (usada en la IS) no se circunscribe al área del diseño, sino que se menciona explícitamente en el área de requisitos. En concreto, en el SWEBOK se menciona la técnica de Prototipado como aplicable tanto para Educción como para Validación de Requisitos. De hecho, el prototipado recibe mayor atención en el capítulo del SWEBOK dedicado a los requisitos, que en el dedicado al diseño (donde únicamente se menciona como posible medio de evaluar un diseño de forma dinámica).

En cuanto a la actividad de Prototipado según se maneja en el campo de la IPO, se trata principalmente del uso de los prototipos de baja fidelidad, los cuales se mencionan explícitamente en el SWEBOK como posibles herramientas para la Educción de Requisitos. Por estas razones, la actividad IPO de Prototipado se ha relacionado con la Educción y Análisis de Requisitos y con la Validación de Requisitos de cualquier proceso.

El Desarrollo del Concepto del Producto pertenece a lo que en el SWEBOK se denomina "diseño de invención". En el capítulo dedicado al diseño software se detalla que el "área de Diseño Software no aborda el diseño-I (diseño de invención, que habitualmente se lleva a cabo como parte del proceso de requisitos software con el objetivo de conceptualizar y

especificar [un sistema] software para satisfacer las necesidades y requisitos descubiertos), puesto que este tema debería considerarse como parte de la actividad de análisis y especificación de requisitos" [SWEBOK, 04]. El Desarrollo del Concepto del Producto se trata de una actividad de conceptualización que consideramos básica para el éxito de las tareas relacionadas con requisitos, y además encaja bien en la descripción del diseño de invención. Por estas razones, consideramos al Desarrollo del Concepto del Producto como relacionado con las actividades de Educación y Análisis de Requisitos.

El Diseño de la Interacción, por el contrario, no tiene fácil acomodo entre los temas tratados en el SWEBOK. A diferencia del Prototipado y el Desarrollo del Concepto del Producto, las tareas relacionadas con el Diseño de la Interacción no tiene una relación directa, uno a uno, con actividades concretas del desarrollo. El SWEBOK considera los temas de interacción y presentación (cómo estructurar y organizar la interacción con usuarios y la presentación de información) como uno de los aspectos del diseño software. Dichos aspectos son propiedades que afectan al rendimiento y semántica de los componentes del diseño de forma sistemática. Según se ha descrito en el capítulo 4, el Diseño de la Interacción IPO se encarga de la definición de los entornos de interacción y su comportamiento, en concreto se ocupa de la coordinación de la interacción entre el usuario y el sistema, por tanto es parte del aspecto de interacción y presentación mencionado en el SWEBOK. También se ocupa el Diseño de la Interacción de la especificación de los detalles de la IU. El SWEBOK se refiere al diseño de la IU no como parte de la IS, sino como una disciplina relacionada. Por un lado, tenemos que el Diseño de la Interacción abarca más allá de lo concerniente a la especificación de los detalles visuales de la IU. Por otro, en las técnicas IPO no están desligados los temas de interacción y presentación de los de especificación de los detalles de la IU. Así pues, creemos que es preciso considerar un nuevo tipo de actividad en el desarrollo donde dichas técnicas tengan acomodo. Puesto que las tareas desempeñadas en el Diseño de la Interacción encajan en la definición de diseño del Glosario de Terminología de la IS detallada más arriba, hemos considerado que tiene cabida entre las actividades de diseño de cualquier desarrollo.

En la Figura 5.2 se muestra la relación entre las actividades IPO relacionadas con el diseño y los correspondientes tipos de actividades IS.

5.2.3 Actividades IPO Relacionadas con Evaluación

La Evaluación de Usabilidad es una actividad que se realiza de forma bastante independiente del resto de actividades de evaluación del desarrollo. Tiene una complejidad alta por la diversidad de tareas que pueden realizarse para llevarla a cabo, así como por el alto número de técnicas disponibles con este fin. Debido a dicho grado de independencia del resto de actividades de evaluación, hemos optado por crear un nuevo tipo de actividad denominado Evaluación de Usabilidad, que agrupa todas las actividades relacionadas con este tipo de evaluación. Dentro de este conjunto, distinguimos entre los tres grandes grupos de actividades de Evaluación de Usabilidad: Evaluación por Expertos, Test de Usabilidad y Estudios de Seguimiento de Sistemas Instalados.

Dentro de las actividades de Evaluación de la Usabilidad, hay que destacar el grupo de actividades de la Evaluación por Expertos por las siguientes razones. Mientras los Test de Usabilidad y los Estudios de Seguimiento de Sistemas Instalados se basan principalmente en un sistema funcionando, la Evaluación por Expertos no necesariamente. Así, estas actividades pueden servir para validar productos con los que se trabaja en las actividades de requisitos. Por esta razón, la Evaluación por Expertos se relaciona con la actividad de Validación de Requisitos además de con las actividades de evaluación propiamente dichas.

En la Figura 5.3 se muestra la relación entre las actividades IPO relacionadas con la evaluación y los correspondientes tipos de actividades IS.

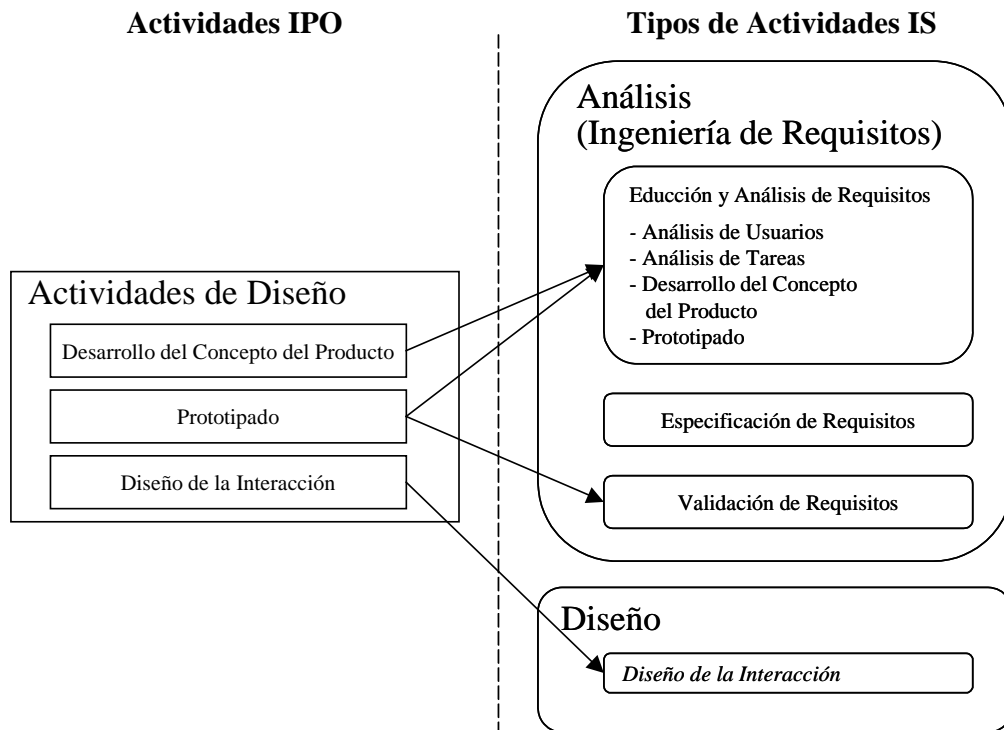


Figura 5.2 - Correspondencia entre Actividades IPO Relacionadas con el Diseño y Tipos de Actividades IS

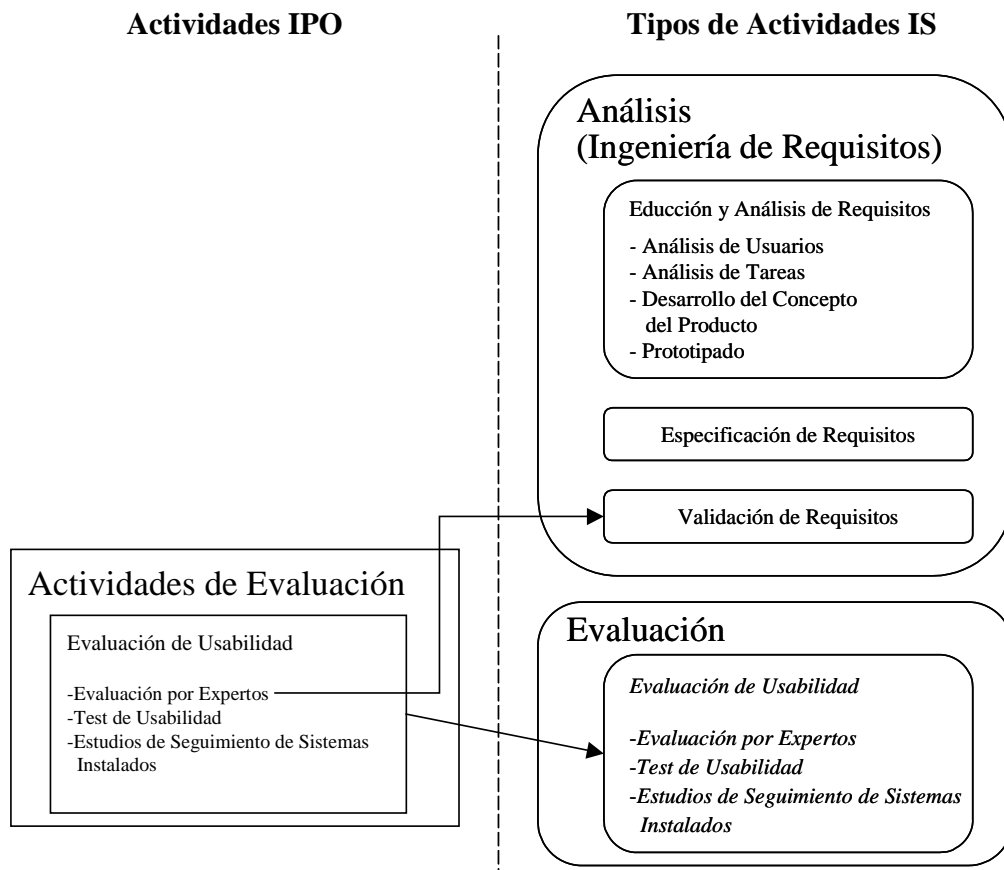


Figura 5.3 - Correspondencia entre Actividades IPO Relacionadas con la Evaluación y Tipos de Actividades IS

5.2.4 Descripción de la Relación entre Actividades IPO y Actividades IS

Las relaciones entre actividades IPO y las actividades IS, tal y como se han descrito en las secciones anteriores, se muestran en la Figura 5.4. En la parte izquierda se encuentran las actividades de un proceso centrado en el usuario, y en la parte derecha los tipos de actividades de un proceso de desarrollo de la IS relacionados con la usabilidad. Cada actividad IPO está unida mediante una flecha con el tipo de actividad correspondiente del desarrollo. Aquellos tipos de actividades que no son comunes en los procesos de la IS, y que hemos añadido para englobar técnicas IPO que no encajan en actividades IS, aparecen resaltados en *itálica* en la figura.



Figura 5.4 - Relación entre las Actividades IPO y los Tipos de Actividades IS Afectados por la Usabilidad

Nótese que en la parte derecha de la Figura 5.4 únicamente se han incluido los tipos de actividades del desarrollo relacionados con la usabilidad. Para conseguir un mapa completo del desarrollo habría que incluir en dicha parte el resto de actividades que forman parte de la

construcción de software como, por ejemplo, el Diseño Arquitectónico, el Diseño Detallado o las Pruebas de Software. No se han incluido porque no existe en la IS un consenso ampliamente aceptado acerca de los nombres de las actividades que forman parte de la construcción de software, y la obtención de tal catálogo de actividades requeriría la realización de un estudio que no tiene relación directa con el presente trabajo de investigación. A este respecto, nótese que el estudio realizado acerca de los tipos de actividades del desarrollo relacionados con la usabilidad ha resultado más dificultoso de lo inicialmente previsto, debido a esta falta de visión común en la IS. Por ejemplo, en el caso de las actividades relacionadas con los requisitos, entre la Versión de Prueba (*Trial Version*) del SWEBOK [SWEBOK, 01] y la del 2004 [SWEBOK, 04] cambian los tipos de actividades consideradas para dicha área de conocimiento de la IS, puesto que la actividad de Gestión de Requisitos que aparecía en la Versión de Prueba no se considera como actividad en la versión del 2004.

Las relaciones que se han descrito sirven a modo de guía general para enlazar la visión de proceso de la IPO con la que se maneja habitualmente en la IS. Sin embargo, para la aplicación de técnicas específicas es preciso considerar una a una la actividad del desarrollo en la que su aplicación puede resultar más útil para la mejora de la usabilidad del producto final. Así, esta asignación entre actividades nos servirá de vehículo para la asignación de técnicas individuales a las actividades de IS, pero no de un modo rígido y cerrado, sino como una primera aproximación que vamos a refinar en las secciones posteriores de este capítulo.

5.3 Asignación de Técnicas IPO a Tipos de Actividades IS

Basándonos en lo descrito para las actividades en la sección anterior vamos a presentar en ésta la asignación de técnicas IPO a tipos de actividades IS. Debido a la diversidad del campo de la IPO, mencionada en secciones anteriores, la tarea presenta cierta dificultad. Las técnicas IPO no siempre están descritas en la literatura enmarcadas en una actividad, e incluso cuando están asignadas a una actividad, es preciso comprobar, para cada técnica individual, si tiene alguna matización que afecte a la asignación que realizamos a una actividad concreta. Por esta razón, en esta sección vamos a considerar una a una la asignación de técnicas a tipos de actividades, agrupándolas según los tipos de actividades IS que aparecen en la parte derecha de la Figura 5.4, y destacando, para cada técnica, lo que puede aportar a dicho grupo de actividades.

5.3.1 Técnicas Asignadas a Actividades de Análisis

Los tres tipos de actividades IS de análisis (ingeniería de requisitos) que hemos identificado como afectadas por la usabilidad son la Educación y Análisis de Requisitos, la Especificación de Requisitos, y la Validación de Requisitos.

5.3.1.1 Educación y Análisis de Requisitos

En la IS se reconoce que la Educación y Análisis de Requisitos es una de las tareas clave para el éxito del proyecto de desarrollo completo. En los esfuerzos de educación, las técnicas IPO pueden ofrecer fuentes de información y medios de extraer conocimiento suplementarios a los que se emplean tradicionalmente en la ingeniería de requisitos. Así mismo, en el análisis de la información obtenida, con los modelos de tareas y de usuarios que se emplean habitualmente en el campo de la IPO, se pueden centrar más eficazmente las tareas de requisitos en los verdaderos usuarios del sistema a desarrollar, y sus necesidades reales. En lo referente a la

negociación de requisitos (una parte importante del Análisis de Requisitos), las técnicas con carácter participativo están especialmente indicadas para enriquecer las labores de negociación con dicha componente participativa.

Debido a la gran importancia que estos temas tienen en el campo de la IPO, existe un gran número de técnicas que encajan con los fines de las actividades de Educación y Análisis de Requisitos. Quince de las técnicas IPO seleccionadas corresponden a este tipo de actividades. Para ayudar a los desarrolladores a ubicar mejor este amplio número de técnicas, y para permitir una mejor comunicación con expertos IPO para estas actividades clave, vamos a agrupar las técnicas correspondientes a este tipo de actividades según la actividad IPO a la que corresponden según la clasificación de técnicas realizada en el capítulo 4: Análisis de Usuarios, Análisis de Tareas, Desarrollo del Concepto del Producto y Prototipado. Aquellas técnicas que no están descritas en la literatura IPO como pertenecientes a una actividad concreta se incluyen en un grupo sin etiquetar.

5.3.1.1.1 Análisis de Usuarios

El Análisis de Usuarios supone una actividad crucial en los procesos centrados en el usuario, y la presencia de este tipo de actividad en cualquier desarrollo es el pilar sobre el que se pueden construir el resto de actividades de usabilidad que se apliquen durante el desarrollo. Las técnicas seleccionadas que tienen relación con este tipo de análisis son las siguientes:

- **Perfiles de Usuarios:** La recopilación de información sobre los usuarios previstos del sistema se convierte en un proceso sistemático mediante el uso de perfiles de usuario. Los distintos tipos de características que pueden resultar relevantes en un estudio de usuarios se detallan en la definición de cada perfil. De esta forma, se centran los esfuerzos de educación en aquello que más útil puede resultar para el diseño de un sistema con un nivel de usabilidad adecuado. Al estructurar la información sobre usuarios, el uso de esta técnica puede ayudar a los desarrolladores no familiarizados con el Análisis de Usuarios tal y como se realiza en la IPO, pues les marca detalladamente la información relevante sobre usuarios que deben recoger en sus tareas de Educación y Análisis de Requisitos.
- **Mapa de Roles de Usuario:** Cuando se tiene cierto número de tipos de usuarios potenciales del sistema, la estructuración de las relaciones entre los mismos mediante un Mapa de Roles de Usuario resulta útil para tener la vista general de usuarios del sistema. Este tipo de modelos que ofrecen una visión general son muy útiles en las tareas de negociación de requisitos, con el fin de contrastar con todas las partes involucradas si se están abordando los objetivos adecuados en cuanto a usuarios previstos del sistema.
- **Personas:** Esta técnica ayuda a sintetizar todos los datos de que se disponga sobre los usuarios previstos del sistema, en unos usuarios arquetípicos que puedan usarse para alcanzar consenso en el equipo de desarrollo y para centrar las discusiones de diseño. Ayuda también a determinar qué es lo que el producto debe hacer, relacionado con las necesidades a satisfacer, por lo que puede contribuir a todo el proceso de Análisis de Requisitos. Al proveer un lenguaje común para referirse a los usuarios concretos del sistema (concreto, frente al término genérico y a menudo equívoco "usuarios del sistema"), ayuda a alcanzar consenso en el equipo de desarrollo. En concreto, evita un problema típico entre un cierto número de desarrolladores: El identificar las capacidades y querencias de los futuros usuarios del sistema con las propias (del desarrollador), conduciendo este tipo de actitud a la producción de sistemas software que únicamente usuarios con marcado perfil tecnológico pueden usar.

5.3.1.1.2 Análisis de Tareas

Un ingeniero software está familiarizado con el concepto de funcionalidades que el sistema ofrece al usuario, y una de las actividades que está acostumbrado a realizar es la de descomposición funcional. El Análisis de Tareas según se realiza en los procesos centrados en el usuario persigue un fin similar, pero con un enfoque distinto, centrado en los objetivos últimos del usuario en su utilización del sistema. Las técnicas que incluimos en este apartado pueden cumplir el cometido de centrar la realización de un análisis funcional del sistema en las tareas últimas que el usuario desea realizar. El usuario utiliza un sistema software como medio para satisfacer su deseo de llevar a cabo dichas tareas. Así, el análisis de las funciones que debe ofrecer el sistema se realizará con un enfoque centrado en el usuario que asegure que el sistema desarrollado responda a las necesidades y forma de trabajar del usuario. Las técnicas para este fin son las siguientes:

- **Casos de Uso Esenciales:** Esta técnica complementa el uso de la técnica de Casos de Uso de la IS, cuya utilización resulta muy habitual como parte de las actividades de análisis de los desarrollos orientados a objetos. Desde un punto de vista centrado en el usuario, la técnica de Casos de Uso se suele aplicar de forma incorrecta. Esto es debido a que la elaboración de los Casos de Uso incluye una serie de decisiones que afectan a la usabilidad, las cuales se toman sin seguir ningún criterio referente a los objetivos del usuario. Los Casos de Uso Esenciales ofrecen un modo de centrar las decisiones de la interacción entre el usuario y el sistema en los objetivos del usuario en los pasos que realiza, y en cómo el sistema da soporte a la toma de las decisiones del usuario en cada paso, mediante las respuestas que va ofreciendo.
- **HTA:** Esta técnica resulta útil para estructurar la información observada acerca de cómo el usuario organiza las tareas que usualmente lleva a cabo en su trabajo. Por tanto, el uso de esta técnica puede complementar los esfuerzos de Educación y Análisis de Requisitos cuando se trata de un sistema que pretende dar soporte al usuario en la realización de sus tareas habituales de trabajo.
- **Escenarios de Tareas:** Esta técnica lleva a situaciones concretas lo especificado en los casos de uso. Ilustra las tareas más representativas de cada tipo de usuario, y así ayuda a alcanzar consenso en el equipo de desarrollo sobre cómo el sistema va a dar soporte a las tareas del usuario. De esta forma, cumple un papel importante en el Análisis de Requisitos.
- La variante de *Card Sorting* denominada **Task Sorting** aporta a la descomposición funcional que pueda realizarse en el Análisis de Requisitos la visión del usuario acerca de las tareas que lleva a cabo. Hace participar al usuario en las labores de Análisis de Requisitos que implican poner orden en los datos obtenidos de la Educación. Así, añade una componente participativa a la descomposición de tareas que facilita de forma considerable las tareas relativas a la negociación de requisitos con todas las partes involucradas. En la selección de técnicas realizada en el capítulo 4 no hemos considerado por separado las distintas variantes de la técnica *Card Sorting* por tener las mismas características básicas. La única diferencia relevante para nuestro estudio entre *Task Sorting* y la técnica base sería la inclusión en este subgrupo de técnicas correspondientes a la actividad IPO de Análisis de Tareas. Consideramos que se trata de una diferencia mínima, al estar ambas dentro del mismo tipo de actividad IS (Educación y Análisis de Requisitos), por lo que no vamos a diferenciar en lo sucesivo la variante de la técnica base (*Card Sorting*). La descripción de la ubicación en actividades de esta última se detalla más adelante en la sección 5.3.1.1.5.

5.3.1.1.3 Desarrollo del Concepto del Producto

Entre las actividades relacionadas con la Educación y Análisis de Requisitos, el establecimiento de la visión general sobre el producto a desarrollar cumple un papel importante, tanto más importante cuanto mayor es el grado de innovación del proyecto (como opuesto a proyectos de mecanización de tareas manuales). Así mismo, para la negociación de requisitos es fundamental que todas las partes involucradas compartan una misma visión acerca del producto a desarrollar, esto es, un mismo concepto del producto. Las siguientes técnicas contribuyen a estos objetivos:

- **Escenarios y Storyboards:** Cuando se está intentando transmitir a todas las partes implicadas qué tipo de sistema se quiere construir, los escenarios y *storyboards* ayudan a centrar la narración de cómo va a ser el sistema en usuarios concretos que realizan tareas específicas. Esta técnica se puede combinar con la de Personas para conseguir una mejor definición del tipo de sistema que se va a construir, para quién va destinado, y qué necesidades pretende cubrir. Los detalles que añaden a una visión general sirven para centrar las discusiones de negociación de requisitos, para evitar que las discusiones se mantengan a un nivel abstracto que imposibilite a clientes y usuarios participar.
- **Tormenta de Ideas Visual:** La técnica de Tormenta de Ideas es una técnica de creación de ideas en grupo que, por sus características, está especialmente indicada cuando se están intentando definir las características de un producto software innovador. La variante visual de la técnica, que incluye como motor de la discusión la elaboración de bosquejos de la IU, permite participar en la producción de ideas a usuarios, cuyo aporte puede realizarse más fácilmente que si la dinámica se centrara en disquisiciones de tipo técnico. En cuanto esta técnica pretende obtener la idea del producto a desarrollar, podemos considerarla como parte de los esfuerzos de Educación de Requisitos.

5.3.1.1.4 Prototipado

Los prototipos son nombrados en el SWEBOK [SWEBOK, 04] como una herramienta útil para clarificar requisitos, en el apartado dedicado a Educación de Requisitos. Un gran riesgo en el desarrollo de un sistema es que se esté persiguiendo un objetivo erróneo, que no se esté desarrollando el sistema que se necesita. Normalmente este tipo de errores surgen de malentendidos entre el equipo que redacta los requisitos y el resto de partes involucradas. Para minimizar el riesgo de que este problema ocurra, cuando se trabaja con clientes y usuarios la siguiente técnica puede resultar de utilidad:

- **Prototipos de Papel:** Como parte de las reuniones de requisitos se pueden usar bosquejos a mano de la apariencia de la IU, o bien dibujos realizados por ordenador mediante un programa de dibujo. Este tipo de prototipos permiten transmitir al cliente la idea del sistema a construir que tiene el equipo, y poder así contrastar si se está trabajando en la dirección correcta. Frente a los prototipos que corren en un ordenador, cuentan con la ventaja de que el cliente no tiene la impresión de que el sistema está ya casi construido, puesto que la apariencia de borrador transmite la idea de que aún queda mucho trabajo por desarrollar hasta llegar al producto final. A pesar de tratarse únicamente de dibujos en papel, este tipo de prototipos son útiles también para mostrar la dimensión interactiva del sistema. Así, un miembro del equipo puede cambiar entre distintos dibujos para mostrar al cliente qué se ve en cada momento según las acciones que tome el usuario, permitiendo de esta forma expresar de una forma fácilmente comprensible cómo se ha pensado que funcione la interacción.

5.3.1.1.5 Técnicas que no Están Enmarcadas en una Actividad IPO Concreta

Entre las técnicas que pueden aplicarse para la Educación y Análisis de Requisitos tenemos otras que no están descritas como pertenecientes a ninguna de las actividades IPO de los puntos anteriores, aunque pueden tener estrechos vínculos con varias de ellas. El aporte de estas técnicas a la actividad de Educación y Análisis de Requisitos se detalla a continuación:

- **Análisis Competitivo:** Como se ha indicado en secciones anteriores, la IS da mucha relevancia al proceso de desarrollo software, enfatizando que la aplicación de un buen proceso genera un buen producto. Se han publicado una multitud de métodos de IS, cada uno de ellos sugiriendo que su aplicación producía mejores productos software. Llevando esta afirmación al extremo, no sería necesario observar soluciones ya existentes, pues la aplicación del método por sí mismo asegura la creación de un producto de calidad. Por otro lado, la parte interna de los sistemas software es la que ha recibido tradicionalmente mayor atención en la IS. Considerando la combinación de ambos factores, no sorprende que la comparación con otros productos software no haya sido la norma en la mayor parte de procesos de la IS. A diferencia de la parte interna de un sistema, la IU y el diseño de la interacción en general sí que se pueden observar en un producto competidor. La aplicación de esta técnica dota al equipo de desarrollo de inspiración sobre mecanismos de interacción a incluir en el desarrollo del sistema, y sobre problemas a evitar.
- **Investigación Contextual:** Esta técnica está estrechamente relacionada con las tareas de Educación de Requisitos en lo referente al estudio de los usuarios y de las tareas que realizan habitualmente. Su uso está indicado para aquellos proyectos de desarrollo en los que se tiene la posibilidad de tratar directamente con usuarios representativos, dispuestos a colaborar en la tarea común de desentrañar cómo realizan sus tareas habituales. Con esta técnica se consigue una comprensión más profunda de las necesidades del usuario que con las tradicionales entrevistas, debido a que los usuarios muchas veces no conocen conscientemente las razones que motivan sus actos de forma detallada.
- **Diagramas de Afinidad:** Esta técnica permite organizar información, bien extraída de sesiones de Educación de Requisitos, bien de sesiones de Tormenta de Ideas en el Desarrollo del Concepto del Producto. La componente colaborativa e integradora permite alcanzar una visión común por parte de todos los participantes. En concreto, permite la inclusión de usuarios finales en las sesiones de Análisis de Requisitos.
- **Observación Etnográfica:** Al igual que la Investigación Contextual tratada más arriba, esta técnica está estrechamente relacionada con las tareas de Educación de Requisitos en lo referente al estudio de los usuarios y de las tareas que realizan habitualmente. Se trata de una técnica que complementa a una entrevista tradicional, puesto que permite conocer los datos sobre el comportamiento del usuario de primera mano, no a través de su visión posiblemente distorsionada por diversos factores (timidez, deseo de causar una buena impresión, etc.). Así mismo, es especialmente útil cuando se quiere adecuar el sistema software a desarrollar a la cultura de la organización cliente.
- **JEM:** Esta técnica describe un proceso estructurado, guiado y colaborativo para implicar a los usuarios junto con desarrolladores en las actividades de modelado realizadas en el Análisis de Requisitos. Se trata de una extensión de la técnica JAD (*Joint Application Design*), conocida en la IS, que introduce el enfoque centrado en el usuario en dicha técnica. JEM está especialmente indicada cuando se quiere dar un

tinte participativo a las actividades de Educación y Análisis de Requisitos, pero de una forma organizada que permita un alto grado de control sobre todo el proceso.

- **Card Sorting:** Esta técnica permite comprender la lógica que manejan los usuarios, obteniéndose una representación estructurada de la información, por tanto puede servir como herramienta para la Educación y Análisis de Requisitos. Resulta útil cuando se dispone ya de una información sobre el dominio, pero se quiere organizar dicha información según la estructura mental de los usuarios. Se trata de una técnica participativa relativamente sencilla que produce mejores resultados para este fin concreto que las entrevistas. También puede contribuir en el diseño de prototipos, pues la estructura de los menús puede establecerse basándose en la información obtenida por medio de la aplicación de esta técnica.

5.3.1.2 Especificación de Requisitos

La Especificación de Requisitos tiene que ver con la elaboración de un documento donde se reflejan los requisitos que el sistema debe cumplir, y en particular tiene que ver con la estructura, calidad y verificabilidad de dicho documento. En cuanto a la usabilidad, las técnicas detalladas en el punto anterior pueden contribuir al proceso de extraer y elaborar la información sobre requisitos, pero en cuanto a la influencia en la estructura del documento, se circunscribe a la inclusión de un apartado con Especificaciones de Usabilidad. La descripción de cómo esta técnica contribuye a la Especificación de Requisitos se detalla a continuación:

- **Especificaciones de Usabilidad:** Son objetivos de usabilidad que se establecen para el producto software a obtener. Deben ser verificables para que puedan servir como guía para conocer si el proyecto está avanzando en la dirección establecida inicialmente, en el marco de un desarrollo iterativo. La inclusión de estas especificaciones en los documentos de Especificación de Requisitos introduce la usabilidad como uno más entre los aspectos que pueden ser establecidos cuantitativamente y por adelantado, evitando de esta forma que se perciba la usabilidad como un valor cambiante que depende del observador. Puesto que las Especificaciones de Usabilidad se describen de tal forma que incluyen el modo de verificación de su cumplimiento, son apropiadas para su inclusión en los test de aceptación del producto. Finalmente, las Especificaciones de Usabilidad actúan también como inspiración para los desarrolladores a lo largo de todo el desarrollo, especialmente en aquellos momentos en que es preciso escoger entre alternativas de diseño.

5.3.1.3 Validación de Requisitos

El SWEBOK [SWEBOK, 04] nombra cuatro temas relevantes respecto a la Validación de Requisitos: La realización de revisiones de requisitos, el prototipado, la validación de modelos y los test de aceptación. Las técnicas de usabilidad que pueden ser útiles en la validación de requisitos pueden afectar a uno o a varios de estos subtemas. En lo referente a prototipado y test de aceptación, ya se ha mencionado en secciones anteriores la aportación de los Prototipos de Papel y las Especificaciones de Usabilidad, respectivamente. En lo referente a los otros dos temas (revisiones de requisitos y validación de modelos), las siguientes técnicas IPO pueden contribuir a esta actividad de Validación de Requisitos según se indica a continuación:

- **Evaluación Heurística:** Junto a las revisiones de requisitos tradicionales, puede plantearse la Evaluación Heurística de usabilidad, con el objetivo específico de valorar las posibles deficiencias de usabilidad que puedan estar implicadas en los

productos elaborados en las actividades de requisitos (especificación de casos de uso, prototipos, etc.).

- **Inspecciones:** Aunque también tienen que ver con actividades de revisión de requisitos, se centran principalmente en la validación de prototipos. Son similares a la Evaluación Heurística, pero centradas en un tema particular como puede ser la adhesión a estándares o a guías de diseño, o la consistencia del diseño de la interacción. Dada la importancia que tiene la elaboración de prototipos en las actividades de requisitos, con respecto al objetivo de conseguir productos software con alta usabilidad, estas técnicas son especialmente relevantes como parte de la Validación de Requisitos.
 - **Inspecciones Colaborativas:** Esta variante aporta la integración de usuarios en el equipo que va a realizar la inspección, de una forma estructurada. En el SWEBOOK [SWEBOOK, 04] se menciona que en las reuniones de revisión de requisitos al menos un representante del cliente debe estar en el equipo de inspección en aquellos proyectos dirigidos por el cliente (*customer-driven projects*). En un proceso centrado en el usuario deben incluirse representantes de los usuarios en dicho equipo, y esta técnica aporta una forma estructurada de abordar el proceso de inspección.
- **Recorrido Cognitivo:** Esta técnica permite validar un prototipo desde el punto de vista de las actividades cognitivas que se obliga a realizar al usuario. Por tanto, en lo referente a la Validación de Requisitos, está relacionada tanto con el Prototipado como con la validación de modelos, puesto que evalúa la calidad de un prototipo entendido como modelo de la interacción entre el usuario y el sistema.
- **Recorrido Pluralístico:** Esta técnica sirve para la Validación, en tanto organiza las sesiones de revisión de requisitos (en la parte de los prototipos) en las que participan todas las partes involucradas. No persigue únicamente consignar la validez del prototipo desarrollado, sino que puede jugar un papel importante en la negociación de requisitos, puesto que contribuye a conseguir una sintonía entre todas las partes involucradas en el desarrollo. En especial, contribuye a que el usuario sienta que los temas que más le importan son atendidos convenientemente, y que el equipo de desarrollo comprende su visión del problema.

5.3.2 Técnicas Asignadas a Actividades de Diseño

La única gran actividad añadida a las actividades de diseño de la IS que se suelen realizar en cualquier desarrollo, es la de Diseño de la Interacción. De forma adicional, como se indicó en secciones anteriores, hay otras técnicas IPO relacionadas con el diseño cuya aplicación no se ubica en el Diseño de la Interacción.

5.3.2.1 Diseño de la Interacción

Al tratarse de una actividad añadida a las que suelen realizarse en cualquier desarrollo, incluye únicamente las técnicas IPO que ofrecen herramientas de trabajo para las tareas de Diseño de la Interacción, y para representar los resultados generados por la realización de dichas tareas. Estas técnicas son las siguientes:

- **Guía de Estilo del Producto:** Los modelos y otra documentación de diseño sirven en cualquier desarrollo para asegurar una visión común en el equipo de desarrollo acerca de las decisiones de diseño, y para servir de base para futuras modificaciones y/o

extensiones. La Guía de Estilo sirve para estos mismos fines, pero se refiere a todos los aspectos relativos al Diseño de la Interacción. Resulta especialmente útil cuando se quiere contar con equipos multidisciplinares en los que hay desarrolladores con escasa experiencia previa en temas de IPO, puesto que la Guía de Estilo les marca la pauta a seguir en todo lo referente al Diseño de la Interacción.

- **Árboles de Menús:** En general, podemos observar que la estructura de los elementos de la IU no se modela de forma regular en la IS, debido a que los desarrolladores no cuentan con técnicas para realizar ese tipo de modelado. Los Árboles de Menús proporcionan una técnica para el modelado de un tipo concreto de IUs, las que están basadas en una estructura de menús. Resulta una técnica relativamente sencilla de usar, que proporciona una representación visual de la estructura de menús y que permite la toma de decisiones sobre el diseño de dicha estructura.
- **Diagramas de Transición de Estados de la Interfaz:** Al igual que en la técnica anterior, esta técnica proporciona una forma de modelar un tipo concreto de IUs, aquellas basadas en distintos modos, o bien aquellas basadas en un sistema de ventanas modal (únicamente se permite interactuar con la ventana activa en cada momento). Esta técnica puede ser de utilidad también en la confección de manuales de uso, puesto que permiten transmitir al usuario la lógica de la interacción con el sistema.
- **Modelo del Contenido de la Interfaz:** Esta técnica permite realizar las tareas de diseño de la interacción con una base gráfica que favorece la discusión de alternativas. Está indicada para IUs basadas en un sistema de ventanas con distintos espacios de interacción. Al igual que las dos técnicas anteriores, cubre el vacío existente en la IS en cuanto a modelado de la IU.
- **Mapa de Navegación:** Esta técnica permite la representación de las posibilidades de navegación entre distintos contextos de interacción, por lo que su uso está indicado en el mismo tipo de sistemas que la técnica anterior. La consistencia entre los distintos elementos que forman la IU es una de las características clave de cara a la usabilidad. Mediante esta técnica se puede observar una visión general de la navegación entre los distintos espacios de interacción, lo cual permite apreciar las desviaciones del objetivo general de consistencia que puedan aparecer.

5.3.2.2 Otras Técnicas Relacionadas con el Diseño

Entre las técnicas IPO seleccionadas, dos de ellas están relacionadas con el diseño pero no están asignadas en la literatura IPO a ninguna actividad en concreto. Son las siguientes:

- **Análisis de Impacto:** Esta técnica ofrece a cualquier equipo de desarrollo preocupado por la usabilidad, un mecanismo de toma de decisiones estructurado para la asignación de prioridades a los problemas de usabilidad observados. Las modificaciones en el diseño que haya que valorar pueden tener impacto únicamente en el Diseño de la Interacción, o bien afectar también a la funcionalidad interna del sistema.
- **Organización de la Ayuda según Casos de Uso:** Los desarrolladores, cuando se les plantea el diseño de un subsistema de ayuda, a menudo no cuentan con ninguna técnica que les permita abordar tal tarea de una forma organizada. La técnica de Organización de la Ayuda según Casos de Uso puede suplir tal carencia, ligando la estructura del subsistema de ayuda a los casos de uso, y sirviendo de pauta al desarrollo de tal subsistema. Las facilidades de ayuda, cuando no se elaboran en base

a una pauta (como, por ejemplo, la sugerida por esta técnica), corren el riesgo de convertirse en un repositorio de información al cual los usuarios acaban por no acudir por las dificultades para encontrar la información que se necesita en un determinado momento.

5.3.3 Técnicas Asignadas a Actividades de Evaluación

Como se ha indicado más arriba en la sección 5.2.3, las actividades de Evaluación de Usabilidad tienen características que las hacen bastante independientes del resto de actividades de evaluación. Las actividades de Evaluación de Usabilidad constituyen el tema que ha recibido tradicionalmente mayor atención en la IPO, y por tanto se trata del más maduro. Así, frente a otras actividades en las que establecer una taxonomía de técnicas no resultaba fácil, en este tipo de actividades sí se cuenta con una estructura ampliamente aceptada en el campo de la IPO. Puesto que el nuevo grupo de actividades de Evaluación de Usabilidad tiene un alto grado de independencia del resto de las actividades de evaluación, en este punto no vamos a justificar la asignación de técnicas a cada tipo de actividad, sino que vamos a destacar su aportación al desarrollo entendido desde una perspectiva global.

Los tres grandes grupos de actividades de Evaluación de Usabilidad son la Evaluación por Expertos, los Test de Usabilidad y los Estudios de Seguimiento de Sistemas Instalados. Vamos a detallar las técnicas pertenecientes a la Evaluación de Usabilidad en el marco de estos tres grandes grupos.

5.3.3.1 Evaluación por Expertos

Las técnicas seleccionadas que pertenecen al grupo de técnicas de Evaluación por Expertos han sido tratadas en la sección 5.3.1.3 en lo referente a su posible aportación a la Validación de Requisitos. En esta sección vamos a detallar cuánto pueden aportar en las actividades de evaluación fuera del marco de la Validación de Requisitos.

- **Evaluación Heurística:** Esta técnica es la menos estructurada de las que se incluyen en esta sección. Frente a otras técnicas de evaluación, esta técnica aporta el centrarse en la usabilidad del producto software a evaluar.
- **Inspecciones:** Son similares a las inspecciones de software que se realizan habitualmente en desarrollos software, únicamente cambia el objeto de la inspección, que suele ser la IU o un prototipo de la misma. Un tipo de Inspecciones, las de consistencia, sí son un aporte específico de la IPO, puesto que se centran en una característica especialmente importante de cara a la usabilidad del producto software.
 - **Inspecciones Colaborativas:** Esta variante de las Inspecciones aporta la componente participativa que permite integrar al usuario en el equipo de desarrollo.
- **Recorrido Cognitivo:** Al igual que las Inspecciones, los recorridos son conocidos como técnica para buscar defectos en el software. Este tipo de recorridos se centra en un tema especialmente importante desde el punto de vista de la usabilidad: La carga cognitiva que se impone al usuario en su uso de la aplicación software.
- **Recorrido Pluralístico:** Lo más destacado de esta técnica participativa es que permite que el usuario aprecie que los temas que más le importan son atendidos convenientemente, y que también perciba que el equipo de desarrollo comprende su visión del problema.

5.3.3.2 Test de Usabilidad

Los Test de Usabilidad constituyen el tipo de técnica más habitual en los procesos de desarrollo centrados en el usuario, puesto que este enfoque se basa en la premisa de que no se puede asegurar cuán usable es un determinado prototipo o producto software, sin antes probarlo con usuarios representativos llevando a cabo las tareas para las que da soporte el sistema. El Test de Usabilidad aporta al desarrollo una forma de evaluar la usabilidad de forma objetiva, puesto que intenta reproducir lo más fielmente posible el uso real del sistema. Para ello utiliza como participantes a usuarios representativos que realizan sus tareas más habituales. Ofrece criterios de decisión sobre la calidad del sistema en cuanto a usabilidad, y estos criterios son más objetivos que los que pueden obtenerse de una demostración del sistema (o de un prototipo del mismo) al cliente y/o usuarios. Una demostración en la que el sistema es manejado por un miembro del equipo de desarrollo únicamente muestra las bondades del sistema, y además sigue la lógica del equipo de desarrollo. El cliente y el usuario pueden dar su visto bueno a un sistema que les han demostrado de esta forma, para descubrir al instalarlo en su organización que hay problemas importantes de usabilidad, debidos a que no encaja con las peculiaridades del trabajo diario, o con la concepción del trabajo que tienen los usuarios.

A pesar de lo básico de la premisa central en la que se basan los Test de Usabilidad (probar el sistema con usuarios representativos realizando las tareas para las que está concebido el sistema), existen variantes de la técnica básica, cuya aportación al desarrollo se detalla a continuación:

- **Pensar en Voz Alta:** Esta variante aporta a la realización de Test de Usabilidad la expresión del razonamiento interno del usuario, el conocimiento del cual puede ser clave para el análisis de los problemas de usabilidad identificados.
- **Medición del Rendimiento:** Con vistas a que el resultado del Test de Usabilidad sea lo más objetivo posible, esta variante mide la eficiencia en uso de cada participante en su uso del sistema. De esta forma, se obtienen valores cuantitativos que reflejan la usabilidad del sistema en lo referente al atributo de eficiencia en uso. Por tanto, esta variante se aplica cuando se dispone de un prototipo suficientemente completo como para permitir un uso del mismo al participante, y cuando la eficiencia en uso es uno de los atributos de usabilidad relevantes. La aplicación de esta variante es incompatible con la variante anterior (Pensar en Voz Alta), puesto que el esfuerzo de verbalizar que realiza el participante le ralentiza en su uso del sistema, haciendo así inválidas las posibles mediciones del rendimiento.
- **Información Post-Test:** Para desligar la experiencia del uso del sistema y la atención que dicho uso requiere del usuario de la verbalización de sus acciones, esta variante posibilita que el usuario analice su sesión de Test de Usabilidad una vez ésta ha terminado. El uso de esta variante permite conjugar las ventajas de la Medición del Rendimiento con las de Pensar en Voz Alta.
- **Test de Usabilidad en Laboratorio:** Esta técnica plantea la realización de los Test de Usabilidad en unas instalaciones especialmente concebidas para ello. La razón para tener dicho tipo de instalaciones yace en la complejidad de los equipos que pueden emplearse en un Test de Usabilidad para el registro del audio y/o vídeo, y para la observación por parte de miembros de todas las partes involucradas en el desarrollo (desarrolladores, cliente, promotor del proyecto, etc.). Un Laboratorio de Usabilidad proporciona un entorno controlado y consistente en el cual realizar los Test de Usabilidad. Así, la comparación de resultados de distintos usuarios o distintos sistemas se hace más fácil. Se trata de un tipo de instalaciones costosas, por lo que

aconsejamos su construcción y uso únicamente en organizaciones de tamaño medio-alto cuyo presupuesto de usabilidad puede acomodar el coste que suponen.

- La técnica de **Registro del Uso** puede aportar en el Test de Usabilidad información adicional sobre las acciones del usuario participante en el test. Puesto que su papel es más relevante como parte del grupo de técnicas que forman parte de los Estudios de Seguimiento de Sistemas Instalados, se ha incluido como parte de este último grupo en vez del tratado en el presente punto.

5.3.3.3 Estudios de Seguimiento de Sistemas Instalados

En labores de mantenimiento y/o extensión de sistemas ya implantados en la organización del usuario, las tareas relacionadas con los requisitos y con la evaluación de la situación actual y los problemas a resolver, tienen unas características propias al disponerse de mucha más información que cuando se desarrolla un sistema partiendo de cero. En lo referente a la usabilidad, hemos seleccionado tres técnicas que permiten evaluar la usabilidad de sistemas ya instalados, cuya contribución a las labores de mantenimiento y/o extensión se describe a continuación:

- **Registro del Uso:** El uso de esta técnica está íntimamente relacionado con el Análisis de Tareas realizado como parte de la Educación y Análisis de Requisitos. Puesto que se ha modelado cómo se espera que sea el uso dado al sistema, tener mecanismos de registro del uso real permite al equipo de desarrollo observar las desviaciones entre el uso esperado y el real. Este tipo de información genera información objetiva y detallada sobre posibles problemas de usabilidad, funcionalidades con menos uso del esperado, o patrones de interacción que se repiten y que llevan a errores del usuario.
- **Cuestionarios, Entrevistas y Encuestas:** Estas técnicas proporcionan la información relativa a la satisfacción subjetiva del usuario. En las entrevistas flexibles se puede obtener, además del valor de dicho atributo de usabilidad, información acerca de las áreas del sistema con mayores deficiencias de usabilidad, o en las que es preciso trabajar con mayor prioridad debido a la importancia que tienen para el usuario.
- **Retroalimentación del Usuario:** Todas las técnicas tratadas anteriormente otorgan al usuario un papel pasivo, en el que no parte de él o ella la iniciativa de aplicar la técnica. En la familia de técnicas de Retroalimentación del Usuario, sin embargo, es el usuario quien, debido a algún problema o deficiencia en el sistema software que está usando, toma la iniciativa de acudir a la organización de desarrollo de software para informar del asunto. Cuando se cuenta con servicio de atención en línea o de ayuda, la información recopilada acerca de las consultas de los usuarios resulta de gran utilidad a la hora de identificar y priorizar los problemas de usabilidad a tratar con mayor rapidez.

5.3.4 Resumen

Como resumen de la asignación de técnicas IPO a tipos de actividades IS, la Tabla 5.1 presenta dichas técnicas agrupadas según el tipo de actividad al que corresponden. Nótese que las técnicas de Evaluación por Expertos aparecen tanto en el grupo de actividades del mismo nombre en el apartado de evaluación, como en la Validación de Requisitos en el apartado de análisis. Aquellas técnicas IPO seleccionadas con la valoración de "muy útil" aparecen con fondo blanco en la columna derecha de la tabla, mientras que aquellas técnicas con la valoración "útil" aparecen con fondo gris.

Tabla 5.1- Técnicas IPO Agrupadas según el Tipo de Actividad IS en el que su Aplicación Resulta más Útil

Tipo de Actividad IS		Técnica IPO
Análisis (Ingeniería de Requisitos)	Educción y Análisis de Requisitos	<i>Card Sorting</i>
		Análisis Competitivo
		Diagramas de Afinidad
		Investigación Contextual
		JEM
	Análisis de Usuarios*	Observación Etnográfica
		Personas
		Mapa de Roles de Usuario
		Perfiles de Usuarios
	Análisis de Tareas*	Casos de Uso Esenciales
		Escenarios de Tareas
		HTA
	Desarrollo del Concepto del Producto*	Escenarios y <i>Storyboards</i>
Prototipado*	Tormenta de Ideas Visual	
Prototipado*	Prototipos de Papel	
Especificación de Requisitos	Especificaciones de Usabilidad	
Validación de Requisitos	Inspecciones	
	Evaluación Heurística	
	Inspecciones Colaborativas	
	Recorrido Cognitivo	
	Recorrido Pluralístico	
Diseño	Diseño de la Interacción	Árboles de Menús
		Diagramas de Transición de Estados de la Interfaz
		Guía de Estilo del Producto
		Mapa de Navegación
		Modelo del Contenido de la Interfaz
		Análisis de Impacto
		Organización de la Ayuda según Casos de Uso
Evaluación	Evaluación por Expertos	Inspecciones
		Evaluación Heurística
		Inspecciones Colaborativas
		Recorrido Cognitivo
		Recorrido Pluralístico
	Test de Usabilidad	Pensar en Voz Alta
		Información Post-Test
		Medición del Rendimiento
	Estudios de Seguimiento de Sistemas Instalados	Test de Usabilidad en Laboratorio
		Retroalimentación del Usuario
		Cuestionarios, Entrevistas y Encuestas
		Registro del Uso

* No se trata de tipos de actividades IS, sino actividades IPO incluidas para ofrecer al desarrollador una visión estructurada de las 15 técnicas IPO que pueden aplicarse en actividades de Educción y Análisis de Requisitos.

Capítulo 6

Momentos de Aplicación de Técnicas IPO en el Desarrollo

6.1 Introducción

El enfoque adoptado para la integración de la usabilidad en el proceso software ha sido proporcionar un marco de integración que sirva de base para la toma de decisiones acerca de qué técnicas incluir en el desarrollo, y en qué actividades. Sin embargo, estas recomendaciones no son suficientes. Es necesario garantizar que el proceso de la organización cumple ciertas condiciones necesarias para lograr un producto con un nivel de usabilidad aceptable. Estos prerrequisitos, como veremos a continuación, se reducen a estar basado en un enfoque de desarrollo iterativo.

Por otro lado, un proceso iterativo se lleva a cabo en ciclos, pero no todos los ciclos son iguales. Mientras que en un desarrollo según el ciclo de vida en cascada las fases del desarrollo y las actividades del mismo coinciden, en un desarrollo iterativo no es así. Cada ciclo de un desarrollo iterativo implica trabajar en distintos tipos de actividades y, además, el esfuerzo relativo y el énfasis puesto en cada tipo de actividad cambia a lo largo del tiempo de desarrollo [Larman, 04]. En los primeros ciclos se dedica un mayor esfuerzo a labores relacionadas con delimitar el problema y establecer las líneas básicas de la solución, mientras que según va avanzando el desarrollo las tareas de diseño e implementación adquieren más peso. Además de la asignación de técnicas a actividades, realizada en el capítulo anterior, es necesario ofrecer a los desarrolladores una guía acerca de los momentos o etapas del desarrollo en los que la aplicación de cada técnica IPO puede resultar más útil. En este capítulo se detalla dicha guía acerca de los momentos de aplicación de las técnicas IPO en un desarrollo iterativo.

En primer lugar, se realiza en la sección 6.2 el estudio sobre condiciones mínimas a satisfacer por el proceso de desarrollo establecido en la organización. A continuación, en la sección 6.3, se establece el esquema de etapas en un proceso iterativo, el cual servirá de base sobre la que desarrollar la guía. En la sección 6.4 se detalla la clasificación de técnicas seleccionadas

según su grado de adecuación a cada etapa del desarrollo. Finalmente, la sección 6.5 incluye un cuadro-resumen de la guía sobre momentos de aplicación.

6.2 Prerrequisitos sobre el Proceso de Desarrollo

Frente a la decisión de qué modelo de proceso de desarrollo elegir para basar la propuesta de integración de la usabilidad que constituye el presente trabajo, hemos optado por no ligar la solución a ningún modelo de proceso concreto, para no limitar la aplicabilidad de la solución a aquellas organizaciones que sigan el modelo de proceso elegido. Así, el objetivo es proporcionar una solución suficientemente flexible como para encajar con la diversidad de procesos de desarrollo vigentes en las organizaciones de construcción de software.

Sin embargo, de cara a que el proceso resultante tras la integración cumpla adecuadamente con el objetivo de construir software con un nivel de usabilidad apropiado, es necesario estudiar qué características debe tener el proceso de desarrollo original. La cualidad que queremos alcanzar con las modificaciones propuestas por el marco de integración es que el proceso modificado sea centrado en el usuario. Por tanto, vamos a estudiar qué características de un proceso de ese tipo pueden obtenerse por la adición de actividades y técnicas (y por tanto, las puede proporcionar el marco de integración) y cuáles son intrínsecas al proceso (y por tanto, son prerrequisitos para el proceso original). Una vez identificadas las características intrínsecas al proceso, éstas pueden utilizarse por cualquier organización interesada en la integración de la usabilidad para decidir si puede realizarse tomando como base su proceso de desarrollo actual, o, por el contrario, debe considerar migrar a otro tipo de proceso, si realmente tiene interés en cumplir objetivos de usabilidad.

6.2.1 Características de un Proceso Centrado en el Usuario

El campo de la IPO, aparte de la diferencia terminológica con la IS (algunos autores hablan de "diseño" centrado en el usuario en vez de "proceso" centrado en el usuario), ofrece distintas visiones de lo que constituye un proceso centrado en el usuario. De todas formas, hay un núcleo común que presentamos en el estudio que se detalla en la presente sección. Las fuentes en las que nos hemos basado para el presente estudio coinciden con las consideradas en el estudio de actividades del capítulo 4.

Hix y Hartson [Hix, 93] describen el diseño centrado en el usuario de una forma poco precisa, al indicar que se trata de desarrollar la interacción desde el punto de vista del usuario, en vez de desde el punto de vista del sistema. Sin embargo, para describir su aproximación al desarrollo de la interacción con el usuario nombran los siguientes principios:

- El desarrollo debería incluir pruebas empíricas desde fases tempranas y de forma continua, centradas en usuarios apropiados realizando tareas representativas.
- Según avanza el desarrollo, debería incorporar procedimientos de refinamiento iterativo y análisis coste/beneficio para determinar los cambios más efectivos a realizar en el diseño de la interacción con el usuario.
- El proceso de gestión debería verificar y controlar el ciclo de vida general del desarrollo y asignar responsabilidades para cada paso.

Nielsen [Nielsen, 93] no describe explícitamente las características de la ingeniería de usabilidad, y no utiliza el término diseño centrado en el usuario. Sí plantea como una de las prácticas base de su enfoque el diseño iterativo. En este sentido, se refiere a un estudio realizado por el autor mediante una encuesta realizada entre especialistas en ingeniería de usabilidad [Nielsen, 92], que refleja una opinión mayoritaria acerca del diseño iterativo y el

análisis de las tareas que el usuario realiza actualmente, como las dos prácticas de usabilidad más importantes.

Cuando Preece et al. [Preece, 94] se refieren al diseño (proceso) centrado el usuario, indican que debe tener las siguientes características:

- Implicar a usuarios al máximo posible de forma que puedan influir en el diseño,
- integrar el conocimiento y la experiencia de las distintas disciplinas que contribuyen al diseño IPO,
- y ser altamente iterativo de forma que se puedan realizar pruebas para comprobar que el diseño verdaderamente cumple los requisitos del usuario.

Wixon y Wilson [Wixon, 97] basan su marco de ingeniería de usabilidad para el diseño y evaluación de productos en el establecimiento de objetivos de usabilidad y la continua evaluación de los mismos para mejorar el nivel de usabilidad del producto. En cuanto al proceso, destacan el diseño y pruebas iterativos. En concreto, indican que "las pruebas iterativas son un paso esencial en la ingeniería de usabilidad".

Shneiderman [Shneiderman, 98] defiende que un proceso que se preocupa por la usabilidad necesita ser no jerárquico en el sentido de que no sea estrictamente ni *top-down* ni *bottom-up*; y que es radicalmente transformacional, lo que implica la producción soluciones interinas que pueden acabar por no jugar ningún papel en el diseño final. Esta definición involucra un enfoque fuertemente iterativo.

En una definición que adopta un punto de vista diferente, Constantine y Lockwood [Constantine, 99] definen los elementos de un enfoque centrado en el uso como sigue:

- Guías de diseño pragmáticas.
- Un proceso de diseño dirigido por modelos.
- Actividades de desarrollo organizadas.
- Mejora iterativa.
- Medidas de calidad.

El estándar ISO 13407 sobre Procesos de Desarrollo Centrados en el Humano para Sistemas Interactivos [ISO13407, 99] indica que la incorporación de un enfoque centrado en el humano se caracteriza por lo siguiente:

- La implicación activa de usuarios y una clara comprensión de los requisitos de usuarios y sus tareas,
- una adecuada asignación de funciones entre los usuarios y la tecnología,
- la iteración de soluciones de diseño, y
- un diseño multidisciplinar.

Mayhew [Mayhew, 99] resume la filosofía de su ciclo de vida de la ingeniería de usabilidad en los siguientes puntos:

- El diseño de la interfaz de usuario es clave.
- La integración de la ingeniería de usabilidad con la IS debe ser particularizada.
- El análisis de requisitos vale la pena.
- El diseño puede aproximarse en un proceso estructurado de descomposición (*top-down*).
- El diseño, las pruebas y el desarrollo deberían ser iterativos.
- El ciclo de vida completo puede ser estratificado en subconjuntos de funcionalidad.

- Hay una variedad de técnicas para llevar a cabo cada tarea del ciclo de vida.
- Las técnicas alternativas hacen que el ciclo de vida sea flexible y adaptable.
- Una implementación óptima del ciclo de vida requiere participación completa de equipos multidisciplinares.

6.2.2 Condiciones de un Proceso para Integrar la Usabilidad

De todas las características reseñadas en la sección anterior, podemos extraer cuatro elementos clave que constituyen la esencia del proceso centrado en el usuario:

- Implicación de usuarios,
- adecuada comprensión del usuario y de las tareas que realiza,
- conocimiento multidisciplinar,
- y un proceso iterativo.

La última característica (un proceso iterativo) es la única citada en todas las fuentes consultadas, y además se trata de la única característica intrínseca del proceso de desarrollo. Un proceso que no está basado en un enfoque iterativo no puede convertirse en iterativo por el mero hecho de añadir una serie de actividades y/o técnicas.

El resto de características se pueden conseguir mediante la incorporación al proceso de las técnicas que forman el marco de integración que se propone en el presente trabajo de investigación, según se justifica a continuación:

- **Implicación de usuarios.** Para conseguir un proceso centrado en el usuario es necesario algún grado de implicación de usuarios representativos. Es preciso destacar que la participación de usuarios en el desarrollo supone un cambio importante en la mentalidad de los desarrolladores y en su forma de abordar los problemas. Por ejemplo, los modelos y diagramas con los que se trabaja no deberían requerir un conocimiento técnico profundo, de modo que los usuarios puedan hacer aportaciones al diseño, y es preciso asegurar que la voz de los usuarios no quede relegada frente a las opiniones subjetivas de los técnicos. Ciertas técnicas de usabilidad cumplen este papel, según se ha destacado en secciones anteriores, cubriendo así la necesidad de implicación de usuarios en el proceso.
- **Adecuada comprensión del usuario y de las tareas que realiza.** Aunque la IS se preocupa especialmente por el tema de los requisitos, la importancia de la comprensión del usuario visto como parte de un contexto más amplio es clave para la usabilidad del producto final. Se trata de abordar la comprensión del problema con una mirada más amplia de los meros requisitos funcionales, de modo que se afronten las necesidades últimas del usuario. Por otro lado, la comprensión de las tareas que realiza el usuario, además de contribuir al objetivo general de comprensión del usuario, proporciona una base sobre la que diseñar tareas que encajen con la forma habitual de hacer las cosas del usuario. La adición de técnicas IPO en las actividades de requisitos contribuye a la adecuada comprensión del usuario y sus tareas.
- **Conocimiento multidisciplinar.** Para comprender al usuario adecuadamente en su contexto de trabajo y organizacional, y para proveerle de una interacción adecuada a sus características y limitaciones, es necesario tomar en consideración conocimientos de diversas disciplinas. Las técnicas IPO que forman parte del marco de integración

propuesto en el presente trabajo de investigación están imbuidas de dicho conocimiento multidisciplinar por pertenecer al campo de la IPO, y, por tanto, su aplicación cumple este requisito. No obstante, queremos precisar que la propuesta que realizamos no pretende que los ingenieros software "sustituyan" a expertos de otras disciplinas para realizar las actividades del desarrollo que les son propias. Ciertas técnicas requerirán de expertos en IPO para su aplicación, mientras que otras técnicas más sencillas sí podrían ser aplicadas por ingenieros software si se les da la formación adecuada, en el caso de no disponer de suficiente personal con formación básica en IPO.

Así, la única característica identificada como intrínseca al proceso y, por tanto, necesaria en cualquier proceso de desarrollo candidato a la integración de la usabilidad por medio del marco aquí propuesto, es el estar basado en un enfoque iterativo. La complejidad del lado humano en la interacción entre una persona y un sistema informático hace casi imposible la creación de un diseño adecuado al primer intento. Los procesos iterativos son práctica común en la IPO. Por su parte, en la IS se trata de un enfoque de desarrollo reciente pero con especial prominencia en la literatura actual de la IS, por lo que su identificación como prerequisite para el proceso de desarrollo vigente no resulta ajena a las mejores prácticas actuales de la IS.

6.3 Tipos de Ciclos en un Proceso Iterativo

Debido a que cada método o proceso definido en una organización tiene su propia nomenclatura de fases e hitos del desarrollo, consideramos necesario establecer un marco genérico según el cual se puedan reflejar para cada técnica IPO las indicaciones referentes al momento de aplicación. Vamos a diferenciar tres etapas en los ciclos de un proceso iterativo: Ciclos iniciales, ciclos centrales y ciclos de evolución. Esta división particular en etapas se lleva a cabo por las siguientes razones:

- En los procesos de desarrollo de la IPO se trabaja al inicio del desarrollo en torno a la elaboración de un adecuado concepto del producto. De hecho, en cualquier proceso iterativo los **ciclos iniciales** se dedican a los puntos identificados como de mayor riesgo, y en el tema de la usabilidad, el no estar desarrollando un producto con un concepto adecuado constituye uno de los mayores riesgos. Ciertas técnicas IPO se centran en la producción y/o evaluación de las tentativas iniciales de diseño que se manejan y, por tanto, su aplicación está especialmente indicada en los ciclos iniciales. Una vez el concepto del producto se ha establecido, no suele ser necesario aplicar dichas técnicas en el resto del desarrollo. Por esta razón, se distingue entre los ciclos iniciales y los ciclos centrales, los cuales comienzan una vez se ha centrado el desarrollo en un concepto del producto concreto. Así, el hito que define el fin de esta etapa se denomina **Concepto del Producto Establecido**.
- Los **ciclos centrales** se definen como aquellos que se llevan a cabo entre los ciclos iniciales y los de evolución. En estos ciclos se trabaja ya con un concepto del producto compartido por todos los integrantes del equipo de desarrollo. Estos ciclos suelen formar el grueso del desarrollo.
- Finalmente, ciertas técnicas (principalmente de evaluación de la usabilidad) pueden aplicarse únicamente cuando se dispone de una versión del sistema instalada en el entorno de trabajo del usuario final. Así, para poder indicar a partir de qué momento estas técnicas son aplicables, definimos los **ciclos de evolución** como aquellos que se llevan a cabo una vez se tiene una parte del software instalada en el entorno de uso previsto. El hito que define el comienzo de esta etapa se denomina, por tanto, **Parte del Sistema Instalada y en Funcionamiento**.

En la Figura 6.1 se puede observar una representación de las tres etapas en las que se han dividido los ciclos de un proceso iterativo.

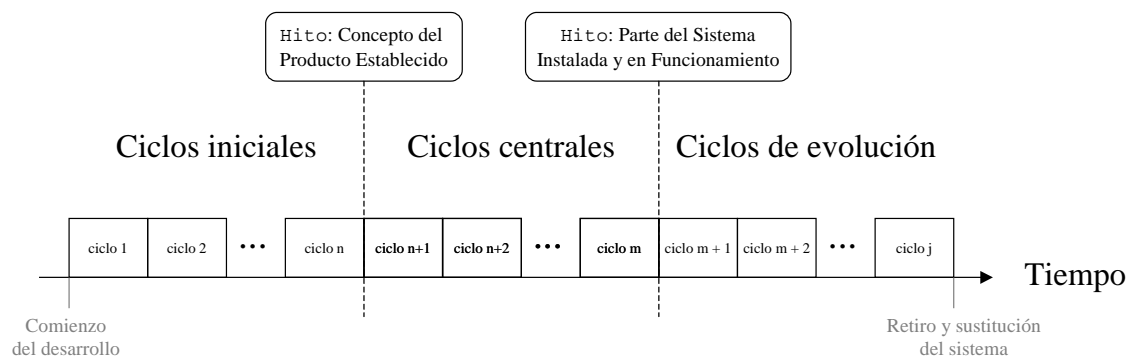


Figura 6.1 División de los Ciclos Iterativos en Etapas

6.4 Asignación de Técnicas IPO a Etapas del Desarrollo

Vamos a reflejar para cada técnica IPO seleccionada su grado de adecuación a cada una de estas tres etapas en un proceso iterativo. Para cada etapa clasificaremos las técnicas según los siguientes tres valores de idoneidad:

- **Especialmente apropiada.** Significa que dicha técnica puede resultar de la máxima utilidad cuando se aplica en esta etapa del desarrollo.
- **Neutra.** La técnica puede aplicarse en dicha etapa, sin destacar la etapa como la más apropiada frente a otros momentos del desarrollo.
- **No es habitual.** Significa que no es la etapa más apropiada donde corresponde aplicar esta técnica y existen otras técnicas más apropiadas. Por tanto, no resulta habitual su aplicación en esta etapa.

A continuación se detalla la clasificación de las técnicas por etapas según los valores de idoneidad para cada una de ellas.

6.4.1 Técnicas para los Ciclos Iniciales

En los métodos IPO se puede observar que en los momentos iniciales del desarrollo se da un mayor énfasis a las actividades relacionadas con la observación y estudio de los usuarios en su entorno habitual de trabajo, con el fin de identificar sus características principales y las de las tareas que realizan. De hecho, un adecuado análisis de los usuarios y sus tareas es uno de los puntos básicos de un enfoque centrado en el usuario. Otro tipo de actividad que se identifica en el campo de la IPO con los esfuerzos iniciales del desarrollo es la relacionada con el desarrollo del concepto del producto. En proyectos con cierto carácter innovador (como opuesto a la mecanización rutinaria de tareas que se realizan actualmente), se identifica como importante la obtención de una visión común, por parte de todo el equipo de desarrollo, de los principios básicos según los cuales va a funcionar el sistema software. Esta visión común o esquema general de funcionamiento es importante que se establezca en momentos tempranos del desarrollo, puesto que sirve de puntal para todo el resto del desarrollo.

En la Tabla 6.1 se detalla la clasificación de las técnicas IPO seleccionadas respecto a su adecuación para los ciclos iniciales.

Tabla 6.1 - Adecuación de las Técnicas IPO Seleccionadas a los Ciclos Iniciales

Etapa	Valoración	Técnicas Especialmente Apropriadas	Técnicas Neutras	Técnicas No Habituales
Ciclos Iniciales	Muy Útiles	<ul style="list-style-type: none"> • Escenarios y <i>Storyboards</i> • Personas • Prototipos de Papel 	<ul style="list-style-type: none"> • Árboles de Menús • <i>Card Sorting</i> • Casos de Uso Esenciales • Especificaciones de Usabilidad • Inspecciones 	<ul style="list-style-type: none"> • Pensar en Voz Alta • Retroalimentación del Usuario
	Útiles	<ul style="list-style-type: none"> • Análisis Competitivo • Diagramas de Afinidad • Inspecciones Colaborativas • HTA • Investigación Contextual • Mapa de Roles de Usuario • Observación Etnográfica • Perfiles de Usuario • Recorrido Pluralístico • Tormenta de Ideas Visual 	<ul style="list-style-type: none"> • Escenarios de Tareas • Diagramas de Transición de Estados de la Interfaz • Evaluación Heurística • JEM • Mapa de Navegación • Organización de la Ayuda según Casos de Uso • Recorrido Cognitivo 	<ul style="list-style-type: none"> • Análisis de Impacto • Cuestionarios, Entrevistas y Encuestas • Guía de Estilo del Producto • Información Post-Test • Medición del Rendimiento • Modelo del Contenido de la Interfaz • Registro del Uso • Test de Usabilidad en Laboratorio

6.4.1.1 Técnicas Especialmente Apropriadas

Las técnicas que son especialmente apropiadas para los ciclos iniciales son, en primer lugar, las técnicas de observación, que permiten un primer contacto con los usuarios y sus tareas; y, en segundo lugar, las técnicas que ayudan a establecer el concepto del producto.

Las técnicas consideradas muy útiles que son especialmente apropiadas en esta etapa son las siguientes:

- **Escenarios y *Storyboards*.** La finalidad de estas técnicas es hacer pensar al equipo de desarrollo en el contexto de uso, en temas tangenciales a la pura funcionalidad que va a tener que usar el usuario, y que van más allá de los requisitos clásicos. Por tanto, su uso es especialmente indicado en etapas tempranas, porque sirven para no perder de vista el contexto de uso previsto en todo trabajo de diseño posterior.
- **Personas.** A pesar de que esta técnica requiere de un estudio a fondo de los usuarios potenciales, el resultado tiene como finalidad guiar todo el proceso de diseño posterior, por tanto, parece razonable que se aplique de forma prioritaria en los ciclos iniciales del desarrollo.

- **Prototipos de Papel.** Frente a los prototipos software tradicionalmente usados en la IS, los cuales tienen un coste relativamente alto, los prototipos de papel tienen un coste muy bajo, y por esta razón están especialmente indicados en momentos muy tempranos del desarrollo, en los que se tienen ideas de diseño no elaboradas que se quieren contrastar con el usuario.

Las técnicas consideradas simplemente útiles que son especialmente apropiadas en esta etapa son las siguientes:

- **Análisis Competitivo.** En la búsqueda de las características que debería tener el producto software a desarrollar, juega un papel relevante el análisis de sistemas similares. Puesto que un análisis competitivo puede resultar muy útil para el proceso de generación de ideas de diseño del concepto del producto, consideramos que esta técnica es especialmente apropiada para los ciclos iniciales.
- **Diagramas de Afinidad.** Esta técnica puede servir para dos objetivos. Por un lado, se utiliza para organizar las notas libres tomadas durante una investigación contextual o etnográfica y, como tal, debería aplicarse junto a estas técnicas. Por otro lado, puede servir para organizar ideas surgidas en una reunión tipo Tormenta de Ideas, las cuales suelen realizarse también en etapas tempranas del desarrollo para definir el concepto del producto.
- **HTA.** Puesto que esta técnica sirve para modelar la forma en la que los usuarios organizan sus actividades y qué medios utilizan para llevarlas a cabo, resulta especialmente apropiada para su aplicación en los ciclos iniciales. Esto es debido a que es en estos ciclos cuando se realiza el grueso de observación de los usuarios para intentar comprender mejor sus esquemas mentales de realización de actividades.
- **Inspecciones Colaborativas y Recorrido Pluralístico.** Para la aplicación de estas técnicas no es necesaria la existencia de software, puesto que pueden aplicarse sobre prototipos en papel. Debido a que permiten la participación en la evaluación de todas las partes involucradas en el desarrollo, resultan especialmente apropiadas para los ciclos iniciales, puesto que para el establecimiento del concepto del producto es importante que todas las partes involucradas hayan evaluado dicho concepto.
- **Investigación Contextual.** Esta técnica ofrece una alternativa a la tradicional técnica de entrevistas aplicada en la ingeniería de requisitos. Puesto que ayuda también a comprender la forma de razonar del usuario y sus verdaderos objetivos y motivaciones, es especialmente adecuada en los momentos iniciales del desarrollo.
- **Mapa de Roles de Usuario y Perfiles de Usuario.** La modelización de los usuarios tipo, su entorno y sus principales tareas, es una actividad que está especialmente indicada para realizarse al principio del desarrollo, aunque también pueda refinarse y completarse en etapas posteriores.
- **Observación Etnográfica.** La observación de los usuarios para aprender cuál es su modo de trabajar y de razonar es adecuada para los momentos iniciales del desarrollo, pues ayuda a educir las características relevantes de los usuarios, que habrá que tener en cuenta en todo el desarrollo.
- **Tormenta de Ideas Visual.** Como se ha mencionado más arriba, las reuniones tipo Tormenta de Ideas son especialmente apropiadas para el inicio de un proyecto, para evaluar posibles soluciones sin ningún tipo de condicionante previo.

6.4.1.2 Técnicas Neutras

Algunas técnicas son aplicables en los ciclos iniciales, aunque no se pueden destacar como especialmente relevantes en esta etapa del desarrollo frente a otras etapas posteriores. La mayoría de las técnicas de modelado caen dentro de esta categoría, puesto que, al ir avanzando el desarrollo iterativo, se van refinando los modelos y extendiéndose con los elementos nuevos tratados en cada ciclo. La mayoría de técnicas que se aplican en actividades de diseño también pueden aplicarse en los ciclos iniciales, junto con algunas técnicas de evaluación de la usabilidad que no requieren un sistema funcionando para poder llevarse a cabo.

Las técnicas muy útiles aplicables de forma neutra en los ciclos iniciales son las siguientes:

- **Árboles de menús y *Card Sorting*.** Estas dos técnicas sirven para modelar y diseñar los menús de la aplicación, respectivamente. Los menús de la aplicación pueden considerarse tanto en los ciclos iniciales, como en ciclos posteriores del desarrollo. Por esta razón no se destacan especialmente estas técnicas en los ciclos iniciales, a pesar de que pueden aplicarse en ellos.
- **Casos de Uso Esenciales.** Las tareas que el usuario va a realizar con el sistema se modelan a lo largo de todo el desarrollo, no tienen una etapa especialmente indicada.
- **Especificaciones de Usabilidad.** Las Especificaciones de Usabilidad requieren de unas tareas ya definidas sobre las que basar la especificación. En la medida en que se puede trabajar en los ciclos iniciales con algunas tareas que ayuden a formar el concepto del producto, esta técnica puede aplicarse en los ciclos iniciales.
- **Inspecciones.** Esta técnica de evaluación de la usabilidad puede aplicarse en cualquier ciclo del desarrollo, pues puede llevarse a cabo sobre representaciones de las pantallas de la IU.

Las técnicas etiquetadas como útiles aplicables de forma neutra en esta etapa son las siguientes:

- **Diagramas de Transición y Mapa de Navegación.** Estas técnicas pueden emplearse para describir grosso modo la transición entre las principales ventanas del sistema que se está desarrollando. Este tipo de representación permite visualizar la estructura de alto nivel de la interacción, la cual es una forma de transmitir el concepto del producto. Puesto que estas técnicas pueden servir en las discusiones sobre el concepto del producto en los momentos iniciales del desarrollo, consideramos que se pueden aplicar en los ciclos iniciales, aunque también se aplican en ciclos posteriores en los que se da un mayor énfasis al diseño de aspectos concretos de la navegación.
- **Escenarios de Tareas.** Al igual que ocurre con la técnica ya mencionada de Casos de Uso Esenciales, por dedicarse esta técnica al estudio de las tareas que el usuario va a realizar con el sistema, su empleo no tiene una etapa especialmente indicada.
- **Evaluación Heurística.** Al igual que las Inspecciones, esta técnica de evaluación de la usabilidad puede aplicarse en cualquier etapa del desarrollo.
- **JEM.** Esta técnica está enfocada al modelado de usuarios y (sobre todo) de tareas, de forma colaborativa, incluyendo a usuarios, desarrolladores y otras posibles partes implicadas. Puesto que el modelado se va a realizar durante todo el desarrollo, no

parece indicado destacar esta técnica como especialmente apropiada para los ciclos iniciales, sino que puede aplicarse durante todo el desarrollo.

- **Organización de la Ayuda según Casos de Uso.** El diseño del subsistema de ayuda se puede abordar en cualquier momento del desarrollo, por tanto esta técnica, que ofrece una forma de organizar la estructura del subsistema de ayuda, puede aplicarse en cualquier etapa del desarrollo.
- **Recorrido Cognitivo.** Esta técnica de evaluación de la usabilidad puede realizarse sobre una representación de las pantallas a evaluar. Al requerir únicamente prototipos en papel pueden aplicarse en los ciclos iniciales, aunque no se cuente con ninguna parte del sistema software ya implementada.

6.4.1.3 Técnicas No Habituales

En esta sección se detallan las técnicas seleccionadas que no se han mencionado en los dos apartados anteriores, por no ser habitualmente aplicadas en los ciclos iniciales

Las técnicas muy útiles cuya aplicación no es habitual en los ciclos iniciales son las siguientes:

- **Pensar en Voz Alta y Retroalimentación del Usuario.** Se trata de técnicas de evaluación de usabilidad que requieren un sistema funcionando. Este tipo de técnicas son poco habituales en los ciclos iniciales del desarrollo porque no suele realizarse un prototipo software hasta los ciclos centrales.

Las técnicas etiquetadas como útiles cuya aplicación no es habitual en los ciclos iniciales son las siguientes:

- **Análisis de Impacto.** Esta técnica se emplea para decidir qué problemas de usabilidad se abordan primero, esto es, para establecer una ordenación por prioridad entre los mismos. Para su aplicación se requiere una lista de los problemas de usabilidad identificados en actividades de evaluación de la usabilidad. Puesto que en los ciclos iniciales se trabaja principalmente en torno a la definición del concepto del producto, no suele bajarse a problemas concretos de usabilidad hasta los ciclos centrales.
- **Guía de Estilo del Producto.** Cuando se establece una Guía de Estilo del Producto, suele tener un grado de detalle tal que no se aborda su realización hasta que el concepto general del sistema está establecido, por tanto no es habitual su aplicación en la etapa inicial del desarrollo.
- **Modelo del Contenido de la Interfaz.** Esta técnica requiere de un esfuerzo de realización que la hace relativamente costosa. Por tanto, no resulta habitual su aplicación antes de haber establecido el concepto del producto. Resulta especialmente apropiada para ciclos posteriores, pues su finalidad consiste en transmitir a los programadores de la IU qué elementos debe contener ésta.
- Como se ha comentado más arriba para las técnicas de Pensar en Voz Alta y Retroalimentación del Usuario, las técnicas de evaluación de usabilidad que requieren un sistema funcionando son poco habituales en los ciclos iniciales del desarrollo. Las técnicas incluidas en este grupo son las siguientes:
 - Cuestionarios, Entrevistas y Encuestas.
 - Información Post-Test.

- Medición del Rendimiento.
- Registro del Uso.
- Test de Usabilidad en Laboratorio.

6.4.2 Técnicas para los Ciclos Centrales

Ciertas técnicas IPO tienen un carácter de trabajo detallado, que pueden considerarse como especialmente indicadas para los ciclos centrales, aquellos que comienzan una vez el concepto del producto se ha decidido y, por tanto, puede abordarse una elaboración detallada referente a aspectos de la interacción que se habían tratado a un nivel más abstracto en los ciclos iniciales. Por otro lado, dado que estos ciclos pueden identificarse con el grueso del desarrollo, la mayoría de técnicas son apropiadas para estos ciclos centrales, como se verá más abajo.

En la Tabla 6.2 se detalla la clasificación de las técnicas IPO seleccionadas respecto a su adecuación para los ciclos centrales.

6.4.2.1 Técnicas Especialmente Apropriadas

Las técnicas que son especialmente apropiadas para los ciclos centrales son aquellas que requieren una elaboración más detallada, y que no suelen ser útiles antes de haberse decidido un concepto del producto específico. Son técnicas que están relacionadas con el diseño detallado de la interacción, con su parte visual concreta. Se trata de las siguientes dos técnicas, etiquetadas como útiles:

- **Guía de Estilo del Producto.** La Guía de Estilo está íntimamente relacionada con los elementos visuales de la IU, los cuales se elaboran en el diseño detallado de la interacción que se lleva a cabo en los ciclos centrales. Por esta razón consideramos que es especialmente apropiada para su aplicación en los ciclos centrales.
- **Modelo del Contenido de la Interfaz.** Puesto que la principal finalidad de esta técnica es servir de herramienta de comunicación entre los diseñadores de la interacción a alto nivel y el equipo que programa la IU, su aplicación está especialmente indicada en los ciclos centrales, cuando ya se está trabajando de una forma más organizada, frente a los ciclos iniciales, donde no es habitual elaborar especificaciones formales.

6.4.2.2 Técnicas Neutras

En esta sección se detallan las técnicas que son aplicables de forma neutra en los ciclos centrales, aunque no se pueden destacar como especialmente relevantes en esta etapa del desarrollo frente a otras etapas.

Las técnicas muy útiles que caen dentro de esta clasificación son las siguientes:

- **Árboles de menús y *Card Sorting*.** Estas técnicas, por su utilidad para el modelado de los elementos de la IU pueden aplicarse en los ciclos centrales, los cuales tendrán que incluir actividades de Diseño de la Interacción.
- **Casos de Uso Esenciales.** Como se ha indicado anteriormente, las actividades de modelado de tareas del sistema se llevan a cabo a lo largo de todo el desarrollo, no tienen una etapa especialmente indicada.

- **Especificaciones de Usabilidad.** Si en los ciclos centrales se definen las tareas que a va llevar a cabo el usuario con el sistema, puede aplicarse esta técnica para establecer las Especificaciones de Usabilidad correspondientes.

Tabla 6.2 - Adecuación de las Técnicas IPO Seleccionadas a los Ciclos Centrales

Etapa	Valora- ción	Técnicas Especialmente Apropiadas	Técnicas Neutras	Técnicas No Habituales
Ciclos Centrales	Muy Útiles	<i>No hay</i>	<ul style="list-style-type: none"> • Árboles de Menús • <i>Card Sorting</i> • Casos de Uso Esenciales • Especificaciones de Usabilidad • Inspecciones • Pensar en Voz Alta • Prototipos de Papel 	<ul style="list-style-type: none"> • Escenarios y <i>Storyboards</i> • Personas • Retroalimentación del Usuario
	Útiles	<ul style="list-style-type: none"> • Guía de Estilo del Producto • Modelo del Contenido de la Interfaz 	<ul style="list-style-type: none"> • Análisis Competitivo • Análisis de Impacto • Cuestionarios, Entrevistas y Encuestas • Diagramas de Transición de Estados de la Interfaz • Escenarios de Tareas • Evaluación Heurística • Información Post-Test • Inspecciones Colaborativas • JEM • Mapa de Navegación • Mapa de Roles de Usuario • Medición del Rendimiento • Organización de la Ayuda según Casos de Uso • Perfiles de Usuario • Recorrido Cognitivo • Recorrido Pluralístico • Registro del Uso • Test de Usabilidad en Laboratorio 	<ul style="list-style-type: none"> • Diagramas de Afinidad • HTA • Investigación Contextual • Observación Etnográfica • Tormenta de Ideas Visual

- **Inspecciones y Pensar en Voz Alta:** Entre las técnicas de Evaluación de Usabilidad, únicamente aquellas que requieren un sistema instalado en el futuro entorno de uso no pueden aplicarse en los ciclos centrales. Como estas dos técnicas no pertenecen a dicho grupo, se pueden aplicar en los ciclos centrales.

- **Prototipos de Papel.** Cualquier tipo de prototipado es adecuado para su aplicación en los ciclos centrales y, en concreto, también lo son aquellos de baja fidelidad como los prototipos de papel.

Las técnicas etiquetadas como útiles cuya aplicación calificamos de neutra en los ciclos centrales son las siguientes:

- **Análisis Competitivo.** Esta técnica, además de ayudar a desarrollar el concepto del producto, puede servir como fuente de inspiración para cualquier actividad de Diseño de la Interacción que se lleve a cabo en los ciclos centrales.
- **Análisis de Impacto.** Esta técnica es útil en el comienzo de un ciclo para decidir cuáles de los problemas de usabilidad identificados previamente se van a abordar en primer lugar. Por tanto, es plenamente aplicable en los ciclos centrales.
- **Diagramas de Transición de Estados de la Interfaz y Mapa de Navegación.** Las razones por las que la aplicación de estas técnicas es neutra en los ciclos centrales, son las mismas que para las técnicas de Árboles de Menús y *Card Sorting*.
- **Escenarios de Tareas y JEM.** Al igual que ocurre con los Casos de Uso Esenciales, estas técnicas no tienen una etapa especialmente indicada, por tratarse de técnicas de modelado de tareas del sistema.
- **Mapa de Roles de Usuario y Perfiles de Usuario.** Es posible que ciertos datos sobre los usuarios y/o su estructuración en tipos sea preciso refinarlos o completarlos en los ciclos centrales.
- **Organización de la Ayuda según Casos de Uso.** Como se mencionó más arriba, el diseño del subsistema de ayuda se puede abordar en cualquier momento del desarrollo, por tanto esta técnica también puede aplicarse en los ciclos centrales del desarrollo.
- **Recorrido Cognitivo y Recorrido Pluralístico.** Como se mencionó más arriba, estas técnicas son aplicables en todo el desarrollo, incluyendo los ciclos centrales.
- **Técnicas de evaluación de la usabilidad.** Son aplicables todas las técnicas excepto aquellas que requieren contar con un producto ya instalado en el contexto de uso para el que se ha desarrollado (únicamente la técnica de Retroalimentación del Usuario). Las técnicas incluidas en este grupo son las siguientes:
 - Cuestionarios, Entrevistas y Encuestas.
 - Evaluación Heurística.
 - Información Post-Test.
 - Inspecciones Colaborativas.
 - Medición del Rendimiento.
 - Recorrido Cognitivo y Pluralístico.
 - Registro del Uso.
 - Test de Usabilidad en Laboratorio.

6.4.2.3 Técnicas No Habituales

Las técnicas cuya aplicación no resulta habitual en los ciclos centrales pertenecen a dos grupos: Aquellas que están destinadas a obtener la información inicial sobre usuarios y a

elaborar el concepto del producto, y aquellas que necesitan para su aplicación de un sistema instalado en el entorno de uso previsto.

Las técnicas muy útiles cuya aplicación no resulta habitual en los ciclos centrales son las siguientes:

- **Escenarios y Storyboards.** Estas técnicas se aplican para elaborar el concepto del producto. Puesto que en los ciclos centrales dicho concepto está ya elaborado, su aplicación no es habitual en dichos ciclos.
- **Personas.** Esta técnica condensa las características principales de usuarios arquetipo, las cuales se emplean para que la elaboración del concepto del producto responda a las necesidades de dichos usuarios arquetípicos. Por esta razón no es habitual su aplicación tras los ciclos iniciales, aunque los productos que genera sí que se utilizan para guiar decisiones de diseño a lo largo de todo el desarrollo.
- **Retroalimentación del Usuario.** Esta técnica requiere un sistema instalado en el entorno de uso previsto, por tanto no se suele poder aplicar en los ciclos centrales.

Las técnicas etiquetadas como útiles cuya aplicación no resulta habitual en los ciclos centrales son las siguientes:

- **Diagramas de Afinidad y Tormenta de Ideas Visual.** Estas técnicas se aplican a la hora de elaborar el concepto del producto. Puesto que en los ciclos centrales dicho concepto está ya elaborado, no es habitual la aplicación de estas técnicas.
- **Observación Etnográfica, Investigación Contextual y HTA.** Sirven para obtener información sobre la forma de pensar del usuario y modelar su modo de trabajo, por lo que se aplican con mayor frecuencia en los ciclos iniciales y no es habitual su aplicación en ciclos posteriores.

6.4.3 Técnicas para los Ciclos de Evolución

Los ciclos de evolución tienen como única característica especial que ya se dispone de parte del sistema instalada en el entorno del usuario final. Por tanto, no se diferencian excesivamente de los ciclos centrales.

En la Tabla 6.3 se detalla la clasificación de las técnicas IPO seleccionadas respecto a su adecuación para los ciclos de evolución.

6.4.3.1 Técnicas Especialmente Apropriadas

Ciertas técnicas de evaluación de la usabilidad se basan en que el sistema o parte de él esté en uso. Por tanto, estas técnicas son especialmente apropiadas para su aplicación en los ciclos de evolución. Las técnicas que responden a estas características se detallan a continuación, primero aquella etiquetada como muy útil, y tras ella las etiquetadas como útiles: **Retroalimentación del Usuario** (muy útil), **Cuestionarios, Entrevistas y Encuestas** (útil), y **Registro del Uso** (útil).

6.4.3.2 Técnicas Neutras

Las técnicas aplicables de forma neutra en los ciclos de evolución coinciden con las de los ciclos centrales. Las únicas diferencias estriban en que aquellas técnicas especialmente

apropiadas para los ciclos centrales (Guía de Estilo del Producto y Modelo del Contenido de la Interfaz), son neutras en los ciclos de evolución, y viceversa con dos técnicas que son neutras en los ciclos centrales pero que hemos destacado como especialmente apropiadas para esta etapa (Cuestionarios, Entrevistas y Encuestas, y Registro del Uso).

Tabla 6.3 - Adecuación de las Técnicas IPO Seleccionadas a los Ciclos de Evolución

Etapa	Valoración	Técnicas Especialmente Apropriadas	Técnicas Neutras	Técnicas No Habituales
Ciclos de Evolución	Muy Útiles	<ul style="list-style-type: none"> • Retroalimentación del Usuario 	<ul style="list-style-type: none"> • Árboles de Menús • <i>Card Sorting</i> • Casos de Uso Esenciales • Especificaciones de Usabilidad • Inspecciones • Pensar en Voz Alta • Prototipos de Papel 	<ul style="list-style-type: none"> • Escenarios y <i>Storyboards</i> • Personas
	Útiles	<ul style="list-style-type: none"> • Cuestionarios, Entrevistas y Encuestas • Registro del Uso 	<ul style="list-style-type: none"> • Análisis Competitivo • Análisis de Impacto • Diagramas de Transición de Estados de la Interfaz • Escenarios de Tareas • Evaluación Heurística • Guía de Estilo del Producto • Información Post-Test • Inspecciones Colaborativas • JEM • Mapa de Navegación • Mapa de Roles de Usuario • Medición del Rendimiento • Modelo del Contenido de la Interfaz • Organización de la Ayuda según Casos de Uso • Perfiles de Usuario • Recorrido Cognitivo • Recorrido Pluralístico • Test de Usabilidad en Laboratorio 	<ul style="list-style-type: none"> • Diagramas de Afinidad • HTA • Investigación Contextual • Observación Etnográfica • Tormenta de Ideas Visual

Las técnicas muy útiles aplicables de forma neutra en los ciclos de evolución son las siguientes: Árboles de menús, *Card Sorting*, Casos de Uso Esenciales, Especificaciones de Usabilidad, Inspecciones, Pensar en Voz Alta, y Prototipos de Papel.

Las técnicas etiquetadas como útiles que son aplicables de forma neutra en los ciclos de evolución son las siguientes: Análisis Competitivo, Análisis de Impacto, Diagramas de Transición de Estados de la Interfaz, Escenarios de Tareas, Evaluación Heurística, Guía de Estilo del Producto, Información Post-Test, Inspecciones Colaborativas, JEM, Mapa de Navegación, Mapa de Roles de Usuario, Medición del Rendimiento, Modelo del Contenido de la Interfaz, Organización de la Ayuda según Casos de Uso, Perfiles de Usuario, Recorrido Cognitivo, Recorrido Pluralístico, y Test de Usabilidad en Laboratorio.

6.4.3.3 Técnicas No Habituales

Todas las técnicas cuya aplicación no es habitual en los ciclos centrales tampoco lo es en los ciclos de evolución, excepto algunas actividades de evaluación de la usabilidad que requieren un sistema instalado en el contexto de uso previsto, y que ya se han mencionado en el apartado de técnicas especialmente apropiadas.

Las técnicas muy útiles cuya aplicación no es habitual en los ciclos de evolución son las siguientes: Escenarios y *Storyboards*, y Personas.

Las técnicas etiquetadas como útiles cuya aplicación no es habitual en los ciclos de evolución son las siguientes: Diagramas de Afinidad, HTA, Investigación Contextual, Observación Etnográfica y Tormenta de Ideas Visual.

6.5 Adecuación de las Técnicas IPO a las Etapas de un Desarrollo Iterativo

La adecuación de cada técnica a cada momento de desarrollo, según se ha detallado en las secciones anteriores, se recoge en la Tabla 6.5. Las técnicas de usabilidad están organizadas según los tipos de actividades IS en los que se aplican. Para las celdas correspondientes a la adecuación de las técnicas a etapas del desarrollo se han utilizado tres tonos de gris para reflejar el grado de adecuación de cada técnica a cada momento del desarrollo, según ilustra la Tabla 6.4. El tono más oscuro indica que la técnica es especialmente apropiada para dicha etapa, el tono intermedio indica un carácter neutro, y el tono de gris más claro indica que la técnica no se aplica habitualmente en dicha etapa.

Tabla 6.4 - Gradación de la Adecuación de una Técnica a un Momento del Desarrollo

Contenido	Leyenda
	Especialmente apropiada
	Neutra
	No es habitual

Tabla 6.5 - Adecuación de las Técnicas IPO Seleccionadas a las Etapas de un Desarrollo Iterativo

Tipo de Actividad IS		Técnica IPO	Etapas en un Desarrollo Iterativo			
			Ciclos de Evolución	Ciclos Centrales	Ciclos de Evolución	
Análisis (Ingeniería de Requisitos)	Educción, Análisis y Negociación de Requisitos	<i>Card Sorting</i>				
		Análisis Competitivo				
		Diagramas de Afinidad				
		Investigación Contextual				
		JEM				
		Observación Etnográfica				
		Personas				
		Análisis de Usuarios*	Mapa de Roles de Usuario			
			Perfiles de Usuarios			
			Casos de Uso Esenciales			
	Análisis de Tareas*	Escenarios de Tareas				
		HTA				
	Desarrollo del Concepto del Producto*	Escenarios y <i>Storyboards</i>				
		Tormenta de Ideas Visual				
		Prototipado*	Prototipos de Papel			
		Especificación de Requisitos	Especificaciones de Usabilidad			
	Validación de Requisitos		Inspecciones			
			Evaluación Heurística			
Inspecciones Colaborativas						
Recorrido Cognitivo						
Recorrido Pluralístico						
Diseño	Diseño de la Interacción	Árboles de Menús				
		Diagramas de Transición de Estados de la Interfaz				
		Guía de Estilo del Producto				
		Mapa de Navegación				
		Modelo del Contenido de la Interfaz				
		Análisis de Impacto				
			Organización de la Ayuda según Casos de Uso			
			Inspecciones			
			Evaluación Heurística			
			Inspecciones Colaborativas			
Evaluación	Evaluación por Expertos	Recorrido Cognitivo				
		Recorrido Pluralístico				
		Pensar en Voz Alta				
		Información Post-Test				
		Medición del Rendimiento				
	Test de Usabilidad		Test de Usabilidad en Laboratorio			
			Retroalimentación del Usuario			
			Cuestionarios, Entrevistas y Encuestas			
	Estudios de Seguimiento de Sistemas Instalados		Registro del Uso			

* No se trata de tipos de actividades IS, sino actividades IPO incluidas para ofrecer al desarrollador una visión estructurada de las 15 técnicas IPO que pueden aplicarse en actividades de Educación y Análisis de Requisitos.

Capítulo 7

Integración de

las Técnicas IPO

en el Proceso

7.1 Introducción

El objetivo último del presente trabajo de investigación es servir a la organización de desarrollo de software que quiera integrar la usabilidad en su proceso en la consecución de tal objetivo. En concreto, servir a los ingenieros software que pretenden integrar las prácticas de usabilidad en sus actividades de desarrollo. Para lograr este objetivo es preciso ofrecerles la información usando la terminología y conceptos de desarrollo a los que están habituados, esto es, los de la IS. Además, es preciso presentar los resultados de investigación de una forma práctica y estructurada para su utilización directa por los desarrolladores.

En este capítulo se presentan los resultados del presente trabajo de investigación destilados de tal forma que pueden ser aplicados directamente por los desarrolladores. Hemos resumido en una serie de vistas la estructuración de los aspectos de usabilidad que forman el marco de integración de la usabilidad en el proceso de desarrollo. Dependiendo de la faceta del proceso en la que esté interesado el desarrollador puede acudir a la vista concreta que organiza los resultados según dicho criterio. Estas vistas no están pensadas como vías de acceso al marco de integración que se van a utilizar de forma independiente, sino que, por el contrario, ofrecen al ingeniero software una mayor flexibilidad en la difícil tarea de seleccionar las técnicas IPO que mejor pueden enriquecer su proceso de desarrollo. De esta forma, se da la posibilidad de acceder a la caracterización de técnicas que contiene el marco desde distintas dimensiones, para que se pueda obtener una visión global de los condicionantes que es preciso considerar en cada decisión.

Las dimensiones que pueden interesar desde el punto de vista de su proceso son el tipo de actividad y el momento del desarrollo, y desde el punto de vista de las técnicas IPO éstas ordenadas alfabéticamente. A continuación se detalla cada una de las vista (por técnicas, por actividades y por momentos del desarrollo) y se indican posibles usos del marco. Finalmente, la información que contiene el marco de integración con sus distintas vistas se ha empaquetado así mismo en una herramienta web, la cual se introduce en la última sección de este capítulo.

7.2 Vista por Técnicas

En el caso de estar considerando la inclusión o no de una determinada técnica IPO, el desarrollador está interesado en acceder a la información de la técnica buscándola por sí misma y no por su ubicación en un proceso de desarrollo. Así, esta vista presenta las técnicas organizadas por orden alfabético para permitir la localización de una técnica IPO concreta por su nombre. Un desarrollador puede conocer ya una técnica o conjunto de técnicas IPO que desea incluir en su proceso, o puede haber conocido las ventajas de una técnica en concreto y quiere conocer más sobre ella y su relación con el proceso software. Esta vista del marco de integración de usabilidad ofrece en estos casos al desarrollador una información organizada de un modo que le resulta conveniente.

En la Tabla 7.1 se muestran las técnicas IPO seleccionadas organizadas según esta vista, con la valoración de los criterios, la actividad en la que se aplican y la aplicabilidad en cada momento del desarrollo. Las técnicas con valoración total "muy útil" aparecen con fondo blanco, mientras que aquellas con valoración total "útil" aparecen con fondo gris.

7.2.1 Criterios de Caracterización

Cada columna de la Tabla 7.1 indica la valoración de cada técnica según unos criterios específicos que se detallan a continuación:

- **P.U. (Participación de los Usuarios):** Uno de los puntos básicos de un proceso centrado en el usuario, es la implicación activa de los futuros usuarios del sistema. Algunas de las técnicas de usabilidad están destinadas específicamente a favorecer dicha implicación, y aparecen en la tabla con valor **sí** en esta columna.
- **Necesidad de Formación:** Este criterio se refiere a cuánta formación necesita un ingeniero software medio para poder aplicar la técnica con mínimas garantías de éxito. Los valores posibles son los siguientes:
 - Un valor **muy alto** indica que se requiere una experiencia en usabilidad extensa para poder aplicarse, esto es, que se requiere personal experto en la materia para su aplicación.
 - Un valor **alto** indica que su aplicación requeriría al menos un perfil combinado IS-IPO, en el sentido de que el ingeniero software que pudiera aplicarla sería aquel que hubiera recibido una formación extensa en usabilidad.
 - Un valor **medio** indica que, si bien se requiere una formación de cierta importancia, puede ser aplicada por ingenieros software medios con dicha formación.
 - Un valor **bajo** indica que únicamente es necesaria formación básica para poder ser aplicada por ingenieros software medios.
- **Aplicabilidad (General):** Este criterio refleja la generalidad de la técnica, esto es, cuánto de aplicable es a un abanico amplio de tipos de proyectos de desarrollo de software. Los valores que puede tomar son los siguientes:
 - Un valor de aplicabilidad **alto** indica que su uso puede resultar de utilidad prácticamente en todo tipo de proyectos.
 - Si se tiene un valor **medio** para este criterio, quiere decir que la técnica es aplicable en ciertos tipos de proyectos, pero en otros tipos no.

Tabla 7.1 - Técnicas IPO Propuestas Ordenadas Alfabéticamente

Técnica	P.U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Tipo de Actividad	Momentos de Aplicación			Referencia Básica
									C. Iniciales	C. Centrales	C. Evolución	
Análisis Competitivo	no	medio	alto	medio	alto	1	útil	Educción y Análisis de Requisitos	esp. apropiada	neutra	neutra	[Nielsen, 93]
Análisis de Impacto	no	medio	medio	alto	medio	3	útil	Diseño	no es habitual	neutra	neutra	[Hix, 93]
Árboles de Menús	no	bajo	medio	alto	alto	1	muy útil	Diseño de la Interacción	neutra	neutra	neutra	[Shneiderman, 98]
Card Sorting	sí	bajo	alto	medio	alto	3	muy útil	Educción y Análisis de Requisitos	neutra	neutra	neutra	[Robertson, 01]
Casos de Uso Esenciales	no	medio	alto	alto	alto	1	muy útil	Educción y Análisis de Requisitos (Análisis de Tareas)	neutra	neutra	neutra	[Constantine, 99]
Cuestionarios, Entrevistas y Encuestas	sí	medio	alto	medio	medio	3	útil	Evaluación (Estudios de Seguimiento de Sistemas Instalados)	no es habitual	neutra	esp. apropiada	[Mayhew, 99]
Diagramas de Afinidad	sí	bajo	alto	medio	alto	1	útil	Educción y Análisis de Requisitos	esp. apropiada	no es habitual	no es habitual	[Beyer, 98]
Diagramas de Transición de Estados de la Interfaz	no	bajo	alto	alto	medio	2	útil	Diseño de la Interacción	neutra	neutra	neutra	[Wasserman, 85]
Escenarios de Tareas	sí	medio	medio	medio	alto	1	útil	Educción y Análisis de Requisitos (Análisis de Tareas)	neutra	neutra	neutra	[Mayhew, 99]
Escenarios y storyboards	sí	medio	medio	bajo	alto	3	muy útil	Educción y Análisis de Requisitos (Desarrollo del Concepto del Producto)	esp. apropiada	no es habitual	no es habitual	[Carroll, 97]
Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	muy útil	Especificación de Requisitos	neutra	neutra	neutra	[Hix, 93]
Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	neutra	neutra	neutra	[Nielsen, 93]
Guía de Estilo del Producto	no	alto	medio	medio	medio	1	útil	Diseño de la Interacción	no es habitual	esp. apropiada	neutra	[Mayhew, 99]
HTA	no	medio	medio	alto	medio	1	útil	Educción y Análisis de Requisitos (Análisis de Tareas)	esp. apropiada	no es habitual	no es habitual	[Annett, 04]
Información Post-Test	sí	medio	alto	medio	alto	1	útil	Evaluación (Test de Usabilidad)	no es habitual	neutra	neutra	[Constantine, 99]
Inspecciones	no	medio	alto	medio	alto	4	muy útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	neutra	neutra	neutra	[Nielsen, 94]
Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	esp. apropiada	neutra	neutra	[Constantine, 99]
Investigación Contextual	sí	alto	medio	medio	alto	3	útil	Educción y Análisis de Requisitos	esp. apropiada	no es habitual	no es habitual	[Beyer, 98]
JEM	sí	medio	medio	alto	medio	1	útil	Educción y Análisis de Requisitos	neutra	neutra	neutra	[Constantine, 99]
Mapa de Navegación	no	medio	alto	alto	medio	1	útil	Diseño de la Interacción	neutra	neutra	neutra	[Constantine, 99]

Técnica	P.U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Tipo de Actividad	Momentos de Aplicación			Referencia Básica
									C. Iniciales	C. Centrales	C. Evolución	
Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	útil	Educción y Análisis de Requisitos (Análisis de Usuarios)	esp. apropiada	neutra	neutra	[Constantine, 99]
Medición del Rendimiento	sí	medio	medio	medio	medio	3	útil	Evaluación (Test de Usabilidad)	no es habitual	neutra	neutra	[Dumas, 99]
Modelo del Contenido de la Interfaz	no	medio	alto	medio	medio	1	útil	Diseño de la Interacción	neutra	esp. apropiada	neutra	[Constantine, 99]
Observación Etnográfica	no	alto	medio	medio	medio	2	útil	Educción y Análisis de Requisitos	esp. apropiada	no es habitual	no es habitual	[Wixon, 96]
Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	útil	Diseño	neutra	neutra	neutra	[Constantine, 99]
Pensar en Voz Alta	sí	medio	alto	bajo	alto	5	muy útil	Evaluación (Test de Usabilidad)	no es habitual	neutra	neutra	[Nielsen, 93]
Perfiles de Usuario	no	alto	alto	alto	alto	5	útil	Educción y Análisis de Requisitos (Análisis de Usuarios)	esp. apropiada	neutra	neutra	[Mayhew, 99]
Personas	no	medio	medio	medio	alto	3	muy útil	Educción y Análisis de Requisitos (Análisis de Usuarios)	esp. apropiada	no es habitual	no es habitual	[Cooper, 03a]
Prototipos de Papel	sí	bajo	alto	alto	alto	3	muy útil	Educción y Análisis de Requisitos (Prototipado)	esp. apropiada	neutra	neutra	[Snyder, 03]
Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	neutra	neutra	neutra	[Lewis, 97]
Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	esp. apropiada	neutra	neutra	[Bias, 94]
Registro del Uso	no	alto	medio	alto	medio	5	útil	Evaluación (Estudios de Seguimiento de Sistemas Instalados)	no es habitual	neutra	esp. apropiada	[Shneiderman, 98]
Retroalimentación del Usuario	sí	bajo	alto	alto	alto	3	muy útil	Evaluación (Estudios de Seguimiento de Sistemas Instalados)	no es habitual	no es habitual	esp. apropiada	[Shneiderman, 98]
Test de Usabilidad en Laboratorio	sí	medio	medio	medio	medio	4	útil	Evaluación (Test de Usabilidad)	no es habitual	neutra	neutra	[Dumas, 99]
Tormenta de Ideas Visual	sí	bajo	alto	bajo	alto	1	útil	Educción y Análisis de Requisitos (Desarrollo del Concepto del Producto)	esp. apropiada	no es habitual	no es habitual	[Preece, 94]

- Un valor **bajo** en este criterio indica que únicamente ciertos tipos de proyectos, los cuales forman un porcentaje escaso entre todos los tipos de proyectos, son adecuados para su aplicación.
- **Cercanía a la IS:** Este criterio refleja si los principios en los que se basa la técnica coinciden con principios y enfoques habituales de la IS. Puede tomar los siguientes valores:
 - Un valor **alto** en este criterio indica que la técnica puede ser aplicada por los ingenieros software, puesto que se basa en habilidades y enfoques típicos de la IS.
 - Un valor **medio** indica que, si bien la técnica se basa en principios que no pertenecen a la IS, no está tan alejada de la misma como para considerarla ajena al campo.
 - Un valor **bajo** en este criterio indica que la técnica requiere un enfoque de desarrollo y unas habilidades que son ajenas a los que suele tener una persona con formación en IS.
- **Aportación/Esfuerzo (Grado de Aportación de Usabilidad Frente al Esfuerzo):** Este criterio se refiere a cuánta mejora en la usabilidad del producto final puede aportar el uso de la técnica, comparada con el esfuerzo que supone su aplicación. Puesto que los recursos para usabilidad pueden ser escasos en muchos proyectos de desarrollo de software, este criterio aporta información de tipo coste-beneficio. Los valores posibles para este criterio son **alto**, **medio** y **bajo**, reflejando la importancia de la aportación de usabilidad que puede resultar de la aplicación de la técnica, en relación al esfuerzo que tiene que invertir el equipo de desarrollo en su aplicación.
- **Rep. (Representatividad):** Este criterio refleja cuán comúnmente se aplica la técnica en el campo de la IPO. Como indicador de este criterio se ha utilizado el número de autores del estudio que aconsejan la aplicación de la técnica, por tanto el valor de este atributo es numérico y está en el rango **1-6**.
- **Val. Total (Valoración Total):** Este criterio recoge una valor agregado de los criterios anteriores, y refleja el grado de utilidad que la técnica puede aportar al objetivo de la integración de técnicas y actividades de usabilidad en el proceso de desarrollo, en el caso de una organización con un enfoque de IS. Puede tomar los valores **muy útil** y **útil**.
- **Tipo de Actividad:** Tipo(s) de actividad(es) del desarrollo en el/los que la aplicación de la técnica resulta de mayor utilidad para el objetivo de mejorar la usabilidad del producto software. En la sección 7.3.1 más adelante se detallan los posibles valores para este criterio.
- **Momentos de Aplicación:** Grado de adecuación a cada una de las etapas del desarrollo consideradas. Estas etapas se describen más adelante en la sección 7.4.1. Para cada etapa las técnicas pueden tomar los siguientes valores de idoneidad:
 - **Esp. apropiada (Especialmente Apropiada).** Significa que dicha técnica puede resultar de la máxima utilidad cuando se aplica en esta etapa del desarrollo.
 - **Neutra.** La técnica puede aplicarse en dicha etapa, sin destacar la etapa como la más apropiada frente a otros momentos del desarrollo.
 - **No es habitual.** Significa que no es la etapa más apropiada donde corresponde aplicar esta técnica y existen otras técnicas más apropiadas. Por tanto, no resulta habitual su aplicación en esta etapa.

- **Referencia Básica:** Para cada técnica se proporciona una referencia básica de la literatura IPO. Está destinada a servir de punto de partida para conocer los puntos básicos de la técnica y su modo de aplicación.

7.3 Vista por Actividades

Es habitual organizar los procesos de desarrollo de la IS por tipos de actividades. Esta organización de las técnicas IPO que forman parte del marco de integración de la usabilidad en el proceso de desarrollo permitirá a los desarrolladores ubicar con mayor facilidad el lugar de aplicación de las técnicas seleccionadas en su proceso de desarrollo.

7.3.1 Actividades Afectadas por la Usabilidad

En cualquier organización de desarrollo de software con un proceso de desarrollo definido se utiliza una terminología específica para denominar las actividades que forman parte del proceso. A pesar de que existe una gran diversidad entre distintas organizaciones, hay una serie de actividades básicas que aparecen en todos los procesos: Al nivel más alto éstas son análisis o ingeniería de requisitos, diseño y evaluación. Nos hemos basado en estos nombres de actividades comúnmente aceptados en la IS para organizar las técnicas IPO seleccionadas por tipos genéricos de actividad, de tal forma que la traducción a las actividades concretas de cada organización resulta una tarea más fácil.

En la Figura 7.1 se muestran los tipos de actividades del desarrollo afectadas por la usabilidad. Nótese que mientras algunas son actividades de cualquier proceso de desarrollo (con fondo rayado), otras son actividades que es preciso añadir al proceso de desarrollo si se desea incorporar la usabilidad al mismo (fondo blanco). Las primeras son actividades que suelen estar presentes en cualquier proceso, pero cuya modificación para que incluyan el uso de técnicas IPO es necesaria si se desea tratar adecuadamente la usabilidad del producto final. Las segundas son incorporaciones al proceso que habrá que realizar para conseguir el objetivo de la integración de la usabilidad. Para el primer caso, los términos utilizados corresponden a nombres genéricos de actividad, los cuales el desarrollador usuario del marco de integración debe relacionar con la terminología concreta de actividades utilizada en su organización.

A continuación se define cada uno de los tipos de actividades que aparecen en la Figura 7.1 para una mejor comprensión del esquema por actividades:

- **Análisis (Ingeniería de Requisitos)**
 - **Educción y Análisis de Requisitos:** En la Educción de Requisitos se identifican las fuentes de requisitos y se capturan los mismos. El Análisis de Requisitos se ocupa de detectar y resolver posibles conflictos entre requisitos, definir los límites del sistema y cómo debe interactuar con su entorno, y elaborar el paso de requisitos del sistema a requisitos software. Hemos considerado un solo tipo de actividad que agrupa a ambas porque no hay una delimitación clara entre ellas en las técnicas IPO. Dentro de este tipo de actividades, podemos agrupar algunas de las técnicas IPO en subactividades con objetivos específicos, según la terminología IPO:
 - **Análisis de Usuarios:** Se ocupa de identificar cuáles son los usuarios previstos del sistema, y para ellos recoger toda información sobre sus conocimientos, necesidades y características que sea relevante en su interacción con el sistema.

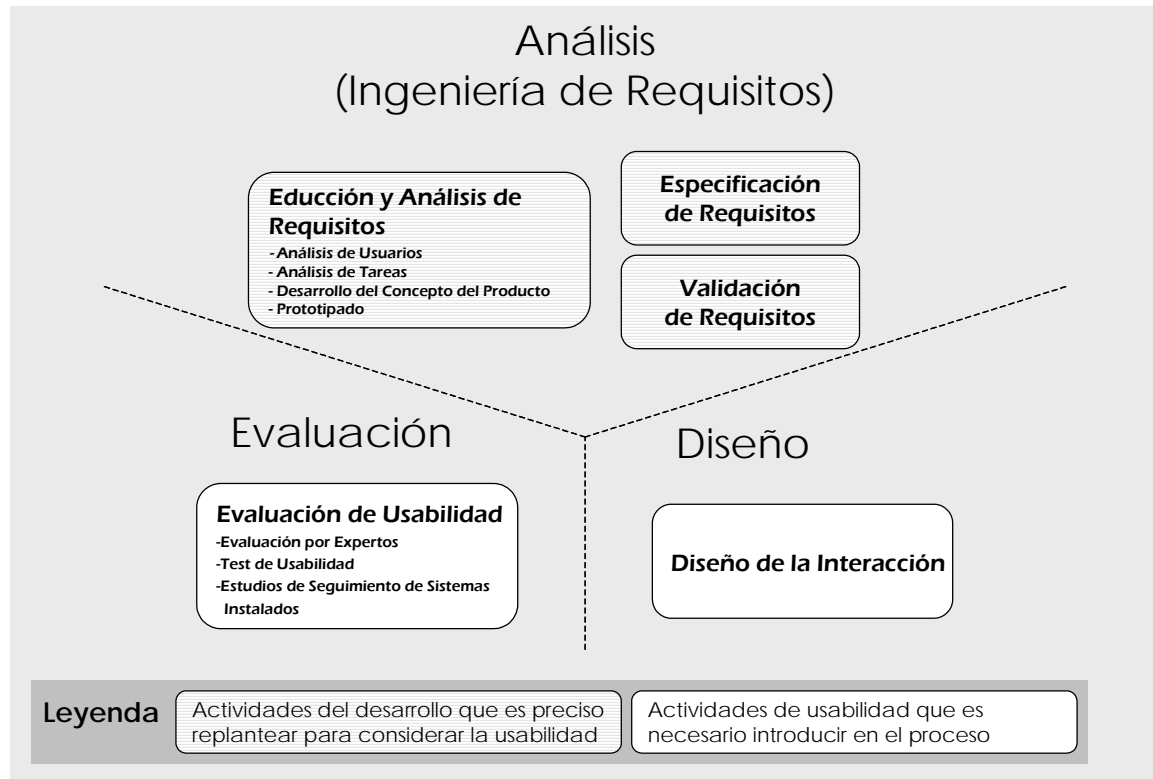


Figura 7.1 - Actividades del Desarrollo Afectadas por la Usabilidad

- **Análisis de Tareas:** Tiene como finalidad obtener descripciones de lo que las personas hacen para llevar a cabo los asuntos de los que se ocupan. Centra las actividades de Educación y Análisis de Requisitos en los objetivos últimos del uso del sistema por parte del usuario.
 - **Desarrollo del Concepto del Producto:** En esta actividad se considera la lógica que seguirá el funcionamiento del sistema, sus espacios de interacción principales y cómo se trabajará con el mismo. Definir un concepto del producto explícito ayuda a que éste se transmita de forma adecuada al usuario, de tal forma que éste pueda desarrollar un correcto modelo mental del sistema.
 - **Prototipado:** El objetivo de esta actividad es lograr una mejor comprensión de las necesidades y deseos del usuario mediante el uso de prototipos que permitan avanzar hacia una comprensión común del problema que se está abordando.
- **Especificación de Requisitos:** La Especificación de Requisitos tiene que ver con la elaboración de un documento donde se reflejan los requisitos que el sistema debe cumplir, y en particular tiene que ver con la estructura, calidad y verificabilidad de dicho documento.
 - **Validación de Requisitos:** Conjunto de actividades que tienen como finalidad examinar los requisitos para asegurarse de que definen el sistema adecuado (el sistema que el cliente/usuario espera).

- **Diseño**
 - **Diseño de la Interacción:** El Diseño de la Interacción se encarga de la definición de los entornos de interacción y su comportamiento. Al incluir el comportamiento implica coordinar la interacción entre el usuario y el sistema. También incluye el diseño de los elementos visuales que forman la interfaz gráfica de usuario, cuando la interfaz es de dicho estilo. Se trata de una actividad relativamente independiente del resto de actividades de diseño, por lo que tiene un espacio propio dentro de las actividades de este tipo.

- **Evaluación**
 - **Evaluación de Usabilidad:** Incluye todas las actividades relacionadas con la evaluación de la usabilidad del producto software en construcción. Este tipo de actividades tiene un alto grado de independencia del resto de actividades de evaluación, y por eso se agrupan en esta nueva actividad todas las labores de evaluación de la usabilidad. Estas técnicas se agrupan en tres grandes tipos:
 - **Evaluación por Expertos:** Este tipo de evaluación no requiere la participación de usuarios representativos utilizando el sistema, y complementa al resto de actividades de evaluación de la usabilidad que se planteen en un proyecto.
 - **Test de Usabilidad:** Incluyen todas las variantes de test de usabilidad con usuarios representativos. Se trata de la evaluación de usabilidad más característica, necesaria en cualquier proyecto de desarrollo software que tenga como objetivo un buen nivel de usabilidad, porque permite probar el software que se va desarrollando en condiciones similares a las de su uso previsto.
 - **Estudios de Seguimiento de Sistemas Instalados:** Este tipo de evaluación se aplica sobre sistemas o prototipos software ya instalados en el entorno previsto de uso. Permite recoger datos fidedignos sobre el uso real del software y las dificultades que plantea.

7.3.2 Tabla Resumen por Tipo de Actividad

Una vez definidos los tipos de actividad en los que se basa esta vista del marco de integración, las técnicas IPO propuestas se muestran en la Tabla 7.2 organizadas por el tipo de actividad en la que su aplicación resulta de mayor utilidad. Al igual que en la Tabla 7.1, se muestran para cada técnica la valoración de los criterios y la aplicabilidad en cada momento del desarrollo, y las técnicas con valoración total "muy útil" aparecen con fondo blanco, mientras que las técnicas con valoración total "útil" aparecen con fondo gris.

Para utilizar esta vista del marco de integración el desarrollador identifica las actividades en las que debe introducir técnicas IPO mediante la descripción de la sección 7.3.1, y para cada actividad consulta la Tabla 7.2 para decidir cuáles de las técnicas aplicables en la actividad escoge para integrar en su proceso. Para dicha elección, además de la diferenciación entre técnicas "muy útiles" y "útiles", el desarrollador cuenta con la caracterización de cada técnica según los criterios establecidos en el capítulo 4, los cuales interpretará según las características particulares del proyecto en cuestión o de la organización que va a construir el software.

Tabla 7.2 - Técnicas IPO Propuestas Organizadas por Actividad

Tipo de Actividad	Técnica	P. U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Momentos de Aplicación			Referencia Básica	
									C. Iniciales	C. Centrales	C. Evolución		
Análisis (Ing. de Requisitos)	Educción y Análisis de Requisitos	<i>Card Sorting</i>	sí	bajo	alto	medio	alto	3	muy útil	neutra	neutra	neutra	[Robertson, 01]
		Análisis Competitivo	no	medio	alto	medio	alto	1	útil	esp. apropiada	neutra	neutra	[Nielsen, 93]
		Diagramas de Afinidad	sí	bajo	alto	medio	alto	1	útil	esp. apropiada	no es habitual	no es habitual	[Beyer, 98]
		Investigación Contextual	sí	alto	medio	medio	alto	3	útil	esp. apropiada	no es habitual	no es habitual	[Beyer, 98]
		JEM	sí	medio	medio	alto	medio	1	útil	neutra	neutra	neutra	[Constantine, 99]
		Observación Etnográfica	no	alto	medio	medio	medio	2	útil	esp. apropiada	no es habitual	no es habitual	[Wixon, 96]
	Análisis de Usuarios *	Personas	no	medio	medio	medio	alto	3	muy útil	esp. apropiada	no es habitual	no es habitual	[Cooper, 03a]
		Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	útil	esp. apropiada	neutra	neutra	[Constantine, 99]
		Perfiles de Usuario	no	alto	alto	alto	alto	5	útil	esp. apropiada	neutra	neutra	[Mayhew, 99]
		Casos de Uso Esenciales	no	medio	alto	alto	alto	1	muy útil	neutra	neutra	neutra	[Constantine, 99]
	Análisis de Tareas *	Escenarios de Tareas	sí	medio	medio	medio	alto	1	útil	neutra	neutra	neutra	[Mayhew, 99]
		HTA	no	medio	medio	alto	medio	1	útil	esp. apropiada	no es habitual	no es habitual	[Annett, 04]
		Escenarios y storyboards	sí	medio	medio	bajo	alto	3	muy útil	esp. apropiada	no es habitual	no es habitual	[Carroll, 97]
	Desarrollo del Concepto del Producto *	Tormenta de Ideas Visual	sí	bajo	alto	bajo	alto	1	útil	esp. apropiada	no es habitual	no es habitual	[Preece, 94]
	Prototipado *	Prototipos de Papel	sí	bajo	alto	alto	alto	3	muy útil	esp. apropiada	neutra	neutra	[Snyder, 03]
	Especificación de Requisitos	Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	muy útil	neutra	neutra	neutra	[Hix, 93]
	Validación de Requisitos	Inspecciones	no	medio	alto	medio	alto	4	muy útil	neutra	neutra	neutra	[Nielsen, 94]
		Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	neutra	neutra	neutra	[Nielsen, 93]
		Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil	esp. apropiada	neutra	neutra	[Constantine, 99]
		Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil	neutra	neutra	neutra	[Lewis, 97]
Recorrido Pluralístico		sí	bajo	medio	medio	medio	4	útil	esp. apropiada	neutra	neutra	[Bias, 94]	

* No se trata de tipos de actividades IS, sino actividades IPO incluidas para ofrecer al desarrollador una visión estructurada de las 15 técnicas IPO que pueden aplicarse en actividades de Educción y Análisis de Requisitos.

Tipo de Actividad		Técnica	P. U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Momentos de Aplicación			Referencia Básica
										C. Iniciales	C. Centrales	C. Evolución	
Diseño	Diseño de la Interacción	Árboles de Menús	no	bajo	medio	alto	alto	1	muy útil	neutra	neutra	neutra	[Shneiderman, 98]
		Diagramas de Transición de Estados de la Interfaz	no	bajo	alto	alto	medio	2	útil	neutra	neutra	neutra	[Wasserman, 85]
		Guía de Estilo del Producto	no	alto	medio	medio	medio	1	útil	no es habitual	esp. apropiada	neutra	[Mayhew, 99]
		Mapa de Navegación	no	medio	alto	alto	medio	1	útil	neutra	neutra	neutra	[Constantine, 99]
		Modelo del Contenido de la Interfaz	no	medio	alto	medio	medio	1	útil	neutra	esp. apropiada	neutra	[Constantine, 99]
		Análisis de Impacto	no	medio	medio	alto	medio	3	útil	no es habitual	neutra	neutra	[Hix, 93]
		Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	útil	neutra	neutra	neutra	[Constantine, 99]
Evaluación	Evaluación por Expertos	Inspecciones	no	medio	alto	medio	alto	4	muy útil	neutra	neutra	neutra	[Nielsen, 94]
		Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	neutra	neutra	neutra	[Nielsen, 93]
		Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil	esp. apropiada	neutra	neutra	[Constantine, 99]
		Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil	neutra	neutra	neutra	[Lewis, 97]
		Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	útil	esp. apropiada	neutra	neutra	[Bias, 94]
	Test de Usabilidad	Pensar en Voz Alta	sí	medio	alto	bajo	alto	5	muy útil	no es habitual	neutra	neutra	[Nielsen, 93]
		Información Post-Test	sí	medio	alto	medio	alto	1	útil	no es habitual	neutra	neutra	[Constantine, 99]
		Medición del Rendimiento	sí	medio	medio	medio	medio	3	útil	no es habitual	neutra	neutra	[Dumas, 99]
		Test de Usabilidad en Laboratorio	sí	medio	medio	medio	medio	4	útil	no es habitual	neutra	neutra	[Dumas, 99]
	Estudios de Seguimiento de Sistemas Instalados	Retroalimentación del Usuario	sí	bajo	alto	alto	alto	3	muy útil	no es habitual	no es habitual	esp. apropiada	[Shneiderman, 98]
		Cuestionarios, Entrevistas y Encuestas	sí	medio	alto	medio	medio	3	útil	no es habitual	neutra	esp. apropiada	[Mayhew, 99]
Registro del Uso		no	alto	medio	alto	medio	5	útil	no es habitual	neutra	esp. apropiada	[Shneiderman, 98]	

7.4 Vista por Momentos del Desarrollo

La dimensión más habitual según la que se planifica un proyecto es según el eje tiempo. Los procesos de desarrollo con enfoque iterativo presentan una mayor dificultad para tratar dicha variable, pero aún así, existen en todo proceso iterativo unos momentos diferenciados. La presentación de los momentos de aplicación más adecuados para cada técnica puede, así, resultar de gran ayuda al desarrollador que, en un momento dado, necesita enfocar la selección de técnicas según su mejor momento de aplicación.

7.4.1 Etapas en un desarrollo iterativo

Según se muestra en la Figura 7.2, los ciclos que se llevan a cabo en cualquier desarrollo iterativo pueden agruparse en tres tipos de etapas consecutivas:

- **Ciclos iniciales:** En cualquier proceso iterativo los ciclos iniciales se dedican a los puntos identificados como de mayor riesgo. Suele trabajarse en torno a la elaboración de un concepto del producto adecuado. Por tanto, cuanto más formado esté el concepto del producto al plantear el proyecto, tanto más corta será esta etapa. Ciertas técnicas se centran en la producción y/o evaluación de las tentativas iniciales de diseño que se elaboran, y por tanto son especialmente apropiadas para los ciclos iniciales. El hito que define el fin de esta etapa se denomina **Concepto del Producto Establecido**.
- **Ciclos centrales:** Se llevan a cabo una vez las líneas generales del producto han sido establecidas y se van desarrollando más a fondo aquellas partes del sistema cuya definición haya quedado a un nivel más abstracto. Esta etapa comienza una vez acaban los ciclos iniciales y continúa hasta que empiezan los ciclos de evolución.
- **Ciclos de evolución:** Estos ciclos comienzan cuando se dispone de una versión del sistema software que se está desarrollando instalada en el entorno de trabajo del usuario final. Así, en estos ciclos se pueden aplicar las técnicas que requieren que se tenga una parte del software instalada en el entorno previsto de uso. El hito que define el inicio de esta etapa se denomina **Parte del Sistema Instalada y en Funcionamiento**.

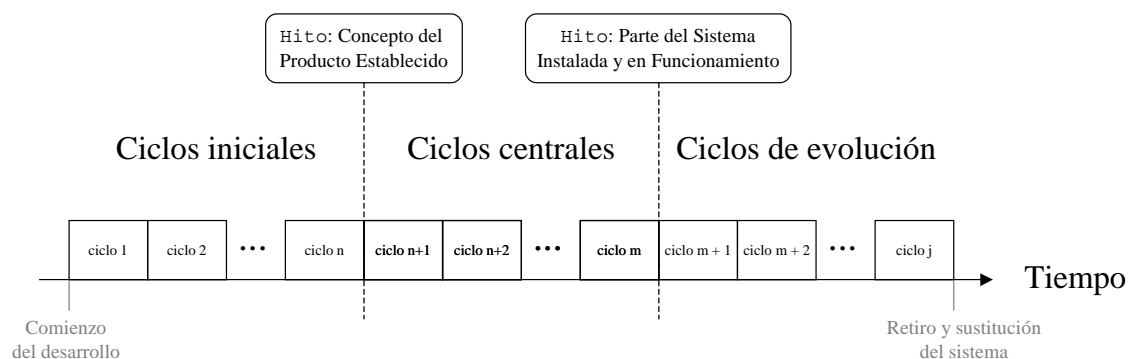


Figura 7.2 - División de los Ciclos Iterativos en Etapas

Estas etapas de cualquier desarrollo iterativo deben ser interpretadas por el usuario del marco de integración, con el fin de establecer su correspondencia con las fases del proceso de

desarrollo llevado a cabo en la organización, para así poder utilizar la información de esta vista adecuadamente.

7.4.2 Tabla Resumen por Momento de Aplicación

En la Tabla 7.3 se muestran las técnicas de usabilidad propuestas organizadas por momento de aplicación, distinguiendo en cada uno qué técnicas son especialmente apropiadas y cuáles son neutras en cuanto a tal momento de aplicación. Dentro de cada grupo, así mismo, se distinguen las técnicas con valoración total "muy útil", con fondo blanco, de las que tienen valoración total "útil", con fondo gris. En cada tipo las técnicas están organizadas según las actividades del desarrollo en las que se aplican. Por otra parte, al igual que en las tablas anteriores, se incluyen la caracterización de cada técnica según los criterios establecidos anteriormente.

Para utilizar esta vista, el usuario del marco de integración acude a la etapa del desarrollo en la que tiene interés, y en ella tiene destacadas cuáles son las técnicas más apropiadas para dicha etapa. Así, tiene un criterio más de decisión para la elección de técnicas IPO a integrar en su proceso de desarrollo.

7.5 Uso del Marco

En un desarrollo iterativo no se suele realizar una planificación predictiva en la que todo el desarrollo se prevé de forma detallada desde el principio, sino que, según se avanza en el proyecto, se va decidiendo cómo afrontar el siguiente ciclo de desarrollo (y el ciclo siguiente a éste se perfila) [Larman, 04]. En un escenario como éste, la selección de las técnicas a aplicar durante el desarrollo también se va realizando según éste progresa, y el uso del marco de integración propuesto se circunscribe a esfuerzos de desarrollo de este tipo.

Así, en los primeros momentos del desarrollo, el ingeniero software puede acudir al marco para elegir la(s) técnica(s) IPO a incluir en una determinada actividad. Acudirá a la vista por actividades, relacionará el término para dicha actividad utilizado en su empresa con los nombres genéricos de tipos de actividades expresados en el marco, y una vez identificado el tipo de actividad que quiere estudiar, elegirá de entre las técnicas ofrecidas en el marco para dicha actividad aquella o aquellas que mejor se acomoden a los ciclos iniciales del desarrollo. Además, tendrá en cuenta, según considere oportuno, los valores para las técnicas en el resto de criterios de caracterización. Así, por ejemplo, escogerá en primer lugar aquellas técnicas etiquetadas con una necesidad de formación menor, en el caso de contar con un equipo que necesita poner a producir software de forma rápida.

El mismo ingeniero software, una vez se cuenta con una versión preliminar del sistema instalada y funcionando en el entorno previsto de uso, puede querer comprobar qué técnicas IPO pueden mejor aportar a la usabilidad del producto final en dicho momento del desarrollo. Entonces acudirán a la vista por momentos, a la parte correspondiente a los ciclos finales. En ella tendrá para poder escoger las técnicas consideradas como especialmente apropiadas para esta etapa. Si, por ejemplo, desea formar al equipo en técnicas cuyo proceso de aprendizaje pueda amortizarse en un abanico amplio de futuros proyectos, escogerá antes aquellas técnicas con un valor alto de aplicabilidad.

Tabla 7.3 - Técnicas IPO Propuestas Organizadas según el Momento Más Adecuado de Aplicación en un Desarrollo Iterativo

Etapa en un Ciclo Iterativo		Técnica	P. U	Nece- sidad de For- mación	Aplicabi- lidad	Cer- canía a la IS	Apor- tación / Es- fuerzo	Rep.	Val. Total	Tipo de Actividad	Referencia Básica
Ciclos Iniciales	Especial- mente apropiadas	Personas	no	medio	medio	medio	alto	3	muy útil	Educción y Análisis de Req. (Análisis de Usuarios)	[Cooper, 03a]
		Escenarios y <i>storyboards</i>	sí	medio	medio	bajo	alto	3	muy útil	Educción y Análisis de Req. (Desarrollo del Concepto del Producto)	[Carroll, 97]
		Prototipos de Papel	sí	bajo	alto	alto	alto	3	muy útil	Educción y Análisis de Req. (Prototipado)	[Snyder, 03]
		Análisis Competitivo	no	medio	alto	medio	alto	1	útil	Educción y Análisis de Req.	[Nielsen, 93]
		Diagramas de Afinidad	sí	bajo	alto	medio	alto	1	útil		[Beyer, 98]
		Investigación Contextual	sí	alto	medio	medio	alto	3	útil		[Beyer, 98]
		Observación Etnográfica	no	alto	medio	medio	medio	2	útil		[Wixon, 96]
		Perfiles de Usuario	no	alto	alto	alto	alto	5	útil		[Mayhew, 99]
		Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	útil		[Constantine, 99]
		HTA	no	medio	medio	alto	medio	1	útil	[Annett, 04]	
		Tormenta de Ideas Visual	sí	bajo	alto	bajo	alto	1	útil	Educción y Análisis de Req. (Desarrollo del Concepto del Producto)	[Preece, 94]
		Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Constantine, 99]
		Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	útil		[Bias, 94]
	Neutras	<i>Card Sorting</i>	sí	bajo	alto	medio	alto	3	muy útil	Educción y Análisis de Req.	[Robertson, 01]
		Casos de Uso Esenciales	no	medio	alto	alto	alto	1	muy útil	Educción y Análisis de Req. (Análisis de Tareas)	[Constantine, 99]
		Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	muy útil	Especificación de Requisitos	[Hix, 93]
		Inspecciones	no	medio	alto	medio	alto	4	muy útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 94]
		Árboles de Menús	no	bajo	medio	alto	alto	1	muy útil	Diseño de la Interacción	[Shneiderman, 98]
		JEM	sí	medio	medio	alto	medio	1	útil	Educción y Análisis de Req.	[Constantine, 99]
		Escenarios de Tareas	sí	medio	medio	medio	alto	1	útil	Educción y Análisis de Req. (Análisis de Tareas)	[Mayhew, 99]
		Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 93]
		Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil		[Lewis, 97]
Diagramas de Transición de Estados de la Interfaz		no	bajo	alto	alto	medio	2	útil	Diseño de la Interacción	[Wasserman, 85]	
Mapa de Navegación	no	medio	alto	alto	medio	1	útil	[Constantine, 99]			
Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	útil	Diseño	[Constantine, 99]		

Etapa en un Ciclo Iterativo		Técnica	P. U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Tipo de Actividad	Referencia Básica
Ciclos Centrales	Especialmente apropiadas	Guía de Estilo del Producto	no	alto	medio	medio	medio	1	útil	Diseño de la Interacción	[Mayhew, 99]
		Modelo del Contenido de la Interfaz	no	medio	alto	medio	medio	1	útil		[Constantine, 99]
	Neutras	<i>Card Sorting</i>	sí	bajo	alto	medio	alto	3	muy útil	Educación y Análisis de Req.	[Robertson, 01]
		Casos de Uso Esenciales	no	medio	alto	alto	alto	1	muy útil	Educación y Análisis de Req. (Análisis de Tareas)	[Constantine, 99]
		Prototipos de Papel	sí	bajo	alto	alto	alto	3	muy útil	Educación y Análisis de Req. (Prototipado)	[Snyder, 03]
		Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	muy útil	Especificación de Requisitos	[Hix, 93]
		Inspecciones	no	medio	alto	medio	alto	4	muy útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 94]
		Árboles de Menús	no	bajo	medio	alto	alto	1	muy útil	Diseño de la Interacción	[Shneiderman, 98]
		Pensar en Voz Alta	sí	medio	alto	bajo	alto	5	muy útil	Evaluación (Test de Usabilidad)	[Nielsen, 93]
		Análisis Competitivo	no	medio	alto	medio	alto	1	útil	Educación y Análisis de Req.	[Nielsen, 93]
		JEM	sí	medio	medio	alto	medio	1	útil		[Constantine, 99]
		Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	útil	Educación y Análisis de Req. (Análisis de Usuarios)	[Constantine, 99]
		Perfiles de Usuario	no	alto	alto	alto	alto	5	útil		[Mayhew, 99]
		Escenarios de Tareas	sí	medio	medio	medio	alto	1	útil	Educación y Análisis de Req. (Análisis de Tareas)	[Mayhew, 99]
		Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 93]
		Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil		[Constantine, 99]
		Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil		[Lewis, 97]
		Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	útil		[Bias, 94]
		Diagramas de Transición de Estados de la Interfaz	no	bajo	alto	alto	medio	2	útil	Diseño de la Interacción	[Wasserman, 85]
		Mapa de Navegación	no	medio	alto	alto	medio	1	útil		[Constantine, 99]
		Análisis de Impacto	no	medio	medio	alto	medio	3	útil	Diseño	[Hix, 93]
		Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	útil		[Constantine, 99]
		Medición del Rendimiento	sí	medio	medio	medio	medio	3	útil	Evaluación (Test de Usabilidad)	[Dumas, 99]
		Información Post-Test	sí	medio	alto	medio	alto	1	útil		[Constantine, 99]
		Test de Usabilidad en Laboratorio	sí	medio	medio	medio	medio	4	útil		[Dumas, 99]
		Cuestionarios, Entrevistas y Encuestas	sí	medio	alto	medio	medio	3	útil	Evaluación (Estudios de Seguimiento de Sistemas Instalados)	[Mayhew, 99]
		Registro del Uso	no	alto	medio	alto	medio	5	útil		[Shneiderman, 98]
		Ciclos de Evolución	Especialmente apropiadas	Retroalimentación del Usuario	sí	bajo	alto	alto	alto	3	muy útil
Cuestionarios, Entrevistas y Encuestas	sí			medio	alto	medio	medio	3	útil	Evaluación (Estudios de Seguimiento de Sistemas Instalados)	[Mayhew, 99]
Registro del Uso	no			alto	medio	alto	medio	5	útil		[Shneiderman, 98]

Etapa en un Ciclo Iterativo	Técnica	P. U	Necesidad de Formación	Aplicabilidad	Cercanía a la IS	Aportación / Esfuerzo	Rep.	Val. Total	Tipo de Actividad	Referencia Básica
Neutras	Card Sorting	sí	bajo	alto	medio	alto	3	muy útil	Educción y Análisis de Req.	[Robertson, 01]
	Casos de Uso Esenciales	no	medio	alto	alto	alto	1	muy útil	Educción y Análisis de Req. (Análisis de Tareas)	[Constantine, 99]
	Prototipos de Papel	sí	bajo	alto	alto	alto	3	muy útil	Educción y Análisis de Req. (Prototipado)	[Snyder, 03]
	Especificaciones de Usabilidad	no	medio	medio	medio	alto	4	muy útil	Especificación de Requisitos	[Hix, 93]
	Inspecciones	no	medio	alto	medio	alto	4	muy útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 94]
	Árboles de Menús	no	bajo	medio	alto	alto	1	muy útil	Diseño de la Interacción	[Shneiderman, 98]
	Pensar en Voz Alta	sí	medio	alto	bajo	alto	5	muy útil	Evaluación (Test de Usabilidad)	[Nielsen, 93]
	Análisis Competitivo	no	medio	alto	medio	alto	1	útil	Educción y Análisis de Req.	[Nielsen, 93]
	JEM	sí	medio	medio	alto	medio	1	útil		[Constantine, 99]
	Mapa de Roles de Usuario	no	bajo	medio	alto	medio	1	útil	Educción y Análisis de Req. (Análisis de Usuarios)	[Constantine, 99]
	Perfiles de Usuario	no	alto	alto	alto	alto	5	útil		[Mayhew, 99]
	Escenarios de Tareas	sí	medio	medio	medio	alto	1	útil	Educción y Análisis de Req. (Análisis de Tareas)	[Mayhew, 99]
	Evaluación Heurística	no	alto	alto	bajo	alto	6	útil	Validación de Requisitos ó Evaluación (Evaluación por Expertos)	[Nielsen, 93]
	Inspecciones Colaborativas	sí	medio	medio	medio	medio	1	útil		[Constantine, 99]
	Recorrido Cognitivo	no	alto	medio	medio	medio	4	útil		[Lewis, 97]
	Recorrido Pluralístico	sí	bajo	medio	medio	medio	4	útil		[Bias, 94]
	Diagramas de Transición de Estados de la Interfaz	no	bajo	alto	alto	medio	2	útil	Diseño de la Interacción	[Wasserman, 85]
	Guía de Estilo del Producto	no	alto	medio	medio	medio	1	útil		[Mayhew, 99]
	Modelo del Contenido de la Interfaz	no	medio	alto	medio	medio	1	útil		[Constantine, 99]
	Mapa de Navegación	no	medio	alto	alto	medio	1	útil		[Constantine, 99]
	Análisis de Impacto	no	medio	medio	alto	medio	3	útil	Diseño	[Hix, 93]
	Organización de la Ayuda según Casos de Uso	no	medio	medio	alto	medio	1	útil		[Constantine, 99]
	Medición del Rendimiento	sí	medio	medio	medio	medio	3	útil	Evaluación (Test de Usabilidad)	[Dumas, 99]
	Información Post-Test	sí	medio	alto	medio	alto	1	útil		[Constantine, 99]
Test de Usabilidad en Laboratorio	sí	medio	medio	medio	medio	4	útil	[Dumas, 99]		

Otra posibilidad de uso del sistema sería el caso de una compañía que construye software, que pretende abordar la integración de unas cuantas técnicas de usabilidad en su proceso para uno de los departamentos con que cuenta. El jefe del departamento, entonces, establecería una asignación preliminar de técnicas IPO a actividades del desarrollo, tras comprobar cómo se corresponde la terminología general del marco con la propia de la empresa, y acceder alternativamente entre la vista por actividades y la vista por momentos. Una vez establecida tal asignación preliminar, puede estar interesado en conocer algún dato concreto de alguna técnica IPO, como, por ejemplo, las necesidades de formación. El jefe del departamento puede querer planificar previamente la formación que su equipo tendrá que recibir para poder

aplicar una técnica IPO concreta, en cuanto a horas empleadas. En este caso, acudirá a la vista por técnicas, ir buscando las técnicas IPO seleccionadas, y en la información asociada podrá comprobar la necesidad de formación para cada una de ellas. Así, podrá actuar en consecuencia, reservando más o menos tiempo para la formación de los miembros de su equipo.

De esta forma, las vistas del marco de integración propuesto están pensadas como vías complementarias de acceso a la información relevante para la integración de la usabilidad en el proceso. En el uso del marco, las vistas pueden combinarse de forma flexible, de forma que se dé mejor soporte a la difícil tarea de selección de técnicas IPO a lo largo del desarrollo, cuando se sigue un enfoque iterativo.

7.6 Herramienta Web para el Uso del Marco de Integración

En la dirección <http://is.ls.fi.upm.es/udis/miembros/xavier/usabilityframework/> está disponible una herramienta que permite seleccionar las técnicas IPO que se deseen integrar en el proceso de desarrollo según las distintas vistas de las que se compone el marco. Permite tanto una selección gráfica como textual del conjunto de técnicas a cuya información se quiere acceder en un determinado momento. En la Figura 7.3 se muestra la página principal de la herramienta.

The screenshot shows the main page of the web tool. At the top, there is a navigation bar with the text "Inicio → Marco de Integración de la Usabilidad en el Proceso de Desarrollo Software" and a search box with a "Buscar" button. Below this is the main heading "Marco de Integración de la Usabilidad en el Proceso de Desarrollo Software". The main content area contains several paragraphs of text explaining the tool's purpose and providing search options. At the bottom, there is a footer with links for "Foro acerca del contenido de estas páginas", "Evalúe la usabilidad de este sitio", "Estadísticas de la evaluación de usabilidad", and contact information for "Xavier Ferré" at the "Facultad de Informática" of the "Universidad Politécnica de Madrid".

[Inicio](#) → Marco de Integración de la Usabilidad en el Proceso de Desarrollo Software

Marco de Integración de la Usabilidad en el Proceso de Desarrollo Software

Este sitio web tiene como objetivo servir a la organización de desarrollo de software que quiere integrar la usabilidad en su proceso de desarrollo con el objetivo de construir productos software usables. El único requisito que debe cumplir el proceso para poder integrar técnicas de usabilidad con dicho objetivo es estar basado en un enfoque iterativo, pues tal característica es clave para desarrollar productos con un buen nivel de usabilidad.

En la selección de las técnicas de usabilidad más adecuadas a su proceso software y organización, el marco de integración de la usabilidad en el proceso puede accerse a través de distintas vistas:

Búsqueda por tipo de actividad: Podrá buscar las técnicas de usabilidad que mejor se ajusten a las actividades que desea usar en su ciclo de vida. Puede hacerlo si lo desea [gráficamente](#). Además podrá ver [todas las relaciones](#) con las técnicas de usabilidad.

Búsqueda por técnica: Acceso a la caracterización de las técnicas por nombre de cada técnica de usabilidad.

Búsqueda por valoración: Cada técnica ha sido asignada una valoración según su adecuación para la integración en el proceso de desarrollo de una organización que sigue mayoritariamente el enfoque de desarrollo y terminología de la ingeniería del software. Esta vista ofrece la posibilidad de acceder a las técnicas por valoración.

Búsqueda por momentos de aplicación: Seleccione las técnicas en función de los momentos de desarrollo más adecuados para su aplicación.

Búsqueda por referencia bibliográfica básica: Acceso a las técnicas por autor que las describe como referencia básica.

[Foro acerca del contenido de estas páginas](#) [Evalúe la usabilidad de este sitio](#) [Estadísticas de la evaluación de usabilidad](#)

Marco de Integración de la Usabilidad en el Proceso de Desarrollo Software [Xavier Ferré](#) [Acerca de este sitio](#) [Facultad de Informática](#) [Universidad Politécnica de Madrid](#)

Figura 7.3 - Página Principal de la Herramienta Web

Ofrece, así mismo, una vista adicional a las presentadas en este capítulo. Aprovechando las posibilidad que el uso de un sistema de base de datos ofrece, el sistema permite acceder al conjunto de técnicas IPO tomadas de una referencia bibliográfica concreta. Esta vista adicional puede resultar de interés para expertos IPO que deseen valorar la selección de técnicas realizada, tal vez para aconsejar a su organización la utilización del marco de integración.

Capítulo 8

Aplicación y

Evaluación de

Resultados

8.1 Introducción

Las propuestas metodológicas cuentan con una especial dificultad para evaluar su viabilidad, puesto que requieren que un equipo de desarrollo modifique sus prácticas de construcción de software para acomodarse a la propuesta, y que tal aplicación se realice en condiciones lo más cercanas posibles a un entorno real de desarrollo. Además, en aras de la evaluación, es necesario que el equipo se deje observar de cerca de modo que puedan comprobarse las prácticas de desarrollo que se están empleando y cómo se están empleando. Por otra parte, de cara a la evaluación de la posible mejora obtenida en el producto resultante, es necesario contar con proyectos anteriores de similares características que permitan la comparación de los resultados obtenidos al aplicar la propuesta con los resultados de dichos proyectos. El objetivo de aislar los factores de contexto que pueden afectar a la comparación no resulta siempre factible, por lo que la validación experimental de las propuestas de tipo metodológico en muchos casos no puede realizarse de manera satisfactoria.

Tomando en cuenta las consideraciones previas, este capítulo detalla el estudio realizado sobre la viabilidad de aplicación del marco de integración propuesto en el presente trabajo de investigación, y la evaluación de los resultados que ha producido su aplicación. En primer lugar, se detalla en la sección 8.2 cómo se ha comprobado la viabilidad de la propuesta mediante su aplicación por parte de la empresa LogicDIS. En la sección 8.3 se refieren los casos de estudio que se centran en la aplicación de la propuesta por parte de estudiantes de la Facultad de Informática de la Universidad Politécnica de Madrid. Finalmente, en la sección 8.4 se discuten los resultados obtenidos.

8.2 Estudio de Viabilidad

La propuesta recogida en el presente trabajo de investigación se ha elaborado en el marco del proyecto STATUS, financiado por la Comisión Europea en su V programa marco bajo el contrato IST-2001-32298. En concreto, se enmarca en el WP5 (Paquete de Trabajo 5) del proyecto, centrado en la obtención de un proceso de desarrollo con integración de técnicas de

usabilidad. Este WP5 ha sido liderado por la UPM, cuyo equipo en el proyecto ha estado formado por las directoras de tesis y el autor del presente trabajo de investigación.

Tal y como se ha indicado en el capítulo 1, la propuesta se ha ido refinando siguiendo las indicaciones y retroalimentación obtenidos de los socios industriales del proyecto. La primera aproximación a la integración de las prácticas de usabilidad en el proceso se plasmó en los entregables D.5.1 [Ferré, 02a] y D.5.2 [Ferré, 02b]. Los cambios realizados para acomodarse a las necesidades reales de los socios industriales como organizaciones de desarrollo de software se ven reflejados en los entregables D.6.6 [Ferré, 04c] y D.1.7 [Folmer, 04] (capítulo 5 dedicado al WP5).

La interacción con los socios industriales y el contacto proyectos reales de desarrollo de software, con las presiones propias de dichos proyectos, nos han permitido dar forma a una propuesta más adaptada a la realidad empresarial y, por tanto, con mayores garantías de éxito. La aplicación del marco de integración por parte de la compañía LogicDIS al desarrollo de su sistema e-Suite nos ha permitido comprobar, así mismo, la viabilidad de la propuesta en un entorno real de desarrollo de un producto comercial. Tal aplicación ha contado con nuestro asesoramiento y con la impartición de un curso de usabilidad por parte del autor del presente trabajo, donde se presentaba el contenido de esta investigación.

Para detallar la aplicación del marco de integración por parte de LogicDIS, en primer lugar vamos a presentar el proyecto e-Suite, para pasar a continuación a detallar la situación previa de LogicDIS y cómo se ha afrontado la integración de la usabilidad en su proceso de desarrollo. Los resultados de la aplicación del marco de integración se discuten a continuación, en las vertientes de grado de integración de la usabilidad y de mejora del nivel de usabilidad percibido por los usuarios.

8.2.1 Aplicación e-Suite

La empresa LogicDIS se dedica principalmente al segmento de negocio de ERPs (*Enterprise Resource Planning*), copando un 40% de dicho segmento en Grecia (en 2002). Mientras la mayoría de productos ERP ofrecen un gran número de funcionalidades y cubren la mayor parte de necesidades de una empresa, se centran principalmente en usuarios internos, que desarrollan su trabajo en las instalaciones de la propia compañía. LogicDIS ofrece un producto, e-Suite, en forma de aplicación web que exporta funcionalidad básica de ERP a Internet, permitiendo a los usuarios disfrutar de las ventajas del ERP sin tener acceso físico al mismo. Los usuarios típicos de este producto son personal de ventas, clientes y ejecutivos de otras compañías. E-Suite soporta todos los ERPs con los que trabaja LogicDIS, constituyéndose de este modo en una herramienta multi-ERP. Así mismo, su diseño permite trabajar con distintos sistemas gestores de bases de datos, pues no cuenta con restricciones relacionadas con sistemas específicos. La aplicación está implementada en Java con la finalidad de hacerla independiente de la plataforma y facilitar su instalación y mantenimiento. Una descripción más detallada del proyecto e-Suite se puede encontrar en [Tsirikos, 04].

LogicDIS se planteó el desarrollo de una nueva versión de e-Suite para incrementar la usabilidad de la versión anterior sin alterar su funcionalidad. De esta forma, se permitía la comparación de la versión anterior de e-Suite con la nueva versión en cuanto a usabilidad.

8.2.2 Prácticas de Desarrollo en LogicDIS

La compañía LogicDIS tiene más de 1200 empleados [STATUS, 01], aunque para el estudio que nos ocupa únicamente nos hemos centrado en el equipo de *e-business* de la compañía,

encargado del desarrollo de la nueva versión de e-Suite. Dicho equipo de desarrollo es el que ha participado en el proyecto STATUS por parte de la empresa LogicDIS.

Tras una visita a la sede de LogicDIS en Atenas por parte del autor del presente trabajo de investigación en julio de 2002, se comprobó que el proceso de desarrollo llevado a cabo por el equipo de *e-business* de LogicDIS tiene un carácter informal, aunque se acomoda al esquema reflejado en la Figura 8.1.

En la figura se muestran las actividades principales del desarrollo en cajas ovaladas con fondo gris y etiquetadas con una letra de la A a la D, mientras que los productos del desarrollo más relevantes aparecen en la parte inferior representados como cajas con el borde con trazo discontinuo. La secuencia entre actividades viene dada por los productos que genera cada una y los que requiere para su ejecución, los cuales están representados en la figura mediante flechas que unen actividades y productos. La iteratividad que proporciona el equipo de LogicDIS al proceso viene dada por el modo de trabajo de cara al cliente. A pesar de que se establecen unos requisitos funcionales, la primera versión del producto software entregada al cliente es tratada como una versión Alfa. Así, sirve de base para desarrollar el producto según necesita el cliente, en un esfuerzo iterativo compartido entre LogicDIS y el cliente para la obtención del sistema que satisfaga al cliente. En la base de este proceder está la confianza mutua que surge con los clientes habituales y que se asienta, así mismo, en el prestigio de LogicDIS.

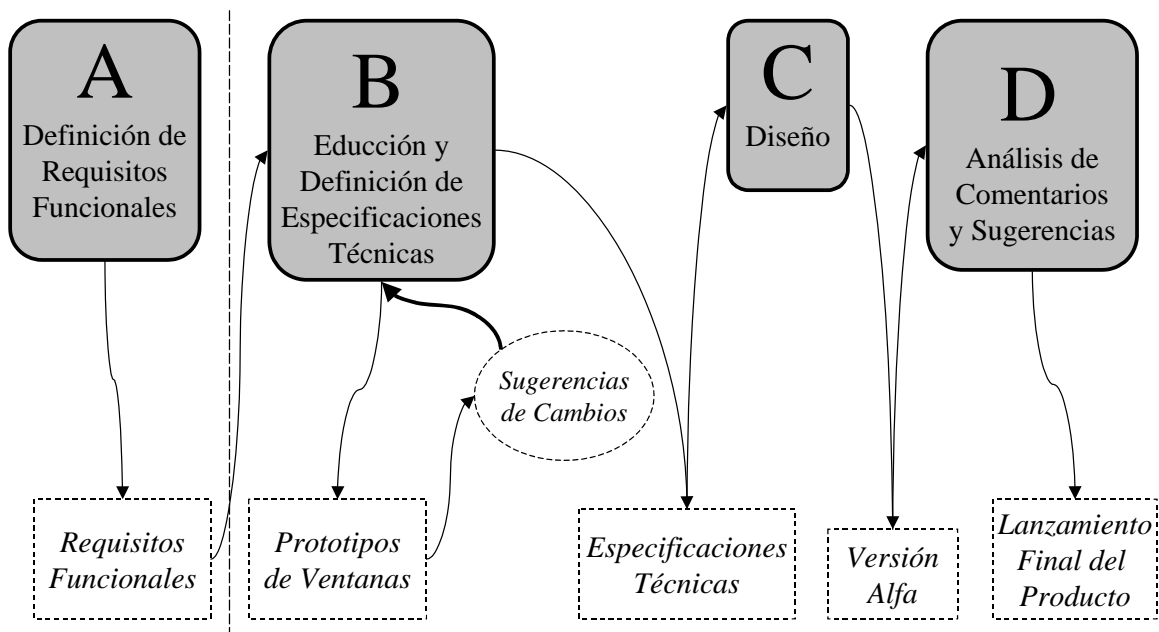


Figura 8.1 – Proceso de Desarrollo Seguido por el equipo de desarrollo de LogicDIS

En lo referente a la situación previa en cuanto a tratamiento de la usabilidad, el equipo de desarrollo no tenía conocimiento sobre técnicas IPO específicas. Únicamente algunos miembros del equipo tenían una idea vaga de la usabilidad. La usabilidad de sus productos no resultaba de un nivel muy deficiente debido a que las prácticas de desarrollo se basan en un alto grado de contacto con el cliente. Pero, aún así, el director del equipo de *e-business* reconocía que la usabilidad no se trataba de forma adecuada, y que existía un alto margen de mejora a este respecto. El director del equipo reconocía la importancia de la usabilidad en el sector *e-business*, e identificó un adecuado tratamiento de la usabilidad como un medio de diferenciación de su empresa frente a la competencia.

En este escenario, el resultado del presente trabajo de investigación constituyó la base de la inclusión por parte del equipo de desarrollo de la nueva versión de e-Suite, de técnicas IPO

con el fin de mejorar el tratamiento de la usabilidad en su proceso de desarrollo. La estrategia que se siguió para incorporar la usabilidad a sus prácticas de desarrollo se detalla en la siguiente sección.

8.2.3 Estrategia para la Integración de la Usabilidad

La transferencia tecnológica no resulta tarea fácil en la IS y menos en el campo del proceso software. La modificación de las prácticas de desarrollo requiere un esfuerzo importante por parte de los desarrolladores y de la empresa para la que desarrollan su labor. Al igual que cualquier otro cambio que suponga una modificación de la forma de trabajo de un grupo de profesionales, el cambio que proponemos en el proceso de desarrollo para la integración de prácticas de usabilidad, es preciso gestionarlo cuidadosamente de tal forma que las ventajas de dar dicho paso se hagan tangibles, y la dificultad del cambio de prácticas se reduzca en la medida de lo posible. A pesar de que la dirección de la organización sea favorable a la integración de la usabilidad en el desarrollo, no es suficiente con un documento teórico proporcionado a los desarrolladores para que éstos cambien la forma en la que construyen software. En particular, la preocupación por la usabilidad supone un cambio en la filosofía y aproximación al desarrollo con respecto a la habitual de los ingenieros software. Por estas razones, resulta necesario suplementar los resultados del presente trabajo de investigación, resumidos en el capítulo 7, con un plan de formación de los desarrolladores en los principios básicos de usabilidad y en el empleo de las técnicas IPO particulares que más se acomoden a las características del equipo de desarrollo y del tipo de proyectos que llevan a cabo. Un borrador de un plan de formación genérico de este tipo se detalla en la parte tercera del entregable D.5.2 del proyecto STATUS [Ferré, 02b].

Para la integración de la usabilidad en las prácticas de desarrollo del equipo de *e-business* de LogicDIS, planificamos un curso de introducción a la usabilidad y a las técnicas que mejor podían servir a los propósitos de LogicDIS. La estructura del curso se dividió en tres partes, según se detalla a continuación:

- Parte I - Introducción a la usabilidad como atributo de calidad: El objetivo de esta parte consistía en familiarizar a los desarrolladores con la idea de usabilidad de un sistema software en primer lugar, para pasar a detallar cómo se descompone el objetivo abstracto que supone la usabilidad. Para destacar la relevancia que la usabilidad tiene en el desarrollo de software en la actualidad se abordó en esta parte del curso cómo las grandes compañías (IBM, Microsoft, etc.) abordan la usabilidad, y en qué nivel de usabilidad se mueven las aplicaciones web. El enfoque de desarrollo centrado en el usuario y sus implicaciones en el proceso de desarrollo habitual se incluyeron también en esta parte.
- Parte II - Técnicas de usabilidad: Para plantear esta parte del curso se tuvieron en cuenta las características del equipo *e-business* de LogicDIS y del proyecto e-Suite para seleccionar para ellos, mediante el marco de integración propuesto en el presente trabajo, las técnicas IPO que mejor podían servir al propósito de integración. En esta parte del curso se realizó una introducción a cada técnica individual, de tal forma que los desarrolladores pudieran tener una idea general de su mecanismo de aplicación y de cómo podía contribuir en el desarrollo de software, y se les proporcionaba, en la documentación del curso, bibliografía a la que acudir para formarse en el uso de la técnica con un nivel mayor de profundidad. Para algunas de las técnicas, aquellas que mejor encajaban en los objetivos del proyecto de rediseño de e-Suite, se realizaron ejercicios a modo de entrenamiento.
- Parte III - Relación con los productos existentes en el proceso de LogicDIS: Tomando como base la descripción de actividades de la Figura 8.1 se abordó con los desarrolladores cómo podría realizarse la integración de las técnicas IPO seleccionadas según el marco,

en sus prácticas concretas de desarrollo en el proyecto de rediseño de e-Suite. La integración que se sugirió es la que se muestra en la Figura 8.2, en la que se ha añadido a las actividades de la Figura 8.1 las técnicas IPO que se pueden aplicar en las cajas que aparecen en la parte inferior de la figura.

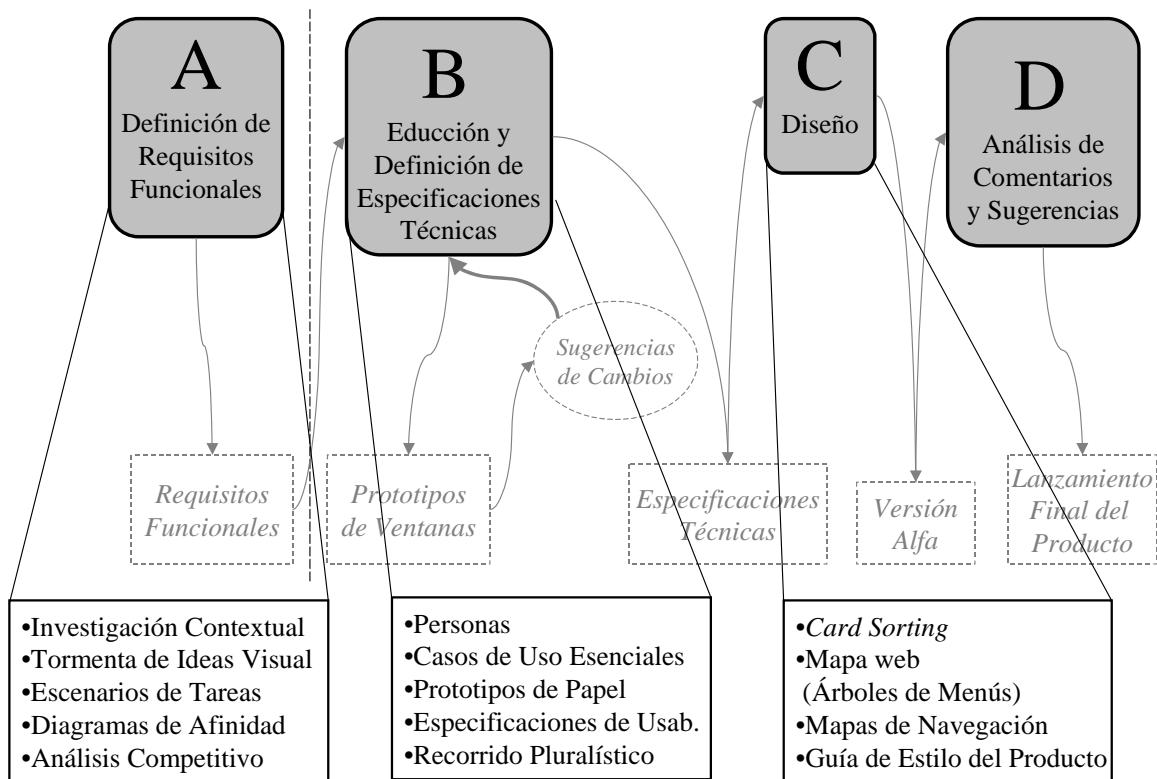


Figura 8.2 - Sugerencia de Integración de Técnicas IPO en el Proceso de Desarrollo de LogicDIS

Este curso de formación en usabilidad fue impartido por el autor del presente trabajo del 18 al 22 de enero de 2004 al equipo de desarrollo de la nueva versión de e-Suite en las oficinas de la compañía en Atenas (Grecia). El curso tuvo una receptividad que podemos considerar alta y que contó con la ventaja del apoyo del director del equipo y del dinamismo que presenta el equipo, formado principalmente por desarrolladores jóvenes con interés en recibir formación en temas de creciente actualidad.

8.2.4 Resultados de la Aplicación del Marco de Integración

Para evaluar el impacto que ha tenido la integración de la usabilidad en las prácticas de desarrollo del equipo de *e-business* de LogicDIS tenemos dos componentes. Por un lado, podemos evaluar el impacto que ha tenido en el proceso de desarrollo en cuanto a: integración de técnicas IPO individuales en el desarrollo; y a la adopción de un enfoque de desarrollo centrado en el usuario por parte del equipo de desarrollo. Por otro lado, podemos comparar la mejora en usabilidad que la nueva versión del producto software ha adquirido en comparación con la versión anterior.

8.2.4.1 Impacto en el Proceso de Desarrollo

Tras la finalización del desarrollo de la nueva versión de e-Suite por parte del equipo de *e-business* de LogicDIS, distribuimos el cuestionario que se incluye en el Anexo B entre los asistentes al curso. Estos profesionales tienen una experiencia de 3 a 8 años en el desarrollo

de software, por lo que podemos considerarlos desarrolladores con un nivel de experiencia media-alta.

Tomando en cuenta el agregado de las técnicas que los desarrolladores nombran, las técnicas IPO aplicadas en el proyecto fueron:

- Técnicas observacionales (Investigación Contextual).
- Escenarios de Tareas.
- Diagramas de Afinidad.
- Análisis Competitivo.
- Personas.
- Casos de Uso Esenciales.
- Especificaciones de Usabilidad.
- Recorrido Pluralístico.
- Pensar en Voz Alta.
- Cuestionarios de Usabilidad.

Por tanto, el equipo de desarrollo ha aplicado 10 de las técnicas propuestas en el marco de integración. Se trata de un número de técnicas alto, si se considera que anteriormente no se aplicaba ninguna de las técnicas IPO por desconocerlas. Así pues, **la integración de la usabilidad puede considerarse muy exitosa desde el punto de vista del número de técnicas nuevas**. El éxito en la aplicación de estas 10 técnicas puede animar al equipo a aplicar en futuros proyectos otras técnicas IPO no consideradas para el diseño de la nueva versión de e-Suite.

Las impresiones que los desarrolladores detallan sobre el uso de las técnicas IPO en este proyecto, y su influencia en el resto de prácticas de desarrollo, muestran que el equipo valora la aportación de las prácticas IPO en su trabajo de desarrollo y que la percepción de la importancia de la usabilidad ya no se ciñe únicamente a lo relacionado con la interfaz gráfica de usuario, sino que también ha afectado a otras actividades del desarrollo. El grado de acuerdo con la afirmación de que el resultado de las actividades de usabilidad ha afectado a la parte visible de la interfaz gráfica de usuario es de 4,33 de media en una escala de 1 a 5, mientras que en lo referente al desarrollo del resto del sistema la media es de 4, como puede observarse en la Tabla 8.1. Nótese que esto es un cambio radical de mentalidad con respecto al desarrollador medio que cree que la usabilidad se refleja únicamente en la interfaz [Ferré, 01a]. Se debe resaltar que un cambio tan importante logrado con un único proyecto de desarrollo es un indicio importante del éxito del marco aquí propuesto.

Tabla 8.1 - Valoración del Impacto de las Prácticas de Usabilidad

Por favor, indique cuánto el resultado de las actividades de usabilidad ha afectado a las siguientes actividades	No ha afectado en absoluto				Ha afectado profundamente	Valor Medio
	1	2	3	4	5	
Desarrollo de la parte visible de la interfaz gráfica de usuario	0%	0%	0%	66,67%	33,33%	4,33
Desarrollo del resto del sistema	0%	0%	33,33%	33,33%	33,33%	4

Entre las ventajas de la utilización de técnicas IPO durante el desarrollo nombradas por los desarrolladores, podemos destacar:

- la mejora de comunicación entre desarrolladores y usuarios,
- una mejor comprensión de la forma de hacer las cosas de los usuarios y de sus necesidades en general,

- y menor necesidad de realizar cambios en el diseño por estar desarrollando el sistema adecuado.

En particular, valoran positivamente la aportación de las técnicas IPO en la comprensión de requisitos, especialmente en lo referente a la comprensión del problema y en la obtención de información de los usuarios.

Como conclusión general, el equipo de desarrollo valora positivamente el uso de técnicas IPO a la vista de los resultados obtenidos. Un 66,67% de los desarrolladores participantes en el estudio están de acuerdo con la afirmación de que, globalmente, el esfuerzo empleado en la aplicación de las técnicas IPO vale la pena considerando los resultados obtenidos; mientras que el 33,33% está completamente de acuerdo, como se muestra en la Tabla 8.2.

Tabla 8.2 - Valoración Global de la Aplicación de Técnicas IPO

Por favor, indique su grado de acuerdo o desacuerdo con la siguiente afirmación	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Completamente de acuerdo
Globalmente, el esfuerzo empleado en la aplicación de técnicas de usabilidad vale la pena, considerando los resultados obtenidos	0%	0%	66,67%	33,33%

La respuesta a esta cuestión viene reforzada por la afirmación por parte de los desarrolladores de que este tipo de técnicas son aplicables a un rango amplio de proyectos. Podemos destacar las respuestas concretas que se refieren a que las técnicas IPO son aplicables siempre que haya interacción con personas, y la que indica que son aplicables, pero en conjunción con otro tipo de técnicas. Esta última afirmación va en consonancia con el enfoque tomado en el presente trabajo de investigación de integrar la aplicación de técnicas IPO con el resto de técnicas IS, en vez de abordar el desarrollo con dos procesos paralelos con puntos de integración limitados.

8.2.4.2 Impacto en la Usabilidad Percibida por los Usuarios

Para LogicDIS, como empresa de construcción de software, es muy importante la percepción que el cliente tenga de la calidad de los productos que desarrolla. En concreto, la usabilidad, puesto que es directamente perceptible por los usuarios, es especialmente relevante para LogicDIS y la compañía cuenta con un gran interés por comprobar si la aplicación de las técnicas IPO propuestas según el presente trabajo de investigación es percibida en el producto software por parte de usuarios expertos en el uso del sistema. El objetivo de tener a los clientes satisfechos es, por tanto, de vital importancia para LogicDIS.

En el marco del proyecto STATUS se planteó un estudio de la usabilidad del producto e-Suite, el cual se realizó con usuarios tanto de la versión anterior como de la nueva versión (desarrollada con la integración de técnicas IPO en el proceso de desarrollo).

Usuarios representativos de la aplicación pertenecientes a dos compañías clientes de LogicDIS (Germanos y Friesland) participaron en el estudio mediante cuestionarios que rellenaron según su experiencia de la usabilidad de cada versión del producto software. Según se detalla en el entregable D.6.3 del proyecto STATUS [Juristo, 04], la media de usabilidad del producto software de la versión anterior de e-Suite es de 2,73 (en una escala de 1 a 5), mientras que el valor recogido para la nueva versión de e-Suite es de 3,41. Por lo tanto, la usabilidad del sistema según se percibe por parte de los usuarios mejoró en un **24,9%** respecto a la versión anterior. De forma adicional, el nivel de usabilidad pasó de estar por debajo del valor medio en la escala empleada (un 3 en la escala de 1 a 5) a estar por encima. Estos resultados confirmaron para LogicDIS **la mejora en la usabilidad de su producto gracias a la integración de técnicas IPO en sus prácticas de desarrollo**, y nos permite aseverar la efectividad percibida de las técnicas IPO integradas en el proceso de desarrollo.

8.3 Casos de Estudio

Frente al estudio de viabilidad presentado en la sección anterior, en el que un equipo de desarrollo formado por profesionales con una experiencia media-alta aplican el marco de integración de la usabilidad en el proceso, queríamos evaluar cómo podría abordarse la integración con el marco propuesto en el caso de desarrolladores noveles o con una experiencia baja-media en el desarrollo de software.

Para una empresa consolidada y líder en su segmento de mercado como es LogicDIS, es razonable pensar que produce unos productos con un nivel de usabilidad no deficiente, puesto que su prestigio se basa en dicho nivel aun cuando no traten la usabilidad adecuadamente. Sin embargo, para un equipo de desarrollo formado por desarrolladores noveles o con escasa experiencia, el riesgo de desarrollar productos con un nivel de usabilidad deficiente es, en principio, mucho mayor, puesto que su base de conocimiento del desarrollo software no incluye temas relacionados con la usabilidad más que de forma tangencial. Por esta razón, considerábamos que el marco de integración de la usabilidad en el proceso de desarrollo propuesto en el presente trabajo de investigación podría afectar de forma diferente a equipos de desarrollo con menor experiencia en desarrollo de software.

Con este fin, planteamos, por una parte, la integración de la usabilidad en un trabajo de tesis fin de master (desarrollador con algo de experiencia y con una visión madura del desarrollo de software) y, por otra parte, la integración de la usabilidad en el desarrollo de trabajos prácticos de una asignatura de grado (desarrolladores noveles con escaso conocimiento no académico).

8.3.1 Caso de Desarrollador con Experiencia Media

8.3.1.1 Aplicación de Gestión de Calendarios

Con el fin de evaluar la aplicación del marco de integración a un desarrollo realizado por un ingeniero software con experiencia media en la construcción de software y con un conocimiento acerca del campo de la IS más maduro que el que puede tener un estudiante de grado, se encomendó a un estudiante del Master de Ingeniería del Software de la Universidad Politécnica de Madrid el desarrollo de un sistema interactivo para la gestión de calendarios, como trabajo de tesis fin de master.

El sistema tenía como destinatario la coordinadora del master, para las tareas propias de elaboración de un calendario de clases para cada edición del master, teniendo en cuenta los profesores que imparten cada módulo, y la labor de realizar cambios en el calendario para encajar las circunstancias que salen a la luz en el proceso de elaboración del mismo. Se trata de un proyecto con un enfoque abierto, que permita su uso por un abanico amplio de usuarios, pero que está centrado en las tareas que necesita desarrollar la coordinadora del master. Así pues, se trata de un caso de desarrollo de software a medida, donde se dispone de un usuario representativo para involucrar en el proceso de desarrollo.

Se dispone de un sistema software realizado por un estudiante del master en el año 2001 que se ocupa de la tarea de elaborar calendarios para distintas ediciones de master. Se contó con este sistema anterior, desarrollado sin la aplicación de técnicas IPO, como punto de referencia en cuanto a la usabilidad esperada del nuevo sistema, pues se pretendía construir un sistema que al menos mejorara el nivel de usabilidad del sistema existente.

8.3.1.2 Integración de la Usabilidad en el Proceso de Desarrollo

En la Figura 8.3 se muestra el proceso seguido por la mayor parte de los alumnos de master cuando llevan a cabo su tesis fin de master en el paradigma de desarrollo orientado a objetos. Como puede observarse en la figura, se tiene una primera fase de Planificación y Especificación de Requisitos y a continuación se abordan los ciclos de construcción. Por tratarse de un proceso de carácter pedagógico, no se suelen llevar a cabo más de dos ciclos. Nótese que, aunque no está reflejado en la figura existe una retroalimentación entre las etapas dentro de un mismo ciclo.

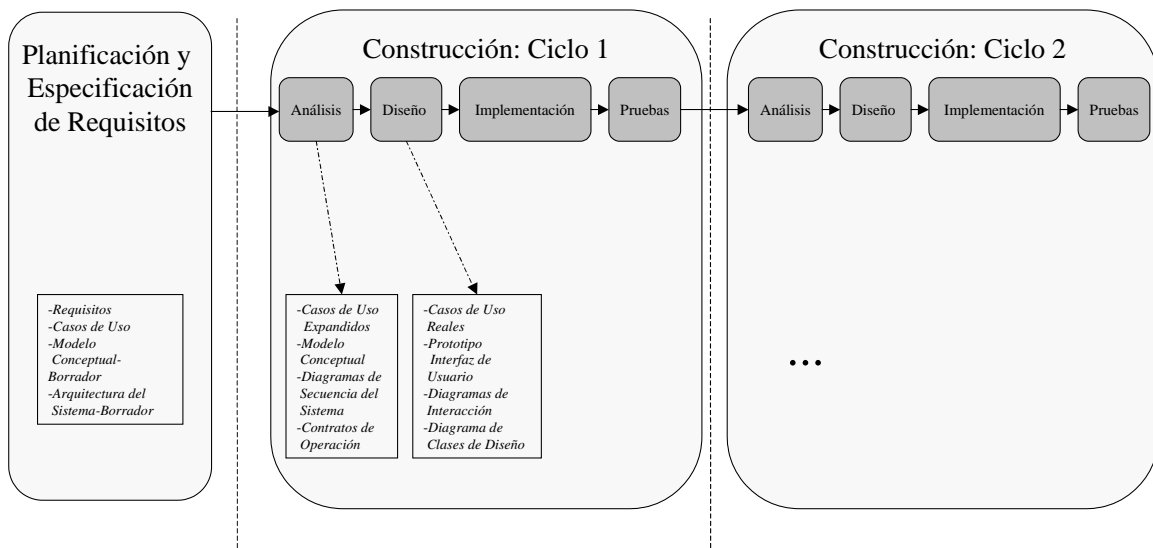


Figura 8.3 - Proceso Típico Seguido por Alumnos de Master

Debido a que la mejora de la usabilidad del sistema anterior se propuso como objetivo de especial relevancia, este proceso típico se ha visto alterado en gran medida en la primera fase. El desarrollo del sistema no ha concluido en el momento de finalización del presente trabajo de tesis doctoral, pero se encuentra suficientemente avanzado como para permitir el estudio de cómo se ha llevado a cabo la integración de la usabilidad en el proceso realizado hasta el momento, puesto que se ya se dispone de un prototipo software.

En el proceso llevado a cabo en este caso de estudio, la fase de Planificación y Especificación de Requisitos se ha expandido de forma notable, y se ha dotado tanto de iteratividad como de un carácter exploratorio del que suele carecer en los desarrollos realizados por alumnos de este tipo. Las técnicas IPO han tomado el protagonismo en la primera fase del desarrollo, con respecto a las técnicas IS. En la Figura 8.4 se muestran las etapas en las que se ha dividido esta fase, junto con los productos de cada etapa (en cursiva) y las técnicas IPO empleadas (en las cajas que aparecen en la parte inferior de la figura). La fase se ha dividido en una etapa de exploración inicial y dos etapas de elaboración hasta llegar a un enfoque estable de la solución, debido al desconocimiento del dominio de la elaboración y gestión de calendarios de cursos tipo master por parte del alumno desarrollador. La inclusión de técnicas IPO se ha realizado mayoritariamente en esta fase inicial, puesto que en fases posteriores del desarrollo únicamente se ha planteado la necesidad de realización de test de usabilidad adicionales, con sus cuestionarios correspondientes.

La selección de las técnicas IPO concretas a emplear en el proyecto se realizó mediante la aplicación del marco que se propone en el presente trabajo de investigación, por parte del desarrollador del sistema con el asesoramiento del autor del presente trabajo. En concreto, se ha identificado por parte del desarrollador la necesidad de llevar a cabo un contacto continuo

a lo largo del desarrollo con el usuario representativo del que se dispone, lo cual no se trata de una aproximación común en este tipo de desarrolladores.

Planificación y Especificación de Requisitos

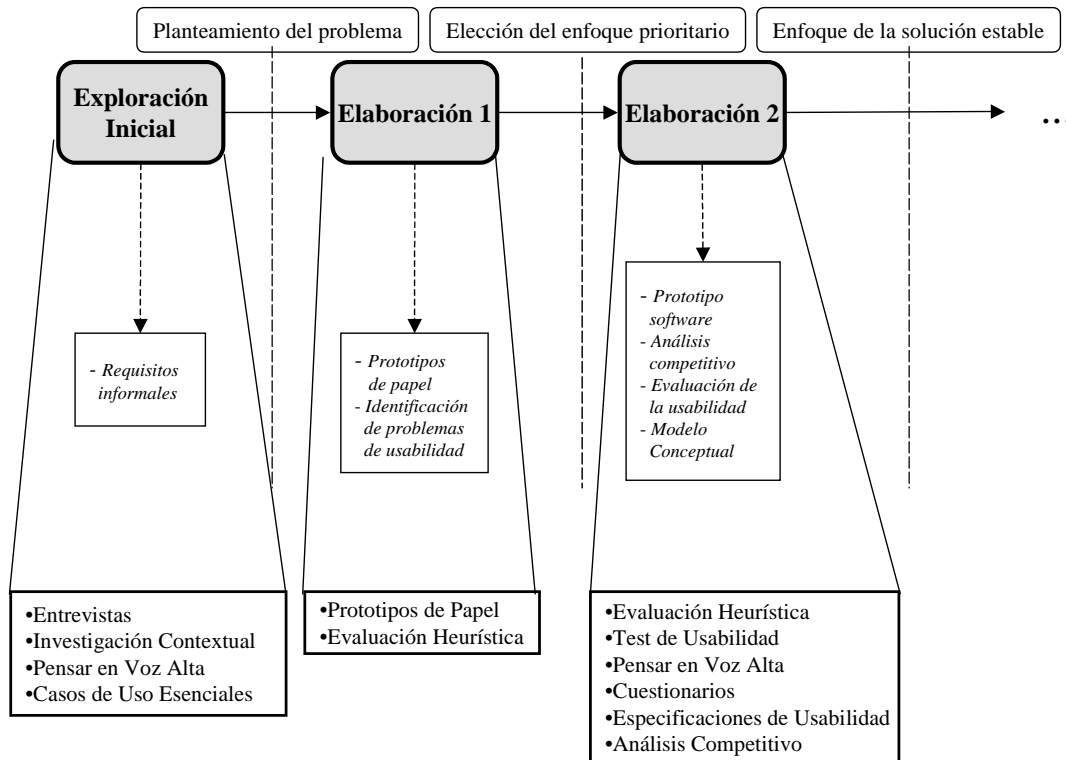


Figura 8.4 - Fase de Planificación y Especificación de Requisitos Llevada a Cabo en el Caso de Estudio

8.3.1.3 Resultados Preliminares de la Aplicación del Marco de Integración

Como se ha indicado más arriba, el desarrollo del sistema no ha finalizado, pero se dispone de un prototipo ejecutable y el enfoque de la solución es estable. Así, consideramos que se dispone de suficientes elementos para poder realizar una valoración de las posibilidades que el uso del marco de integración propuesto ha ofrecido en el desarrollo realizado.

8.3.1.3.1 Impacto en el Proceso de Desarrollo

El cuestionario que se incluye en el Anexo C fue rellenado por el desarrollador de la aplicación. Los resultados muestran que el grado de integración de las distintas técnicas IPO es alto, no sólo con las actividades directamente relacionadas con la parte visible de la interfaz gráfica de usuario, sino también con el resto de actividades del desarrollo. En concreto el desarrollador nombra que la forma de la solución final ha sido establecida en gran parte debido a la aportación de las técnicas IPO. La aportación que las técnicas IPO han realizado a la mejora en la relación con el usuario ha consistido en la posibilidad de abordar nuevas formas de interacción. Los requisitos han sido revisados y reformulados más que en anteriores desarrollos debido, así mismo, a la inclusión de un mayor número de técnicas relacionadas con los mismos (las técnicas IPO). Por otra parte, los Test de Usabilidad y la Evaluación Heurística se identifican como técnicas que han permitido mejorar la interfaz gráfica de usuario y los procesos que el usuario lleva a cabo con el sistema.

En cuanto a la conveniencia de la aplicación de técnicas IPO en el desarrollo, el desarrollador valora positivamente el resultado ofrecido por las técnicas aplicadas en comparación con el esfuerzo empleado en su aplicación, por lo que consideramos que **se ha cumplido el objetivo de facilitar la integración**, al haber posibilitado el marco de integración la elección por parte del desarrollador de aquellas técnicas que podían ofrecerle buenos resultados en cuanto a mejora de la usabilidad. En lo referente a la posibilidad de integrar la usabilidad en futuros desarrollos, estimamos que el desarrollador la considerará con gran probabilidad, puesto que indica que la aplicación de técnicas IPO es adecuada para cualquier problema con el tamaño adecuado y donde el usuario final sea el protagonista de la actividad del software.

8.3.1.3.2 Impacto en la Usabilidad del Producto

Como se ha indicado más arriba, disponíamos de un sistema realizado con fines similares, pero sin la aportación de técnicas IPO en su desarrollo. Esta aplicación, denominada Sistema Inteligente de Gestión de Master [Chavez, 01], se ha probado tanto con la coordinadora del master como con otros usuarios conocedores del dominio de aplicación. El grupo de usuarios adicionales estaba formado por alumnos del Master de Ingeniería del Software, los cuales, debido a su condición de alumnos, conocían el dominio por manejar como consumidores los calendarios gestionados por la coordinadora del master. La decisión de realizar test de usabilidad con este grupo adicional de usuarios se tomó por no reducir el estudio a únicamente un usuario.

Los resultados de la aplicación de los test de usabilidad muestran que los usuarios perciben el sistema desarrollado con el marco propuesto como mucho más satisfactorio que el sistema anterior, según se puede observar en la Tabla 8.3, donde se detallan las respuestas de usuarios que han probado ambas versiones al cuestionario de satisfacción que se incluye en el Anexo D. Se puede observar que todos los usuarios participantes en el estudio identificaban como mejor la versión nueva sobre la anterior, y entre ellos un 46,15% daban la máxima puntuación (2) en dicha comparación.

Tabla 8.3 - Valoración Expresada Directamente por los Usuarios en Comparación con la Versión Anterior del Sistema

	Peor		Mejor		Valor Medio	
	-2	-1	0	1		2
Valoración respecto a la versión anterior	0%	0%	0%	53,85%	46,15%	1,46

Esta diferencia en la usabilidad percibida por los usuarios se refleja también en la valoración por separado de cada sistema, detallada en la Tabla 8.4. Mientras la valoración media del sistema anterior es negativa (-0,67 en una escala de -2 a 2), la valoración del sistema nuevo es positiva (0,58).

Tabla 8.4 - Comparativa de la Satisfacción Expresada por Usuarios de Ambas Versiones

Satisfacción General	Insatisfecho			Satisfecho		Valor Medio
	-2	-1	0	1	2	
Sistema anterior	0%	66,67%	33,33%	0%	0%	-0,67
Prototipo Realizado con la Integración de la Usabilidad a Través del Marco Propuesto	0%	16,67%	16,67%	58,33%	8,33%	0,58

Los usuarios cometen también menor número de errores con el nuevo sistema, teniendo una tasa de finalización con éxito de las tareas encomendadas mejor que con el sistema anterior.

Todos estos datos nos indican que el **objetivo de producir un sistema con mejor usabilidad que el sistema anterior existente ha sido alcanzado**. Así, observamos que la aplicación del

marco de integración propuesto ha permitido al desarrollador la obtención de un producto software con un nivel de usabilidad superior al mínimo establecido en el inicio del proyecto.

8.3.2 Caso de Desarrolladores Noveles

8.3.2.1 Proyecto Práctico de Construcción de un Sistema Software

Para poder valorar las posibilidades de la propuesta en distintos ámbitos de aplicación, consideramos también el uso del marco de integración en los trabajos prácticos de una asignatura de grado de la titulación de Ingeniería Informática impartida en la Facultad de Informática de la Universidad Politécnica de Madrid en el curso 2003-2004.

Los alumnos que están finalizando sus estudios de ingeniero informático tienen un conocimiento limitado del mundo real de desarrollo de software, puesto que su experiencia se limita a una visión académica del desarrollo, basada principalmente en la IS, y desconocen casi completamente el campo de la IPO al comenzar a cursar la asignatura. En cuanto a su concepción del proceso, es un tanto inmadura, en el sentido de poco definida. Lo único que conocen con algo de profundidad son los productos que se les exigen en los trabajos prácticos que desarrollan a lo largo de la carrera. La aplicación del marco de integración propuesto por estos alumnos se planteó para evaluar las posibilidades de aplicación de la propuesta en equipos formados por ingenieros software noveles, con un proceso poco definido y con escasos conocimientos de usabilidad.

La asignatura es de carácter optativo, se denomina Proyecto Práctico de Construcción de un Sistema Software, y se imparte en el último curso de la titulación (quinto). Tiene un carácter eminentemente práctico y se basa en el desarrollo de un sistema software en equipos de 3-4 personas, el cual se lleva a cabo en clases tipo taller. Cada equipo de trabajo aborda un problema distinto, enunciado por otro grupo de compañeros.

Para poder comparar la posible mejora en usabilidad en el producto desarrollado, únicamente en uno de los proyectos se contó con un prototipo desarrollado que permitiera la comparación. En dicho proyecto se pretendía desarrollar una aplicación que coincidía en objetivos con el proyecto llevado a cabo por estudiantes de la misma asignatura en el curso anterior (2002-2003). En dicho curso los alumnos no disponían del marco de integración propuesto, y únicamente contaron con una clase de introducción al tema de la usabilidad y la IPO. Así, pudimos contar con un producto realizado por un equipo formado por desarrolladores con el mismo nivel de experiencia, en el cual habían tratado la usabilidad según sus conocimientos previos (sin el marco de integración propuesto).

8.3.2.2 Integración de la Usabilidad en el Proceso de Desarrollo

Los equipos de trabajo que se formaron en la asignatura tenían libertad para decidir las técnicas concretas que aplicarían en su proyecto, pero todos ellos desarrollaban según el paradigma orientado a objetos y decidían ellos mismos qué técnicas concretas aplicar y cómo en su labor de construcción de un sistema software. El tipo de proceso que siguen los alumnos en esta asignatura no está, por tanto, definido. Sí tiene como característica común el que el desarrollo se aborda en dos ciclos y, así, se trata de un desarrollo iterativo. De esta forma cumple con la única condición que se requiere para la aplicación de la solución propuesta en el presente trabajo.

La usabilidad se propuso como objetivo de calidad relevante a los 10 equipos de trabajo que formaron los alumnos de la asignatura. Se seleccionaron las técnicas IPO que podían adecuarse mejor a las características de la asignatura y a las necesidades de la mayoría de

proyectos realizados utilizando el marco aquí propuesto, y se les dio a los alumnos una breve formación en cada una de las técnicas. Las técnicas en las que se formó a los equipos son las siguientes:

- Diagramas de Afinidad.
- Análisis Competitivo.
- Casos de Uso Esenciales.
- Tormenta de Ideas Visual.
- Escenarios y *Storyboards*.
- Prototipos de Papel.
- Especificaciones de Usabilidad.
- Árboles de menús / Mapa Web.
- Mapa de Navegación.
- Recorrido Pluralístico.
- Evaluación Heurística.
- Pensar en Voz Alta.
- Test de Usabilidad .
- Cuestionarios.

El desarrollo se dividió en dos ciclos al final de los cuales debía realizar cada equipo de desarrollo un test de usabilidad con sus usuarios correspondientes, para conocer el nivel de usabilidad del sistema. Excepto en el uso de la técnica de test de usabilidad, cada equipo escogió libremente qué técnicas IPO aplicar en el desarrollo de su sistema, así como en qué ciclo(s) aplicar cada técnica elegida.

8.3.2.3 Resultados de la Aplicación del Marco de Integración

Para evaluar el impacto de la aplicación del marco de integración propuesto, nos vamos a centrar principalmente en la influencia que se ha observado en el proceso de desarrollo, aunque también apuntaremos algunos datos referentes al impacto sobre el producto.

8.3.2.3.1 Impacto en el Proceso de Desarrollo

Tras la finalización de los dos ciclos de desarrollo que se realizaron en cada proyecto de la asignatura, se distribuyó entre los estudiantes un cuestionario anónimo. Dicho cuestionario se incluye en el Anexo E.

Los cuestionarios recogidos reflejan un uso bastante alto de las técnicas seleccionadas: Únicamente 3 de las 16 técnicas propuestas no fueron utilizadas por ningún estudiante (Diagramas de Afinidad, Análisis Competitivo y Recorrido Pluralístico). Ninguna de las tres técnicas es cercana a la IS según el marco de integración propuesto, y ésa puede ser una razón por la que ningún equipo ha decidido emplearla; lo que ratifica la importancia de incluir este tipo de información en el marco, pues se observa que, tal como esperábamos, la cercanía a la IS es un criterio a la hora de que los desarrolladores decidan aplicar técnicas IPO.

Podemos destacar en su valoración de las técnicas empleadas, que los valores medios indican que para ninguna de ellas consideran los desarrolladores que resulten difíciles de aplicar ni que ralenticen innecesariamente el desarrollo. En la Tabla 8.5 se pueden observar las consideraciones al respecto detalladas por los estudiantes en los cuestionarios. Los valores de

la tabla indican el grado de acuerdo con la afirmación que encabeza la columna en una escala 1-5, donde 1 significa "nada de acuerdo" y 5 "completamente de acuerdo".

En lo referente a la valoración de la aportación que haya tenido la aplicación de las técnicas IPO en diversos aspectos del desarrollo, los participantes en el estudio claramente identifican la aplicación de este tipo de técnicas como una ayuda en el desarrollo de la parte visible de la interfaz gráfica del usuario (media de 4,03 en una escala 1-5). Sin embargo, el grado de acuerdo con la afirmación de que la aplicación de las técnicas ha sido de ayuda en el desarrollo del resto del sistema, baja hasta una media de 2,48. Por tanto, parece que la mayoría de los participantes no ha identificado la posible integración de las técnicas IPO con las utilizadas habitualmente en la IS. La visión predominante en la literatura de IS en la que, implícitamente, los temas de usabilidad se relacionan únicamente con la parte visible de la interfaz gráfica de usuario, podría estar detrás de esta resistencia por parte de los estudiantes de Ingeniería Informática a afrontar una integración de técnicas IPO en temas no directamente relacionados con la IU. Por otra parte, la falta de una visión clara de las actividades IS del desarrollo por parte de los alumnos no ayuda a que puedan identificar claramente cómo las nuevas actividades de usabilidad se integran con éstas.

En cuanto al esfuerzo empleado en las actividades de usabilidad, los participantes se inclinan claramente por **valorar positivamente los resultados obtenidos por la aplicación de las técnicas IPO**, puesto que el grado de acuerdo con la afirmación "El esfuerzo que conlleva aplicar las técnicas de usabilidad compensa a la vista de los resultados obtenidos" toma en media un valor de 4 en una escala 1-5.

Tabla 8.5 - Valoración de las Técnicas IPO Empleadas

Técnica IPO	La técnica es difícil de entender	La técnica es difícil de aplicar correctamente	La técnica es difícil de integrar con el resto de técnicas del desarrollo	El desarrollo se ha ralentizado innecesariamente al incluir la técnica
Casos de Uso Esenciales	2,46	2,65	2	2,23
Tormenta de Ideas Visual	1,31	1,56	1,5	1,69
Escenarios	1,7	2,19	2,19	2,11
<i>Storyboards</i>	1,33	1,67	1,67	1,25
Prototipos de Papel	1,47	1,79	1,63	1,89
Especificaciones de Usabilidad	2,33	2,33	2,78	2,56
Árboles de menús	1,83	1,83	1,5	1,67
Mapa Web	2	2,5	1,5	2,5
Mapa de Navegación	2	2	1,63	1,88
Evaluación Heurística	2	2,71	2	2,13
Pensar en Voz Alta	1,4	1,73	1,73	1,67
Test de Usabilidad	1,6	2,04	1,76	2,08
Cuestionarios	1,5	2	1,75	2,15

Tanto la valoración por parte de los participantes de la dificultad de aplicación de las técnicas, como la valoración positiva de las técnicas IPO empleadas en cuanto a los resultados obtenidos, **podemos considerarlo como un éxito para el marco de integración propuesto en este caso concreto**. Así, se ha cumplido el objetivo de facilitar a desarrolladores formados principalmente en los supuestos de la IS la tarea de integración de la usabilidad en su proceso de desarrollo. Dicha tarea, que supone un trabajo de cierta consideración para un desarrollador novel si no dispone de una herramienta como el marco propuesto, ha resultado relativamente sencilla según la valoración de los propios desarrolladores noveles.

8.3.2.3.2 Impacto en la Usabilidad Percibida del Producto

El grado de satisfacción expresado por los usuarios de los proyectos desarrollados en cuanto a la usabilidad de los productos obtenidos fue alto, aunque no contamos con una medida más precisa de la mejora en usabilidad producida por la aplicación del marco de integración de la usabilidad propuesto. Excepto en uno de los casos, en el que, como se ha indicado más arriba, se disponía de un producto desarrollado con los mismos objetivos que el construido con la aplicación del marco propuesto.

Para poder evaluar la posible mejora en la usabilidad del producto contábamos con un prototipo software desarrollado como producto final por los alumnos del curso anterior. Dicho prototipo se probó con los estudiantes que cursaron la asignatura en el curso 2003-2004. El cuestionario distribuido entre los participantes en el estudio se incluye en el Anexo D.

Los miembros del equipo que desarrolló el nuevo sistema integrando las técnicas IPO mediante el marco propuesto, realizaron a su vez test de usabilidad con usuarios que habían probado previamente el sistema anterior. En la Tabla 8.6 se detalla la respuesta dada en los cuestionarios a la pregunta de satisfacción general. Se puede observar una mejora apreciable en la satisfacción expresada por los usuarios de la nueva versión (2) en comparación con la expresada por los usuarios de la versión anterior (0,18), incluso si ésta ofrecía un nivel aceptable de satisfacción por parte de los usuarios.

Tabla 8.6 - Comparativa de la Satisfacción Expresada por los Usuarios de Ambas Versiones

Satisfacción General	Insatisfecho			Satisfecho		Valor Medio
	-2	-1	0	1	2	
Resultados prototipo curso 02/03 (sin el marco propuesto)	3,70%	22,22%	33,33%	33,33%	7,41%	0,18
Resultados prototipo curso 03/04 (con el marco propuesto)	0%	0%	0%	0%	100%	2

8.4 Discusión

Los resultados obtenidos nos permiten afirmar que **el uso del marco de integración propuesto es viable**, sirviendo a su propósito de facilitar la integración de técnicas y actividades relacionadas con la usabilidad en el proceso de desarrollo software.

En particular, la inclusión de técnicas IPO se ha realizado con éxito, obteniéndose una apreciación por parte de los desarrolladores participantes en los distintos casos de estudio de las ventajas de aplicación de las técnicas en cuanto a los resultados obtenidos. Partíamos de un apoyo por parte de la dirección a la integración de la usabilidad en el proceso de desarrollo, pues el presente trabajo de investigación toma como base dicho soporte. Sin embargo, para la aceptación de un cambio de las prácticas de desarrollo es muy importante contar también con la aquiescencia de los miembros del equipo de desarrollo. La elección de las técnicas IPO que mayor efectividad pueden tener para una organización y tipo de proyecto

en concreto resulta una tarea difícil, que el marco de integración pretende facilitar. Así, al haber empleado técnicas IPO que han producido un impacto favorable en el equipo de desarrollo podemos considerar que, en los casos descritos, **se ha cumplido el objetivo de favorecer la integración real de las técnicas IPO en las prácticas habituales del equipo de desarrollo.**

De las 35 técnicas ofrecidas en el marco de integración únicamente se han empleado entre 10 y 13 técnicas. La aplicación del total de técnicas recogidas en el marco propuesto como nuevas técnicas a integrar en un proceso que no las consideraba previamente es impensable. Precisamente, el uso que hemos previsto para el marco permitiría la selección de aquellas que mejor se acomoden a las circunstancias del proyecto y de la organización. Cualquier organización interesada en mejorar la usabilidad de sus productos software podrá elegir, basándose en la información proporcionada en el marco propuesto, las técnicas IPO que puedan encajar mejor con los objetivos organizacionales y del proyecto en cuestión.

Los casos de aplicación del marco de integración presentados en este capítulo partían de una situación en la que no se empleaban técnicas IPO y los proyectos en los que se han aplicado han sido los primeros en los que se ha tratado la usabilidad del producto. Cuando se parte de una situación en la que el campo de la IPO no está presente en el desarrollo, no consideramos que resulte aconsejable introducir un número excesivo de técnicas IPO. Por otra parte, la integración de la usabilidad en el proceso de desarrollo debe considerarse un esfuerzo de mejora del proceso gradual, de tal forma que las prácticas habituales de desarrollo vayan modificándose según se vaya aumentando el número de técnicas IPO aplicadas en sucesivos proyectos. Mientras un cambio más drástico puede ser adecuado para algunas organizaciones, no lo creemos aconsejable para la mayor parte de organizaciones de desarrollo de software con un proceso consolidado.

En cuanto a la integración con otras prácticas del desarrollo, podemos observar una diferencia notable entre la percepción de los desarrolladores con experiencia y los desarrolladores noveles. Mientras los primeros sí han visto afectadas por la inclusión de técnicas IPO otras actividades del desarrollo además de las relacionadas directamente con la parte visible de la interfaz de usuario, los segundos lo han percibido en un grado mucho menor. Podemos suponer que dicha diferencia se basa en una visión más real del desarrollo de software de los desarrolladores con experiencia, mientras que los desarrolladores noveles tiene una visión menos madura del desarrollo, con la usabilidad perteneciente a un ámbito altamente opaco relacionado con la IU. A pesar de que las técnicas que hayan empleado sí tengan relación con otros aspectos del desarrollo, como pueden ser los requisitos, no lo han percibido como tal. Los estudiantes de grado, por otra parte, en muchos casos no tienen bien conceptualizado el proceso, pues únicamente conocen un conjunto de productos que deben producir. Así, a no ser que una técnica IPO incida directamente sobre un producto y tal relación esté expresada explícitamente, pueden no descubrir tal relación. Sin embargo, los desarrolladores experimentados usan la información adquirida como alimento de otros productos, aunque tal relación no se les haya planteado explícitamente. Los desarrolladores con experiencia también valoran cualquier herramienta que permita mejorar la comunicación con el cliente y los usuarios, pudiendo haber influido así mismo este factor en la diferencia observada con los desarrolladores noveles, los cuales carecen de experiencia práctica en la negociación con clientes y usuarios.

La mejora de usabilidad del producto no ha podido verificarse con el mismo grado de profundidad que en el aspecto de proceso, puesto que la usabilidad es un atributo de calidad fuertemente dependiente del contexto y de las características específicas de los usuarios de un sistema. Se ha optado por evaluar la usabilidad percibida por usuarios representativos. Los resultados obtenidos muestran una mejora importante en la satisfacción del usuario con la usabilidad del sistema en cuestión. Por tanto, podemos considerar que, en los casos descritos,

el objetivo de mejorar la usabilidad percibida de los productos desarrollados mediante la integración de la usabilidad en el proceso de desarrollo se ha alcanzado satisfactoriamente.

Capítulo 9

Conclusiones y

Líneas Futuras

9.1 Introducción

Finalizada la exposición de la solución propuesta y de los resultados obtenidos en la aplicación de la misma, en este capítulo se valoran las conclusiones obtenidas de la realización del presente trabajo de investigación, así como las oportunidades que deja abiertas para continuar la investigación en nuevas direcciones.

9.2 Conclusiones

La usabilidad se percibe cada vez más por las organizaciones de desarrollo de software como un objetivo estratégico, debido a la ampliación de la base de usuarios de sistemas informáticos y al impacto decisivo que tiene en la satisfacción del cliente. Así, la usabilidad ha pasado en la última década de ser un atributo de calidad menor, al que se dedica menos esfuerzo que a otros atributos, a ser un objetivo especialmente relevante en el desarrollo de software.

La usabilidad no se trata adecuadamente como atributo de calidad en los procesos de desarrollo de la IS. La IPO, por el contrario, es una disciplina que dedica especial atención a la usabilidad de los sistemas informáticos y cuenta con un abanico de técnicas especialmente indicadas para tratar la usabilidad del producto software a lo largo de todo el proceso de desarrollo. La IPO cuenta con métodos propios en los que se organizan las actividades y técnicas del desarrollo que tratan la usabilidad, pero dichos métodos siguen una filosofía y enfoque distintos a los de los procesos de desarrollo habituales en la IS. La diferencia entre ambas disciplinas plantea dificultades significativas a la hora de integrar las prácticas IPO en los procesos de desarrollo de la IS. Además de las importantes diferencias terminológicas entre ambas disciplinas, los esfuerzos de integración se encuentran con una descripción de técnicas IPO enmarcada en un proceso de dicha disciplina, dificultando su aplicación fuera de dicho marco. Finalmente, la errónea identificación tradicional de la usabilidad con la actividad de diseño de la IU, acometida habitualmente en momentos tardíos del desarrollo, dificulta la integración de las técnicas IPO. El tratamiento adecuado de la usabilidad, al igual que sucede con otros atributos de calidad, requiere un esfuerzo continuado a lo largo de todo el proceso de desarrollo, y el conjunto de las técnicas IPO disponibles, consecuentemente, cubre todo el desarrollo. Estos obstáculos hacen muy difícil para una organización con el enfoque de desarrollo de la IS la integración de las técnicas IPO en su proceso de desarrollo.

Por tanto, el **problema** abordado en el presente trabajo de investigación es el de la **integración de las técnicas y actividades IPO en el proceso de desarrollo software, en una organización de construcción de software que tiene un proceso definido y que sigue mayoritariamente el enfoque de desarrollo de la IS.**

Como se ha detallado en el capítulo 2, el estudio del estado de la cuestión ha revelado que las distintas propuestas existentes en la literatura tienen importantes deficiencias que las hacen insatisfactorias como solución definitiva al problema abordado en el presente trabajo. En primer lugar, los estándares de proceso de la IS no aseguran un tratamiento adecuado de la usabilidad, aunque la última adición en este sentido, la enmienda primera al estándar 12207 de ISO, sí que requiere que se detalle la integración de las actividades centradas en el humano en el proceso general, pero sin dar indicaciones de cómo abordar tal tarea. Las propuestas de solución que suponen una extensión a métodos IS adolecen, bien de falta de iteratividad real (una condición imprescindible para un adecuado tratamiento de la usabilidad en el desarrollo), o bien están ligadas a un método concreto, limitando así su aplicabilidad general. Por otra parte, las soluciones basadas en métodos IPO adolecen principalmente de falta de detalle en la integración con el proceso de desarrollo general que engloba a las actividades de usabilidad. También adolecen de una visión del desarrollo en la que el proceso IPO alimenta al proceso general de la IS, incluyendo temas clave como los requisitos en el ámbito IPO, lo cual no es admisible desde una perspectiva de IS. El estudio del conjunto de aproximaciones consideradas ha revelado que ninguna de ellas reúne las características necesarias para poder dar una solución satisfactoria al problema planteado: Tener un carácter verdaderamente iterativo, una descripción con suficiente nivel de detalle de las actividades en las que se integran las técnicas IPO, englobar el desarrollo del sistema completo, permitir la integración con un abanico amplio de proyectos, y adherirse a un enfoque centrado en el usuario.

Con el objetivo de ofrecer una **solución** que reúna dichas características, este trabajo propone la utilización de un **marco de integración de la usabilidad en el proceso de desarrollo**. Dicho marco permite la integración de la usabilidad para distintas organizaciones sin tener que cambiar completamente las prácticas de desarrollo de la organización, con la única condición de que el desarrollo tenga un carácter iterativo. Si el proceso existente tiene dicho enfoque, reconocido actualmente en la IS como idóneo para el desarrollo de sistemas de complejidad media-alta en dominios que no se conocen en profundidad, el marco de integración permite seleccionar aquellas técnicas IPO que mejor se acomodan a las necesidades de la organización y a las características del proyecto o tipo de proyectos a desarrollar.

Las **aportaciones** principales que realiza el presente trabajo de investigación se han realizado en el proceso de construcción del marco de integración de la usabilidad propuesto, y son las siguientes:

- **Aportaciones relativas al campo de la IPO.** El campo de la IPO es muy diverso. Cada autor tiene una visión particular del proceso y de las actividades que lo forman. De igual forma, no existe un conjunto ampliamente aceptado de técnicas IPO, favoreciendo cada autor el empleo de ciertas técnicas y utilizando una denominación propia para las mismas. Para realizar el presente trabajo ha sido necesario elaborar un **estudio de las actividades IPO presentes en un proceso centrado en el usuario, y las técnicas que se recomienda aplicar en cada actividad**. En lo referente a las actividades, se ha recogido el conjunto de actividades más representativas del campo de la IPO, deduciendo la parte común entre las propuestas de los distintos autores de la disciplina. Para las técnicas ha sido necesario también realizar un estudio, para identificar los solapamientos entre las distintas denominaciones de técnicas, y para adscribir las 94 técnicas a las actividades representativas obtenidas.

- **Aportaciones relativas a la relación entre prácticas IPO y tipos de actividades de la IS.** Cada técnica IPO se ha estudiado según los rasgos que pueden resultar de interés para el ingeniero software que debe decidir su posible inclusión en el proceso de desarrollo de la organización. Así, se ha elaborado una **caracterización y selección de las técnicas IPO según criterios relevantes para la integración desde una perspectiva de IS.** Esta caracterización realizada para cada técnica de forma individual sirve de base a una priorización según el grado de utilidad para el objetivo de integración de la usabilidad en un proceso de desarrollo de la IS. Se ha dividido el conjunto de técnicas en tres grupos: Muy útiles, útiles y poco útiles para dicho objetivo. Este criterio resumen del resto de criterios nos sirve para seleccionar únicamente 35 técnicas (aquellas etiquetadas como muy útiles y útiles), con el fin de ofrecer a los ingenieros software un número más reducido de técnicas que resulte manejable, puesto que la amplitud considerable del campo de la IPO puede afectar negativamente al objetivo de facilitar la integración para organizaciones con poca experiencia previa en IPO. Una vez seleccionadas las técnicas, ha sido necesario realizar un estudio sobre el objetivo de cada actividad IPO, para establecer una correspondencia con los principales tipos de actividades que se manejan en la IS, e igualmente se ha procedido con cada técnica IPO individual seleccionada. De esta forma los ingenieros software pueden relacionar las técnicas y actividades IPO con los tipos de actividades que les son familiares. Así, el resultado del estudio ha sido la **identificación de la relación entre actividades y técnicas IPO y tipos de actividades IS.** Como consecuencia del resultado anterior también se ha producido una **identificación de los tipos de actividades del desarrollo habitual relacionadas con la usabilidad.** La relación de los tipos de actividades que están relacionadas con el adecuado tratamiento de la usabilidad en el desarrollo es especialmente relevante, porque permite mostrar a los ingenieros software que la concepción de la usabilidad como afectada únicamente por la actividad de Diseño de la IU es errónea, puesto que afecta a un abanico más amplio de actividades del desarrollo.
- **Aportaciones relativas a la influencia de la usabilidad en el proceso de desarrollo y sus etapas.** Puesto que el objetivo del presente trabajo es ofrecer un esquema suficientemente genérico como para acomodarse a los procesos de diversas organizaciones y dominios, ha sido preciso establecer cuáles son los prerequisites que cualquier proceso debe cumplir para ser candidato a la integración de la usabilidad sin sufrir modificaciones de base. Para establecer dichos prerequisites se ha realizado un estudio de las características que describen un desarrollo centrado en el usuario (el enfoque de desarrollo de la IPO, que trata adecuadamente la usabilidad), para identificar cuáles pueden ser incorporadas al proceso mediante la integración de técnicas IPO y cuáles, por el contrario, son características intrínsecas del proceso. Como resultado de este estudio **se ha identificado que un enfoque iterativo de desarrollo es el único prerequisite que debe cumplir cualquier proceso de desarrollo para ser candidato a la integración de prácticas de usabilidad.** Por otra parte, los ingenieros software que quieran integrar prácticas de usabilidad en su desarrollo, además del tipo de actividad en el que encaja cada técnica, pueden estar interesados en conocer qué técnicas son más adecuadas para una etapa concreta de su desarrollo. Puesto que ese tipo de información está descrita en la literatura IPO, se ha realizado un estudio para interpretar y elaborar esa información según un esquema común de etapas del desarrollo, en base a los hitos que son de mayor relevancia desde el punto de vista de la usabilidad. Se ha extraído para cada técnica IPO seleccionada su grado de adecuación a cada una de las tres etapas identificadas en cualquier proceso iterativo. El resultado de este estudio ha sido una **caracterización de las técnicas IPO seleccionadas según el mejor momento de aplicación en un desarrollo iterativo.**

Las aportaciones elementales se agrupan en una **aportación global** que consiste en la **elaboración de un marco de integración de la usabilidad en el proceso de desarrollo**. El marco presenta distintas vistas, con el objetivo de servir de herramienta flexible a cualquier desarrollador interesado en la integración de la usabilidad en su proceso de desarrollo. Este resultado resume el presente trabajo de investigación, incluyendo únicamente la información relevante para un profesional. El formato en el que está expresado el marco está concebido para que resulte de utilidad en la tarea de selección de las técnicas IPO mejor dotadas para las necesidades de integración de la usabilidad en un contexto concreto. El marco de integración también se ha puesto a disposición de toda la comunidad, a través de una herramienta web que facilita el acceso a la información que contiene el marco, disponible en la dirección <http://is.ls.fi.upm.es/udis/miembros/xavier/usabilityframework/>

Los resultados del presente trabajo se han aplicado en la elaboración de un sistema comercial por parte de la empresa LogicDIS, en el marco del proyecto STATUS financiado por la Comisión Europea. En la aplicación de la propuesta se ha podido comprobar su **viabilidad práctica en un entorno industrial**. También se ha aplicado la solución propuesta a la realización de un trabajo de tesis fin de master y de trabajos prácticos de una asignatura de grado de la titulación de Ingeniería Informática. Con estos **casos de estudio** se ha podido comprobar que desarrolladores con un nivel escaso de conocimientos previos acerca de la usabilidad y el campo de la IPO, han podido integrar técnicas IPO en sus prácticas de desarrollo. La integración se ha producido de manera satisfactoria para los desarrolladores, pues identifican como beneficioso el esfuerzo invertido a la vista de los resultados obtenidos. Los desarrolladores con experiencia han identificado así mismo que la usabilidad tiene relación con actividades del desarrollo que van más allá de la parte visible de la interfaz gráfica de usuario. De forma adicional, ha aumentado de forma notable la satisfacción de los usuarios de los sistemas desarrollados en cuanto a usabilidad percibida del producto.

Como se ha detallado en el capítulo 1, diferentes partes del presente trabajo de investigación han generado **publicaciones** en diversos foros de carácter científico: Resultados preliminares ([Ferré, 00] [Ferré, 01a] [Ferré, 01b]), entregables del proyecto STATUS ([Ferré, 02a] [Ferré, 02b] [Ferré, 04c] y capítulo 5 de [Folmer, 04]), y diversos aspectos de las aportaciones arriba indicadas ([Ferré, 05a] [Ferré, 05b] [Ferré, 04a] [Ferré, 04b] [Ferré, 03]).

9.3 Líneas Futuras de Investigación

Pretendemos continuar la investigación reflejada en el presente trabajo y, en concreto, las siguientes líneas parecen prometedoras:

- **Adaptación a métodos ágiles.** Los enfoques de desarrollo de tipo ágil cuentan con ciertas similitudes con algunas prácticas habituales del campo de la IPO, como son el alto grado de iteratividad, o el mayor grado de importancia dado a las personas sobre la formalidad del proceso. Puesto que el marco de integración propuesto se ha basado en las características de organizaciones de desarrollo con un proceso más definido que el empleado en los métodos ágiles, éste requiere de cambios en el enfoque para acomodarse a esta filosofía de desarrollo. Será preciso, por ejemplo, seleccionar aquellas técnicas IPO que son menos costosas de aplicar y que permiten alcanzar un mayor grado de consenso en el equipo de desarrollo, pues serían las que podrían ofrecer, en principio, una mayor utilidad a organizaciones con un enfoque ágil de desarrollo.
- **Particularizar el marco de integración a métodos o procesos de desarrollo específicos.** La solución propuesta describe un marco genérico que es preciso acomodar a la terminología propia de cada organización en cuanto a actividades y

etapas del proceso de desarrollo. Cuando la organización se ciñe a un método o proceso de desarrollo genérico, como puede ser RUP (*Rational Unified Process*) o METRICA, la terminología está establecida en la definición de dichos procesos. Así pues, la particularización del marco de integración para procesos con amplio uso puede resultar de especial interés para las organizaciones de desarrollo de software que los aplican. Dichas particularizaciones podrían constituir *plug-ins* al marco de integración que únicamente se activan si se desea acceder a la información que contiene con dicha perspectiva de desarrollo.

- **Ampliar el marco de integración de forma que resulte de interés también para empresas con experiencia en IPO.** El marco de integración presentado está dirigido a desarrolladores con una formación en IS. La terminología empleada es la de IS y la caracterización de técnicas está dirigida a una audiencia de este tipo. Sin embargo, en organizaciones donde hay una presencia importante de expertos en IPO sería de interés contar con el marco de integración como herramienta de comunicación entre la comunidad de desarrolladores y la de expertos en IPO, para una selección de técnicas IPO a integrar en el proceso de desarrollo que satisfaga a ambos grupos. Para que el marco de integración pueda cumplir esta función, será necesario enriquecerlo con una caracterización adicional de las técnicas según los parámetros más relevantes para las funciones desempeñadas por los expertos en IPO en las organizaciones de desarrollo de software.
- **Incluir los productos obtenidos de la aplicación de las técnicas IPO presentes en el marco de integración.** Para completar la información que se ofrece a los desarrolladores a través del marco, puede incluir para cada técnica IPO el/los productos que se obtienen de la aplicación de la técnica. Así, el marco estaría dotado de una vista adicional por productos generados, la cual incluiría, además de las relaciones entre técnicas y productos la posible precedencia que exista entre distintos productos. De cara a facilitar más a los desarrolladores la integración, también se puede estudiar la relación entre los productos IPO y los productos que se producen de la aplicación de técnicas IS. De esta forma, los desarrolladores podrían relacionar con mayor facilidad las técnicas IPO a incluir en su proceso de desarrollo, con los productos que manejan en sus prácticas de desarrollo habituales.

Bibliografía

- [ACM-SIGCHI, 92] ACM SIGCHI Curriculum Development Group. ACM SIGCHI Curricula for Human Computer Interaction. ACM, 1992.
- [Acuña, 01] S. T. Acuña, A. de Antonio, X. Ferré, L. Maté, M. López. "The Software Process: Modeling, Evaluation and Improvement". In Handbook of Software Engineering and Knowledge Engineering. ed. by S.K. Chang. World Scientific, New Jersey (NJ), USA, 2001. pp. 193-237.
- [Alexander, 04] D. Alexander. deyalalexander: Web Usability and User-centred Design Consulting and Resources. <http://deyalalexander.com/>
- [Anderson, 01] J. Anderson, F. Fleek, K. Garrity, F. Drake. "Integrating Usability Techniques into Software Development". IEEE Software, vol.18, no.1, January/February 2001, pp. 46-53.
- [Annett, 04] J. Annett. "Hierarchical Task Analysis". In The Handbook of Task Analysis for Human-Computer Interaction. ed. by D. Diaper, N. Stanton. Lawrence Erlbaum Associates, Mahwah (NJ), USA, 2004. Chapter 3. pp. 67-82.
- [Bevan, 01a] N. Bevan, J. Earthy. "Usability process improvement and capability assessment". Proceedings of Joint AFIHM-BCS Conference on Human-Computer Interaction IHM-HCI'2001, Volume 2. pp. 105-108.
- [Bevan, 01b] N. Bevan, I. Bogomolni, N. Ryan. "Incorporating Usability into the Development Process at Inland Revenue and Israel Aircraft Industries". Proc. of INTERACT'01. 8th IFIP TC.13 Conference on Human-Computer Interaction. Tokyo, Japan. 9-13, July, 2001. pp. 862-867.
- [Beyer, 98] H. Beyer, K. Holtzblatt. Contextual Design. Defining Customer-Centered Systems. Morgan Kaufmann, 1998.
- [Bias, 94] R. G. Bias. "The Pluralistic Walk-Through: Coordinated Empathies". In Usability Inspection Methods. ed. by J. Nielsen, R. L. Mack. Wiley, 1994.
- [BIUSEM, 95] BIUSEM Consortium. Final Report BIUSEM. Benefits of Integrating Usability and Software Engineering Methods. ESSI Project 10290, 1995.
- [Boehm, 78] B. Boehm. Characteristics of Software Quality. North Holland Publishing Co., 1978.
- [Booch, 94] G. Booch. Object-Oriented Analysis and Design with Applications, 2nd ed. Addison-Wesley, Reading, MA, 1994.
- [Card, 80] S.K. Card, T.P. Moran, A. Newell. "The Keystroke-level Model for User Performance Time with Interactive Systems". Communications of the ACM, vol. 23, no. 7. July 1980. pp 396-410.

- [Carroll, 97] J.M. Carroll. "Scenario-Based Design", in Handbook of Human-Computer Interaction. Second Edition. ed. by M. Helander, T. Landauer, P. Prabhu. Elsevier North-Holland, 1997. Chapter 17. pp. 383-406.
- [Chavez, 01] C. Chavez, L.R. Herrera. Sistema Inteligente de Gestión de Master S.I.G. Master. Tesis para la obtención del título de Master en Ingeniería del Conocimiento. Universidad Politécnica de Madrid, 2001.
- [Cockburn, 01] A. Cockburn. Writing Effective Use Cases. Addison Wesley, 2001.
- [Conallen, 03] J. Conallen. Building Web Applications with UML. Second Edition. Addison-Wesley, Boston, MA, 2003.
- [Cognetics, 03] Cognetics Corporation. The LUCID Framework™.
<http://www.cognetics.com/lucid/index.html>
- [Constantine, 99] L.L. Constantine, L.A.D. Lockwood. Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design. Addison-Wesley, New York, NY, 1999.
- [Cooper, 99] A. Cooper. The Inmates are Running the Asylum. SAMS/Macmillan, 1999.
- [Cooper, 03a] A. Cooper, R. Reimann. About Face 2.0: The Essentials of Interaction Design. Wiley Publishing, Indianapolis (IN), USA, 2003.
- [Cooper, 03b] A. Cooper. Cooper: The Origin of Personas.
http://www.cooper.com/content/insights/newsletters/2003_08/Origin_of_Personas.asp
- [Costabile, 01] M.F. Costabile. "Usability in the Software Life Cycle". In Handbook of Software Engineering and Knowledge Engineering. ed. by S. K. Chang. World Scientific, New Jersey (NJ), USA, 2001. 179-192.
- [Coutaz, 94] J. Coutaz. "Evaluation Techniques: Exploring the Intersection of HCI and Software Engineering". Proc. of the Workshop on Software Engineering and HCI: Joint Research Issues at ICSE'94. Sorrento, Italy. May 16-17, 1994. pp. 35-48.
- [Derniame, 99] J.C. Derniame, B.A. Kaba, D. Wastell. "Software Process: Principles, Methodology and Technology". Lecture Notes in Computer Science 1500. Springer, 1999.
- [DiamondBullet, 04] Diamond Bullet. Usability First. <http://www.usabilityfirst.com/>
- [Dumas, 99] J. S. Dumas, J. C. Redish. A Practical Guide to Usability Testing. Revised Edition. Intellect, Exeter, England, 1999.
- [Ferré, 00] X. Ferré. "Principios Básicos de Usabilidad para Ingenieros Software" V Jornadas de Ingeniería del Software y Bases de Datos 2000. Valladolid (España). 8-10 noviembre 2000.
- [Ferré, 01a] X. Ferré, N. Juristo, H. Windl, L. Constantine "Usability Basics for Software Developers". IEEE Software, vol 18, no. 1 (January/February 2001) pp. 22-29.
- [Ferré, 01b] X. Ferré. "Incorporating Usability into an Object Oriented Development Process" Workshop Usability Throughout the Entire Systems Development Lifecycle, INTERACT'01. Tokyo, Japan. 9-10 July, 2001.

- [Ferré, 02a] X. Ferre, N. Juristo, A. M. Moreno. STATUS Project. Deliverable D.5.1. Selection of the Software Process and the Usability Techniques for Consideration. 2002. <http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Ferré, 02b] X. Ferre, N. Juristo, A. M. Moreno. STATUS Project. Deliverable D.5.2. Specification of the Software Process with Integrated Usability Techniques. 2002. <http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Ferré, 03] X. Ferre. "Integration of Usability Techniques into the Software Development Process". Workshop Bridging the Gaps Between Software Engineering and Human-Computer Interaction, ICSE-2003. Portland (OR), USA. 1-2 May, 2003.
- [Ferré, 04a] X. Ferré, A. M. Moreno. "Integración de la IPO en el Proceso de Desarrollo de la Ingeniería del Software: Propuestas Existentes y Temas a Resolver" V Congreso Interacción Persona-Ordenador (Interacción 2004). Lleida, España. 3-7 Mayo, 2004. pp. 134-141.
- [Ferré, 04b] X. Ferre, N. Juristo, A. M. Moreno. "Improving Software Engineering Practice with HCI Aspects". First International Conference on Software Engineering Research and Applications, SERA 2003, San Francisco (CA), USA, June 25-27, 2003, Selected Revised Papers - Lecture Notes in Computer Science. Vol 3026. Springer, 2004. pp. 349-363.
- [Ferré, 04c] X. Ferre, N. Juristo, A.M. Moreno. STATUS Project. Deliverable D.6.6 Final Results on the Integrated Software Process. 2004. <http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Ferré, 05a] X. Ferre, N. Juristo, A.M. Moreno. "Which, When and How Usability Techniques and Activities Should Be Integrated". In Human-Centered Software Engineering - Integrating Usability in the Development Process. ed. by A. Seffah, J. Gulliksen, M.D. Desmarais. Kluwer/Springer, 2005. (en prensa)
- [Ferré, 05b] X. Ferre, N. Juristo, A.M. Moreno. "Integration of HCI Practices into Software Engineering Development Processes: Pending Issues". In Encyclopedia of Human Computer Interaction. ed. by C. Ghaoui. Idea Group Reference, 2005. (en prensa)
- [Ferré, 05c] X. Ferre, N. Juristo, A. M. Moreno. "Framework for Integrating Usability Practices into the Software Process" 6th Int. Conf. on Product Focused Process Improvement (PROFES 2005). Oulu (Finland), June 13-15, 2005. (aceptado)
- [Folmer, 04] E. Folmer, R. Chatley, X. Ferré, N. Juristo, J. Magee, A.M. Moreno, J. Kramer. STATUS Project. Deliverable D.1.7 Main Achievements and Findings of the Project. 2004. <http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Flückiger, 03] M. Flückiger. "Requirements Engineering with Contextual Design and RUP". Proceedings of INTERACT'03. Zürich, Switzerland, September 1-5, 2003. pp. 761-764.
- [Fuggetta, 00] A. Fuggetta. "Software process: A roadmap" in The Future of Software Engineering, ed. by A. Finkelstein. ACM Press, 2000. pp. 27-34.
- [Glass, 03] R.L. Glass. Facts and Fallacies of Software Engineering. Addison-Wesley, Boston (MA), USA, 2003.
- [Gould, 88] J.D. Gould. "How to Design Usable Systems" in Handbook of Human Computer Interaction, ed. by M. Helander. Elsevier, 1988.

- [Gulliksen, 99] J. Gulliksen. "Bringing in the Social Perspective: User Centred Design". Proc. of HCI-International'99. pp. 1327-1331.
- [Gulliksen, 01] J. Gulliksen, B. Göransson. "Reengineering the Systems Development Process for User-Centred Design". Proc. of Human-Computer Interaction (INTERACT'01). pp. 359-366. Tokyo (Japan)2001.
- [Hackos, 98] J.T. Hackos, J.C. Redish. User and Task Analysis for Interface Design. Wiley, 1998.
- [Hakiel, 97] S. Hakiel. "Delivering Ease of Use". Computing and Control Engineering Journal. Vol. 8, n. 2, April 1997. pp. 81-87.
- [Hix, 93] D. Hix, H.R. Hartson. Developing User Interfaces: Ensuring Usability Through Product and Process. John Wiley and Sons, New York (NY), USA, 1993.
- [Holzinger, 05] A. Holzinger. "Usability Engineering Methods for Software Developers". Communications of the ACM. Vol. 48, no. 1. January, 2005. pp. 71-75.
- [Humphrey, 89] W. Humphrey. Managing the software process. Addison-Wesley, 1989.
- [IEEEGlossary, 90] IEEE. IEEE Std 610.12-1990. IEEE Standard Glossary of Software Engineering Terminology. IEEE, New York (NY), USA, 1990.
- [IEEE1074, 98] IEEE. IEEE Std 1074-1997. IEEE Standard 1074 for Developing Software Life Cycle Processes. IEEE, 1998.
- [IEEE12207.0, 98] IEEE/EIA. IEEE/EIA Standard 12207.0-1996. Industry Implementation of International Standard ISO/IEC 12207:1995. (ISO/IEC 12207) Standard for Information Technology - Software Life Cycle Processes. IEEE, 1998.
- [IEEESoftware, 01] IEEE Computer Society. IEEE Software issue dedicated to Usability Engineering. vol.18, no.1, January/February 2001.
- [ISO12207, 95] ISO/IEC. ISO/IEC International Standard: Information Technology. Software Life Cycle Processes, ISO/IEC Standard 12207:1995. ISO, 1995.
- [ISO12207_1, 02] ISO/IEC. ISO/IEC International Standard: Information Technology. Software Life Cycle Processes. Amendment 1. ISO/IEC 12207:1995/Amd. 1:2002. ISO/IEC, 2002.
- [ISO13407, 99] ISO. ISO 13407. Human-Centred Design Processes for Interactive Systems. ISO, Geneva (Switzerland), 1999.
- [ISO18529, 00] ISO. Technical Report ISO/TR 18529. Ergonomics - Ergonomics of Human-System Interaction - Human-Centred Lifecycle Process Descriptions. ISO, Geneva (Switzerland), 2000.
- [ISO/IEC15504, 98] ISO/IEC. Technical Report ISO/IEC TR 15504. Information Technology - Software Process Assessment. ISO/IEC, Geneva (Switzerland), 1998.
- [Jacobson, 93] I. Jacobson, M. Christerson, P. Jonsson, G. Övergaard. Object-Oriented Software Engineering. A Use-Case Driven Approach. Revised Printing. ACM Press - Addison-Wesley, 1993.

- [Jacobson, 99] I. Jacobson, G. Booch, J. Rumbaugh. The Unified Software Development Process. Addison Wesley, Reading (MA), USA, 1999.
- [Jokela, 02] T. Jokela. "A Method-Independent Process Model of User-Centred Design". Usability: Gaining a Competitive Edge, IFIP 17th World Computer Congress. TC13 Stream on Usability. Kluwer, 2002. pp. 23-38.
- [Jung, 55] C.G. Jung. El Yo y el Inconsciente. Luis Miracle, 1995.
- [Juristo, 04] N. Juristo, N., A.M. Moreno, D. Tsirikos. STATUS Project. Deliverable D.6.3. Analysis and Comparison of Usability of Old and New Developments. 2004. <http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Kawalek, 96] P. Kawalek, D.G. Wastell. "Organisational design for software development: a cybernetic perspective" In Lecture Notes in Computer Science. Proc. of the 5th European Workshop on Software Process Technology (EWSPT'96). Springer-Verlag, 1996. pp. 258-270.
- [Kreitzberg, 96] C.B. Kreitzberg. "Managing for Usability: The Cognetics QUE Design Methodology". In Multimedia a Management Perspective. ed. by A.F. Alber, Wadsworth, Belmont (CA), USA, 1996. pp. 65-88.
- [Kroll, 03] P. Kroll, P. Kruchten. The Rational Unified Process Made Easy. A Practitioner's Guide to the RUP. Addison-Wesley, Boston (MA), USA, 2003.
- [Larman, 02] C. Larman. UML y Patrones, Segunda Edición. Prentice-Hall, 2002.
- [Larman, 04] C. Larman. Agile and Iterative Development. A Manager's Guide. Addison-Wesley, Boston (MA), USA, 2004.
- [Lewis, 97] C. Lewis, C. Wharton. "Cognitive Walkthroughs", in Handbook of Human-Computer Interaction. Second Edition, ed. by M. Helander, T. Landauer, P. Prabhu. Elsevier North-Holland, 1997. Chapter 30. pp. 717-732.
- [Lim, 94] K.Y. Lim, J. Long. The MUSE Method for Usability Engineering. Cambridge University Press, Glasgow, UK, 1994.
- [Lombardi, 04] V. Lombardi. NBS: The Real Origin of Personas. <http://www.noisebetweenstations.com/personal/weblogs/tinderbox/design/process/personas/therealo.shtml>
- [Lorés, 01] J. Lorés, T. Granollers, S. Lana. Introducción a la Interacción Persona-Ordenador. en La Interacción Persona-Ordenador. ed. por J. Lorés. AIPO, 2001.
- [McCall, 77] J.A. McCall, P.K. Richards, G.F. Walters. "Factors in Software Quality," vol. 1, 2, and 3, AD/A-049-014/015/055, National Tech. Information Service, 1977.
- [Mcmullin, 03] J. McMullin. The origin of personas. <http://iaslash.org/node/view/7455>
- [Mayhew, 99] D.J. Mayhew. The Usability Engineering Lifecycle. Morgan Kaufmann, San Francisco (CA), USA, 1999.
- [Metzker, 02] E. Metzker, H. Reiterer. "Use and Reuse of HCI Knowledge in the Software Development Lifecycle. Existing Approaches and what Developers Think". Proc. of "IFIP 17th World Computer Congress. TC13 Stream on Usability: Gaining a Competitive Edge".

Montreal, Quebec, Canada. 25-30 August 2002. Kluwer Academic Publishers, Norwell (MA), USA, 2002. pp. 39-55.

[MSN, 03] MSN - Microsoft Corporation. MSN Personas (MSN Advertising)
<http://advertising.msn.com/home/MSNPersonas.asp>

[Nielsen, 92] J. Nielsen. "The Usability Engineering Life Cycle". IEEE Computer. Vol. 25, no. 3. March 1992. pp. 12-22.

[Nielsen, 93] J. Nielsen. Usability Engineering. AP Professional. Boston (MA), USA, 1993.

[Nielsen, 94] J. Nielsen, R.L. Mack. Usability Inspection Methods. Wiley, 1994.

[Norman, 90] D.A. Norman, The Design of Everyday Things. Doubleday, 1990.

[Nunes, 01] D.N.J. Nunes. Object Modeling for User-Centered Development and User Interface Design: The Wisdom Approach. Ph.D. Thesis. April 2001.
<http://dme2.uma.pt/njn/tiki-index.php?page=PhD>

[Preece, 94] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, T. Carey. Human-Computer Interaction. Addison Wesley, Harlow, England, 1994.

[Rational, 02] Rational Software Corp. What's New in Rational Development Accelerators™ Version 2002. 2002. http://www.rational.com/rda/wn_2002.jsp?SMSESSION=NO

[Robertson, 01] J. Robertson. Information Design Using Card Sorting. Step Two Designs Pty Ltd, 2001. <http://www.steptwo.com.au/papers/cardsorting/>

[Rumbaugh, 91] J. Rumbaugh, M.R. Braha, W. Lorensen, F. Eddy, W. Premerlani. Object-Oriented Modeling and Design. Upper Saddle River, NJ, USA, Prentice-Hall, 1991.

[SAP, 04] SAP. SAP Design Guild. <http://www.sapdesignguild.org/>

[Seffah, 03] A. Seffah, A. Andreevskaja. "Empowering Software Engineers in Human-Centered Design". Proc. of the ICSE'03 Conference. Portland, Oregon (USA). 3-10 May, 2003. pp. 653-658.

[Seffah, 04] A. Seffah, E. Metzker. "The Obstacles and Myths of Usability and Software Engineering". Communications of the ACM. December 2004 Vol. 47, no. 12. pp. 71-76.

[Shneiderman, 98] B. Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison-Wesley, Reading (MA), USA, 1998.

[Snyder, 03] C. Snyder. Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces. Morgan-Kaufmann, 2003.

[Soares, 03a] K. Soares, E. Furtado. "RUPi - A Unified Process that Integrates Human-Computer Interaction and Software Engineering". Proc. of the Workshop "Bridging the Gap Between Software-Engineering and Human-Computer Interaction" at ICSE'03. Portland, OR. May 3-11, 2003. pp. 41-48.

[Soares, 03b] K. Soares, E. Furtado. "Integration of Human-Computer Interaction in a Software Development Process". Proc. of HCI International 2003. 22-27 June, 2003. Crete, Greece. pp. 1539-1543.

- [STATUS, 01] STATUS partners. STATUS (Software Architecture that supports Usability) Project Technical Annex. 2001.
- [SWEBOK, 01] IEEE Software Engineering Coordinating Committee. "Guide to the Software Engineering Body of Knowledge - Trial Version 1.00". IEEE Computer Society, Los Alamitos, (CA), USA, 2001.
- [SWEBOK, 04] IEEE Computer Society Professional Practices Committee. "Guide to the Software Engineering Body of Knowledge - 2004 Version". IEEE Computer Society, Los Alamitos (CA), USA, 2004.
- [Tsirikos, 04] D. Tsirikos. "STATUS Project. Deliverable D6.2 Development Document of Pilot Project by LogicDIS SA". 2004.
<http://www.ls.fi.upm.es/status/results/deliverables.html>
- [Vredenburg, 02] K. Vredenburg, J. Mao, P.W. Smith, T. Carey. "A Survey of User-Centered Design Practice". Proc. of CHI-2002. Minneapolis (MN), USA, 20-25 April 2002. pp. 471-478.
- [Wasserman, 85] A.I. Wasserman. "Extending State Transition Diagrams for the Specification of Human-Computer Interaction". IEEE Transactions on Software Engineering. Vol. 11, no.8. August 1985. pp. 699-713.
- [Wixon, 96] D. Wixon, J. Ramey (editors). Field Methods Casebook for Software Design. John Wiley and Sons, 1996.
- [Wixon, 97] D. Wixon, C. Wilson. "The Usability Engineering Framework for Product Design and Evaluation". In Handbook of Human-Computer Interaction. ed. by M. Helander, T. Landauer, P. Prabhu. Elsevier North-Holland, 1997. pp. 653-688.

Anexo A

Técnicas de Usabilidad

En este anexo se detallan las técnicas recogidas en cada una de las fuentes consideradas para el estudio de técnicas de usabilidad.

[Hix, 93]

Hix y Hartson se centran principalmente en dos temas en este trabajo: La notación UAN (*User Action Notation*), y el establecimiento de especificaciones de usabilidad para ser evaluadas en test de usabilidad. El resto de técnicas mencionadas en este trabajo no se explican con el mismo grado de detalle.

Las técnicas propuestas por Hix y Hartson son las siguientes:

- **Perfiles de Usuario:** Implica la definición de clases representativas de usuarios en términos de las tareas que realizarán y las habilidades y conocimientos que dichos usuarios aportan a las tareas.
- **Escenarios y Representaciones de Pantallas:** Los escenarios son representaciones concretas de objetos de interacción que permiten visualizar el diseño, jugar con ideas y probar posibilidades. El escenario está formado por representaciones de pantallas. Según los autores, un escenario es algo cercano a un prototipo, puesto que un prototipo se puede ver como, simplemente, un conjunto de escenarios más completo y ejecutable.
- **UAN (*User Action Notation*):** UAN es una técnica para representar diseño de interacción con el usuario desde un punto de vista comportamental. UAN aborda el acto mental creativo que supone la resolución de problemas (por ejemplo, al crear nuevos diseños), y el acto físico de documentar una representación del diseño. La abstracción primaria en UAN es la tarea del usuario. Una interfaz se representa como una estructura casi-jerárquica de tareas asíncronas, siendo independiente el secuenciamiento dentro de cada tarea del de otras tareas. Las acciones de los usuarios, la correspondiente reacción de la interfaz, y la información de cambio de estado son representadas al nivel más detallado.
- **Diagramas de Transición de Estados de la Interfaz:** En este tipo de diagramas los nodos representan estados de la interfaz o pantallas, y los arcos representan transiciones de estado basadas en las entradas. Estos diagramas pueden usarse para

complementar las representaciones UAN, indicando secuenciamiento e información de estado o modo.

- **Especificaciones de Usabilidad:** Son objetivos de usabilidad cuantitativos, que se utilizan como guía para conocer cuándo una interfaz es suficientemente buena. Pueden basarse en medidas objetivas o subjetivas. Las **Medidas Objetivas** se asocian normalmente con una tarea concreta de referencia, mientras que las **Medidas Subjetivas** se asocian habitualmente con un cuestionario para usuarios.
- **Prototipado:** Implica la producción de una versión temprana del sistema que ilustra funcionalidades esenciales del posterior sistema operacional.
- **Evaluación Heurística:** Esta técnica implica revisar una IU respecto a la usabilidad de la interacción entre el usuario y el sistema.
- **Técnicas Cualitativas de Generación de Datos:** Estas técnicas se llevan a cabo en test de usabilidad, y son muy importantes cuando el objetivo es la realización de evaluación formativa. Son las siguientes:
 - **Toma del Protocolo Verbal Concurrente:** Consiste en pedir al participante en el test que verbalice su razonamiento al utilizar el sistema.
 - **Toma Retrospectiva del Protocolo Verbal:** El evaluador permite a los participantes trabajar sin interrupciones durante una sesión que se graba. A continuación, inmediatamente tras la sesión, el evaluador y cada participante revisan la grabación de vídeo juntos, y el evaluador pide al participante que analice qué ocurrió en la sesión.
 - **Toma de Incidentes Críticos:** Esta técnica consiste en registrar tanto los incidentes negativos (signos de frustración, bien a través de comentarios, bien con acciones), como los incidentes positivos (expresiones de satisfacción y terminación). Los incidentes negativos ayudan a identificar los principales problemas de usabilidad, mientras que los positivos ayudan a identificar metáforas o detalles a utilizar más a fondo en la IU debido a su éxito.
 - **Entrevistas Estructuradas:** En forma de entrevista post-experimento, el evaluador pregunta a cada participante una serie de cuestiones pre-planificadas.
- **Técnicas de Recogida de Datos:** Se trata de técnicas para recogida de datos en test de usabilidad o cualquier tipo de observación de usuarios. Incluyen las siguientes técnicas:
 - **Toma de Notas en Tiempo Real:** La técnica más básica consiste en que el evaluador se prepara para tomar numerosas notas según se suceden las actividades de una sesión. Apenas se puede considerar una técnica propiamente dicha.
 - **Grabación de Vídeo.**
 - **Grabación de Audio.**
 - **Instrumentación Interna de la Interfaz:** Esta técnica consiste en la programación de registros internos de las acciones que se realizan sobre la interfaz.

- **Análisis Coste/Importancia:** Esta técnica implica considerar la importancia relativa de los problemas de usabilidad identificados en la evaluación de usabilidad, junto con el coste de desarrollo de las posibles soluciones.
- **Test en Laboratorio:** En este tipo de test se trae al participante a la interfaz. Esto es, los participantes son llevados a unas instalaciones tipo laboratorio de usabilidad donde llevan a cabo las tareas de referencia y todo lo asociado a un test de usabilidad.
- **Test de Campo:** De forma alternativa a los test en laboratorio, la interfaz se lleva hasta el participante. Esto es, la versión actual del sistema en desarrollo se instala en el entorno normal de trabajo previsto de los usuarios del sistema. Normalmente se suele plantear este tipo de test de usabilidad en etapas avanzadas del desarrollo, y cuando se desean obtener datos en un plazo más largo.

[Nielsen, 93]

Este trabajo de Nielsen tiene como objetivo ser un libro fácilmente legible y que haga reflexionar, pero no profundiza en la descripción de las técnicas que se mencionan. Las técnicas que se presentan no están organizadas de una manera ordenada, incluso para muchas de ellas el autor no presenta un nombre y definición apropiados, confundiéndose fácilmente con la actividad en la que se aplicarían. Si embargo, con el fin de poder comparar su propuesta con la de otros autores, hemos extraído las técnicas mencionadas¹⁷ en el texto. Las técnicas propuestas por Nielsen son las siguientes:

- **Características de Usuarios Individuales:** Esta técnica trata de obtener acceso a usuarios representativos para reunir información acerca de su experiencia de trabajo, nivel educativo, edad, experiencia previa con ordenadores, etc. También es preciso conocer su contexto de trabajo y social.
- **Análisis Competitivo:** Es deseable analizar productos existentes de forma heurística, de acuerdo a guías de usabilidad establecidas, y llevar a cabo test empíricos con usuarios con estos productos.
- **Objetivos de Usabilidad:** Para cada atributo de usabilidad relevante, se especifican varios distintos niveles de rendimiento como parte del proceso de establecimiento de objetivos de usabilidad. Una posible notación para estos niveles es la **línea de objetivos de usabilidad**.
- **Análisis del Impacto Financiero:** Es un análisis del impacto financiero que puede tener la usabilidad del sistema. Es más sencillo llevarlo a cabo para desarrollos internos o para desarrollos bajo contrato directo de la organización usuaria. Implica calcular cómo el grado de mejora de usabilidad se traduce en ahorro en el tiempo del usuario (tiempo de empleado que cuesta dinero a la organización usuaria), debido a la mejora en rendimiento del usuario. El tiempo salvado por una facilidad de aprendizaje mejorada también puede calcularse.
- **Diseño Paralelo:** Varios diseñadores distintos trabajan en diseños preliminares (trabajando independientemente). Una variantes se denomina Diseño Paralelo

¹⁷ Mencionadas con cierto grado de detalle. El suficiente como para permitir una aplicación de la técnica grosso modo.

Diversificado, en el que se pide a cada diseñador que se centre en diferentes aspectos del problema de diseño.

- **Evaluación Heurística:** Se lleva a cabo observando una interfaz e intentando obtener una opinión acerca de lo bueno y malo de la interfaz. Es mejor tener a varios evaluadores que revisen el mismo diseño de forma independiente, puesto que descubren muchos más errores que un único evaluador. Lo ideal es lo realicen especialistas en usabilidad. Un tipo de evaluación heurística particular es el recorrido pluralístico:
 - **Recorrido Pluralístico:** La Evaluación Heurística se lleva a cabo por usuarios representativos, desarrolladores del producto y especialistas en usabilidad.
- **Prototipado:** Los prototipos son versiones reducidas del sistema completo, bien por recortar el número de funcionalidades del prototipo, bien por reducir el nivel de funcionalidad de las opciones que parecen funcionar pero no hacen nada en realidad.
 - **Escenarios:** Son prototipos que están limitados en dos sentidos: Por una parte reducen el número de funciones que ofrece el sistema, y por otra el usuario no puede interactuar usando datos reales. Representan una sesión de interacción única, por lo que sólo simulan la IU mientras el usuario siga un camino previamente definido. Se trata de un tipo especialmente poco costoso de prototipos, y pueden ser tanto prototipos de papel como prototipos ejecutables elaborados por una herramienta de prototipado.
- **Card Sorting:** Esta técnica permite comprender la representación de información que manejan los usuarios. Consiste en pedir a los usuarios que agrupen una serie de conceptos del dominio, para obtener como resultado una agrupación representativa del modelo del dominio que tiene el usuario en la cabeza. Cada concepto se escribe en una tarjeta, y se pide al usuario que organice las tarjetas en pilas.
- **Test de Usabilidad:** Un test de usabilidad implica probar la versión actual del sistema con usuarios reales. Un test de usabilidad normalmente tiene cuatro etapas: Preparación, introducción, el propio test, y elaboración del informe del test.
- **Análisis de Impacto:** Implica en primer lugar la identificación de problemas de usabilidad, y a continuación volver a las grabaciones realizadas con el fin de investigar exactamente cuántos usuarios han tenido cada problema de usabilidad y cuánto les retrasó cada problema. Pueden usarse para priorizar el arreglo de los problemas de usabilidad en un esfuerzo de rediseño.
- **Pensar en Voz Alta:** Nielsen distingue "pensar en voz alta" de otras técnicas de test de usabilidad indicando que puede ser el método de ingeniería de usabilidad más valioso considerado por sí mismo. Un test con pensar en voz alta implica tener a un sujeto usando el sistema mientras, de forma continua, dice en voz alta lo que está pensando. Su punto fuerte está en los datos cualitativos que se obtienen en vez de en medidas de rendimiento. La idea es obtener la impresión del usuario mientras usa el sistema para evitar toda posible racionalización posterior de sus acciones. Existen varias aproximaciones a la técnica de pensar en voz alta:
 - **Interacción Constructiva:** Implica tener a dos usuarios de test utilizando el sistema juntos. Se denomina también Aprendizaje de Codescubrimiento. Se basa en el hecho de que las personas acostumbran a verbalizar cuando están intentando resolver un problema de forma conjunta.

- **Test Retrospectivo:** La sesión de test de usabilidad se graba en vídeo y se pide al usuario que vea la grabación. Los comentarios de un usuario cuando vea la grabación son en ocasiones más extensos que los que realiza cuando está llevando a cabo la tarea del test. El revisor puede parar la grabación y preguntar al usuario en cualquier momento, sin miedo a interferir con el test, que esencialmente ha sido ya completado.
- **Método de Entrenamiento:** El experimentador (o "entrenador") lleva al usuario en la dirección adecuada cuando éste está usando el sistema. El usuario puede preguntar al experimentador, y las preguntas pueden mostrar problemas de usabilidad que permanecerían ocultos de otro modo. El experimentador contestará en base a su conocimiento del sistema.
- **Laboratorios de Usabilidad:** Se trata de instalaciones especialmente dedicadas a test de usabilidad. Suelen tener espejos de una sola vía y cámaras de vídeo.
- **Grabación en vídeo:** Los test de usabilidad se pueden grabar en vídeo para posterior análisis. Un uso de los vídeos está en relación con la técnica de Análisis de Impacto, para, una vez identificado cada problema de usabilidad, volver a los vídeos a comprobar cuántos usuarios han tenido ese problema y cuánto les retardó en su tarea. También sirven las grabaciones en vídeo como medio de comunicación de la importancia de los problemas de usabilidad, entre expertos en usabilidad y los desarrolladores y la dirección.
- **Observación:** La observación implica visitar a uno o más usuarios y tomar notas y tal vez grabar en vídeo las actividades del usuario. De todas formas, el observador no debería interferir con el trabajo del usuario, de tal forma que todas las tareas del observador deben realizarse de forma no obtrusiva.
- **Cuestionarios y Entrevistas:** Se trata de métodos indirectos de estudio de la IU, porque proporcionan al equipo de desarrollo las opiniones del usuario, pero no información directa de la IU. Son especialmente apropiadas para obtener la satisfacción subjetiva del usuario. Los cuestionarios pueden ser distribuidos por correo, correo electrónico o directamente con el software. Las entrevistas se pueden llevar a cabo en persona o mediante conversación telefónica.
- **Focus Groups:** En un *focus group*, se reúne a un grupo de entre seis y nueve usuarios para discutir nuevos conceptos e identificar temas relevantes en un período de unas dos horas. Cada grupo es llevado por un moderador que es responsable de mantener el enfoque del grupo en cualesquiera que sean los temas de interés.
- **Registro del Uso Real:** Registrar el uso implica tener al ordenador recogiendo automáticamente estadísticas acerca del uso detallado del sistema. Normalmente es una forma de conseguir información acerca del uso de campo de un sistema tras su lanzamiento, pero puede utilizarse como un método suplementario en test de usabilidad.
- **Retroalimentación del Usuario:** La retroalimentación puede recogerse dando a los usuarios acceso a direcciones específicas de correo, grupos de noticias, o tableros de anuncios electrónicos. Los usuarios pueden enviar sus quejas y peticiones de cambio o mejora.
- **GOMS:** Este método implica listar posibles metas y submetas del usuario, los operadores que los usuarios tienen disponibles, los métodos que los usuarios componen de secuencias de dichos operadores para alcanzar las metas, y las reglas de

selección necesarias para decidir qué es lo siguiente que se hace si se tienen varias metas pendientes, o si hay varios métodos para alcanzar una misma meta.

[Preece, 94]

Preece et al. ofrecen una buena visión general del campo de la IPO en la fecha de publicación. Dan detalles sobre un alto número de técnicas, las cuales se detallan a continuación:

- **Especificación de Usabilidad:** Se trata de la misma técnica que las Especificaciones de Usabilidad propuestas por Hix y Hartson.
- **HTA (*Hierarchical Task Analysis*):** Se trata de un método clásico de análisis de tareas, según los autores es uno de los más conocidos. Implica un proceso iterativo de identificación, categorización y descomposición de tareas en subtareas, junto con la comprobación de la precisión de tal descomposición. Se divide en tres etapas: Inicio, progreso y finalización. Para la representación gráfica de la descomposición utiliza los diagramas de estructuras. Estos diagramas muestran la secuencia de actividades ordenándolas de izquierda a derecha; las actividades que se pueden repetir se marcan con un asterisco dentro de la caja, y cuando hay una elección entre un número de actividades éstas se marcan con un pequeño círculo dentro de la caja.
- **Familia de Modelos GOMS (*Goals, Operations, Methods and Selection Rules*):** El objetivo de GOMS es la representación de los métodos o planes e elabora una persona para alcanzar metas específicas. Los métodos se componen de una serie de pasos, formados por operaciones que el usuario realiza. Para cuando hay más de un método disponible para alcanzar una meta o submeta, el modelo GOMS incluye reglas de selección que describen cómo se elige el método apropiado dependiendo del contexto. Existe una familia de modelos GOMS, de los cuales los autores ponen como ejemplo la notación NGOMSL.
 - **NGOMSL:** Según los autores se trata de una manera más "natural" de expresar un modelo tipo GOMS.
- **Tormenta de Ideas Visual:** Se trata de una técnica de confección de bosquejos empleada para explorar diseños alternativos. Tras crear bosquejos iniciales, las mejores ideas pueden ser desarrolladas en mayor profundidad con la construcción de cartulinas representando el diseño, las cuales pueden ser evaluadas con usuarios. A continuación pueden desarrollarse escenarios, software o prototipos de vídeo.
- **Escenarios, *Storyboards* e Instantáneas:** Un escenario es una historia de ficción, personalizada con personajes, eventos, productos y entornos. Las instantáneas son imágenes visuales individuales (normalmente tipo cómic), que capturan una posible acción significativa. Los *storyboards* son secuencias de instantáneas que se centran en las principales acciones en una posible situación.
- **Técnicas de Lógica de Diseño:** Hay varias técnicas para capturar las decisiones de diseño de tal forma que puedan ser revisadas en futuras reuniones:
 - **IBIS (*Issue-Based Information Systems*):** IBIS intenta capturar las decisiones de diseño según progresa el mismo. La actividad central de IBIS es la deliberación, esto es, considerar los pros y los contras de soluciones alternativas a los problemas planteados. La notación utilizada consiste en un

diagrama gráfico llamado un mapa de cuestiones, el cual ilustra las cuestiones y sus relaciones.

- **PHI (*Procedural Hierarchy of Issues*):** PHI es una versión hipertextual de IBIS que aborda los principales temas identificados en IBIS. Las nociones de cuestión y resolución de cuestiones se extienden en PHI, mientras se restringe el orden en el que las cuestiones se identifican.
- **Análisis de Espacios de Diseño:** El diseño se ve en esta técnica como una exploración de un espacio de alternativas. Se anima activamente al diseñador a explorar diseños alternativos. Por tanto, el resultado final no se trata únicamente de una descripción detallada de los "porqués" y "por consiguiente" de un diseño, sino también de obtener diseños de mejor cualidad, puesto que el diseñador habrá explorado muchas más alternativas.
- **Análisis de Afirmaciones:** Se basa en los supuestos psicológicos que el diseñador hace o ha hecho con respecto al uso del sistema, el usuario del sistema, el entorno de uso, etc. Una afirmación relaciona algún aspecto del diseño del sistema con una consecuencia de importancia para el usuario. El análisis de afirmaciones se lleva a cabo creando escenarios de uso del sistema y analizándolos en busca de afirmaciones o supuestos. Deberían incluirse las tareas principales que el sistema va a soportar y los errores clave que el sistema debe ser capaz de manejar.
- **Técnicas de Prototipado:** Hay varios aspectos del sistema en los que el prototipo puede centrarse:
 - **Animación de Requisitos:** Posibles requisitos se muestran en un prototipo, el cual se construye con paquetes específicos para pintar pantallas y añadirles animación para ilustrar los aspectos funcionales.
 - **Prototipado de Baja Fidelidad:** Este tipo de prototipado implica el uso de materiales distintos a los de la versión final, y que tienden a ser más baratos y más rápidos de desarrollar.
 - **Prototipos Guiados:** El usuario observa mientras otra persona, normalmente un miembro del equipo de desarrollo, "conduce" el sistema.
 - **Prototipos Mago de Oz:** El usuario interactúa con una pantalla, pero en vez de un software respondiendo a las acciones del usuario, un desarrollador está situado en otra pantalla respondiendo a las peticiones del usuario. El usuario desconoce el hecho de que las respuestas las da una persona en vez de un sistema software.
- **Observación Directa:** Usuarios individuales pueden ser observados directamente realizando tareas especialmente preparadas o haciendo su trabajo habitual, con el observador anotando su comportamiento o registrando su rendimiento de alguna forma, como por ejemplo tomando el tiempo empleado en ciertas secuencias de acciones.
- **Observación Indirecta - Grabación de Vídeo:** La grabación en vídeo ofrece una alternativa a la observación directa, que se prefiere porque provee con un registro permanente al cual se puede volver cuantas veces sea necesario. El análisis de datos de vídeo puede resultar una tarea costosa en tiempo. Se cita a menudo un ratio de 5:1,

esto es, una hora de vídeo puede requerir cinco horas, un día, o incluso más para ser analizada.

- **Protocolo Verbal:** El protocolo verbal se basa en alguna forma de grabación de audio en una sesión de observación. Añade una dimensión extra a la información reunida al abordar la actividad cognitiva subyacente en el comportamiento físico del usuario. Existen las siguientes técnicas particulares del protocolo verbal:
 - **Protocolo de Pensar en Voz Alta:** Es la misma técnica enunciada por Nielsen.
 - **Protocolo Post-Evento:** Con esta técnica, los usuarios revisan vídeos de sus acciones y proporcionan comentarios acerca de lo que estaban intentando hacer. Nótese que los protocolos post-evento puede contener información recordada que no se utilizó durante la secuencia de tareas: La retrospectiva puede provocar una racionalización de las acciones propias del usuario.
- **Registro Software:** Cuenta con dos ventajas principales: no requiere que el investigador esté presente, y no es obtrusivo. Hay dos tipos de registro, que pueden combinarse entre sí y con la grabación de vídeo. Son los siguientes:
 - **Registro de Pulsaciones en el Tiempo:** Un registro de cada tecla pulsada se almacena, junto con el momento exacto en el que ha ocurrido el evento.
 - **Registro de la Interacción:** La interacción completa se registra, de tal forma que puede reproducirse completa en tiempo real.
- **Entrevistas:** Hay dos tipos principales de entrevistas:
 - **Entrevistas Estructuradas:** La entrevista tiene un estructura fija, y no hay exploración de actitudes individuales.
 - **Entrevistas Flexibles:** Generalmente tienen un conjunto de temas establecidos, pero una una secuencia fija, y el entrevistador es libre de seguir las respuestas de los entrevistados para obtener mayor información de sus actitudes personales.
- **Cuestionarios y Encuestas:** Hay dos tipos de preguntas: cerradas (se pide al encuestado que seleccione una respuesta entre un conjunto de respuestas alternativas) y abiertas (el encuestado puede dar libremente su propia respuesta). Las preguntas cerradas normalmente tienen algún tipo de escala de valoración. Tres de estas escalas son: una escala de valoración multipunto, la escala Likert y la diferencial semántica. En ocasiones se emplea un cuestionario antes y después de los estudios de rendimiento de usuarios. Estos se conocen como pre- y post-cuestionarios.
- **Experimentos Tradicionales:** Este tipo de estudios experimentales son experimentos tradicionales de Psicología que se realizan para tratar aspectos concretos del diseño de interfaces humano-computador. El investigador puede manipular un número de factores asociados con el diseño y estudio del efecto en aspectos variados del rendimiento del usuario. Unos estudios experimentales bien diseñados normalmente tienen una hipótesis claramente enunciada y concluyen con un análisis estadístico de los datos recogidos.
- **Ingeniería de Usabilidad:** Los autores presentan a la ingeniería de usabilidad como un "método" más de la IPO, definido como un proceso en el que la usabilidad de un

producto es especificada cuantitativamente y por adelantado. Entonces, mientras el producto se construye, puede demostrarse si alcanza o no los niveles establecidos de usabilidad. Comprende las siguientes técnicas:

- **Tareas de Referencia:** Test que se realizan para monitorizar el rendimiento del usuario en un laboratorio de usabilidad. Se da a un conjunto de usuarios una serie de tareas específicas a completar en un entorno controlado.
- **Análisis de Impacto:** Esta técnica consiste en listar los atributos de usabilidad junto con las decisiones de diseño propuestas. A continuación, el porcentaje de impacto de cada solución de diseño se estima para cada uno de los atributos y se introduce en la parte de la tabla correspondiente. Es ligeramente diferente el Análisis de Impacto propuesto por Nielsen.
- **Investigación Contextual:** Es una forma de educación que puede usarse para evaluación. Los usuarios e investigadores participan para identificar y comprender los problemas de usabilidad en el entorno habitual de trabajo del usuario.
- **Evaluación Cooperativa:** Los usuarios se implican en las decisiones acerca de cuáles son los temas a evaluar y cuáles son las técnicas que se van a emplear para recoger y analizar las grabaciones de vídeo. La evaluación cooperativa es una técnica diseñada para ser de bajo coste, y puede utilizarse por diseñadores y usuarios sin conocimiento experto en IPO.
- **Etnografía:** Es un método tradicional utilizado en Antropología. Los investigadores etnográficos procuran sumergirse en la situación sobre la cual quieren aprender algo.
- **Métodos de Inspección:** Los especialistas que conocen tanto la tecnología como los usuarios previstos inspeccionan el sistema. Normalmente, el énfasis se pone en inspeccionar el diálogo de interacción usuario-sistema. Dependiendo del enfoque dado a la inspección hay varias variantes de esta técnica:
 - **Inspección de Estándares:** Inspecciones de conformidad con estándares.
 - **Inspección de Consistencia:** Equipos de diseñadores, al menos uno de cada proyecto, que se reúnen para inspeccionar un conjunto de interfaces para una familia de productos.
- **Revisión Estructurada por Expertos:** Los autores hacen una distinción entre métodos de inspección y este tipo de revisión, que se realiza con tareas predefinidas y siguiendo un plan. Las técnicas pertenecientes a esta categoría son las siguientes:
 - **Evaluación Heurística:** Los autores se refieren a la misma técnica descrita por Nielsen.
 - **Recorridos (*Walkthroughs*):** El objetivo de un recorrido en IPO es detectar problemas pronto de tal forma que pueden ser eliminados. Los recorridos implican construir tareas cuidadosamente definidas a partir de una especificación del sistema o de una maqueta de una ventana. Hay dos variantes específicas de recorridos:
 - **Recorrido Cognitivo:** Es una simulación manual de las actividades cognitivas del usuario para identificar problemas potenciales de usabilidad.

- **Recorrido Pluralístico:** Es la misma técnica descrita por Nielsen.
- **Métodos de Evaluación Analítica:** Estos métodos se utilizan para calcular tiempos de rendimiento de tareas que pueden conseguirse por usuarios expertos, extrayendo la información necesaria de las especificaciones de la funcionalidad del sistema, y de un análisis de tareas en el que las tareas están desgranadas en sus componentes. Según los autores uno de los métodos más conocidos es el Modelo a Nivel de Pulsaciones.
 - **Modelo a nivel de Pulsaciones:** Es un modelo de nivel único que suele emplearse para predecir tiempos de realización de tareas cortas realizadas por usuarios expertos. Se basa en el uso de cuatro operadores físico-motores: K (pulsar una tecla), P (apuntar), H (ir al inicio), y D (dibujar); un operador mental para el usuario: M; y un operador de respuesta del sistema: R. Cada operador tiene un tiempo estimado asignado, los cuales han sido obtenidos empíricamente.

[Shneiderman, 98]

Shneiderman presenta una visión general de toda la problemática relacionada con la usabilidad y con la IU. Si embargo, su enfoque difiere del de [Preece, 94] en que no pretende ser extensivo en su descripción del campo de la IPO, pues no detalla cada técnica descrita en la disciplina. Por el contrario, detalla una serie de técnicas que considera útiles. Para la mayoría de estas técnicas presenta únicamente una breve descripción y proporciona una referencia. De todas formas, se incluyen aquí todas aquellas técnicas descritas cuya descripción es suficiente para permitir su aplicación, aunque en algunos casos de forma rudimentaria. Las técnicas que propone son las siguientes:

- **Perfiles de Uso:** Estos perfiles caracterizan a los usuarios en varias combinaciones de conocimiento y patrones de uso.
- **TAG (*Task-Action Grammars*):** Esta notación sirve para caracterizar un conjunto completo de tareas. Una vez el conjunto completo está descrito según esta gramática, se puede probar para demostrar su completitud. De esta forma se asegura la consistencia entre las distintas tareas que se pueden realizar con la interfaz.
- **Modelo de Interfaz Objeto-Acción:** Este es el modelo propio de Shneiderman. Es un modelo explicatorio que se centra en objetos y acciones de tareas y en objetos y acciones de interfaz. Tanto objetos como acciones se descomponen de una forma jerárquica. La idea subyacente es anclar objetos y acciones de la interfaz a objetos y acciones de tareas, los cuales ya conoce el usuario. Este tipo de modelo es especialmente aplicable para dirigir interfaces de manipulación, los cuales han reemplazado a los antiguos lenguajes de comandos e interfaces basados en formularios.
- **Observación Etnográfica:** Puesto que los usuarios de interfaces forman una cultura única, los métodos etnográficos para observarles en el lugar de trabajo es probable que vayan incrementando su importancia. Como etnógrafos, los diseñadores de IU consiguen una mejor comprensión del comportamiento individual y del contexto organizacional.
- **Escenarios:** Para sistemas que sufrirán cambios sustanciales (como en el caso de la reingeniería de procesos) o cuando una aplicación novedosa se plantea, normalmente no existen datos fiables acerca del rango y distribución de frecuencias de tareas y

secuencias. Una forma temprana y fácil de describir un sistema novedoso consiste en escribir escenarios de uso y entonces, si es posible, llevarlos a cabo como si fuera teatro.

- **Revisiones por Expertos:** Los revisores son expertos en la aplicación o en el dominio de la IU. Dependiendo en el centro de atención de la revisión, hay diferentes tipos de revisiones:
 - **Evaluación Heurística:** Como se ha detallado previamente en anteriores autores, los revisores expertos critican una interfaz para determinar la conformidad con una lista corta de heurísticas de diseño.
 - **Revisión de Guías:** Se comprueba la conformidad de la interfaz con el documento de guías organizacional u otros.
 - **Inspección de Consistencia:** Los expertos verifican la consistencia a lo largo de una familia de interfaces, comprobando la consistencia de terminología, color, disposición, formatos de entrada/salida, etc.
 - **Recorrido Cognitivo:** Los expertos mantienen una reunión tipo juicio, con un moderador o juez, para presentar la interfaz y discutir sus méritos y debilidades.
- **Test de Usabilidad y Laboratorios:** Los test de usabilidad tienen un número menos de sujetos que los experimentos controlados, y el resultado es un informe con cambios recomendados, como opuesto a la validación o rechazo de hipótesis.
- **Encuestas:** Las encuestas escritas de usuarios son un acompañante familiar, barato y generalmente aceptable para los test de usabilidad y las revisiones por expertos.
- **Test de Aceptación:** Para proyectos de implementación grandes, el cliente o jefe normalmente establece metas objetivas y medibles para el rendimiento hardware y software. Estas nociones pueden ser fácilmente extendidas a la IU. Los criterios explícitos de aceptación deberían establecerse cuando el documento de requisitos se escribe o cuando se ofrece un contrato.
- **Evaluación durante el Uso Activo:** Los refinamientos al sistema se basan en los resultados de las siguientes técnicas de evaluación de la usabilidad durante el uso activo de un sistema:
 - **Entrevistas y Discusiones Focus Groups:** Las entrevistas se llevan a cabo normalmente con usuarios individuales. Después de una serie de discusiones individuales, las discusiones focus groups son valiosas para comprobar la universalidad de los comentarios.
 - **Registro Continuo del Rendimiento del Usuario:** La arquitectura software debería hacer fácil para los gestores del sistema recoger datos acerca de los patrones de uso del sistema, velocidad de rendimiento del usuario, tasa de errores o frecuencia de peticiones de ayuda en línea.
 - **Operadores en Línea o Telefónicos:** Los operadores son una excelente fuente de información sobre problemas que los usuarios tienen y pueden sugerir mejoras y extensiones potenciales.

- **Buzón de Sugerencias o de Reporte de Errores en Línea:** El correo electrónico puede usarse para permitir a los usuarios enviar mensajes a los encargados del diseño o mantenimiento. Un buzón de sugerencias de este tipo anima a algunos usuarios a hacer comentarios productivos, puesto que la escritura de una carta puede verse como una tarea que requiere demasiado esfuerzo.
- **Newsgroups y Bulletin Boards:** Los usuarios pueden tener preguntas acerca del uso o aplicabilidad de un paquete software, y no pueden dirigirse a nadie en particular. En ese caso, los *newsgroups* y *bulletin boards* pueden ser de utilidad.
- **Revistas y Conferencias para Usuarios:** En sistemas con un número sustancial de usuarios que están geográficamente dispersos, los directivos tiene que trabajar más duro para crear un sentimiento de comunidad. Las revistas impresas tienen un aire de respetabilidad atractivo. Las conferencias permiten a los trabajadores intercambiar experiencias con colegas y las reuniones cara a cara incrementan el sentimiento de comunidad entre usuario.
- **Experimentos Controlados Orientados Psicológicamente:** Los investigadores académicos e industriales están descubriendo que la potencia de los métodos científicos tradicionales puede ser empleada provechosamente en el estudio de las interfaces.
- **Métodos de Especificación:** El autor presenta varios métodos de especificación, según las características de la IU que especifican:
 - **Gramáticas:** Son apropiadas para la especificación formal de lenguajes de comandos. Tienen la ventaja de la precisión y de que se pueden emplear herramientas software para comprobar su completitud y corrección. Sin embargo, son difíciles de seguir cuando crecen y resultan confusas para muchos usuarios. Ejemplos de tipos de gramáticas son las de tipo BNF (*Backus-Naur Form*), y las gramáticas multiparte.
 - **Árboles de Menús:** Representan la estructura de árbol en la que está organizado un sistema de menús, mostrando las relaciones entre los distintos elementos de la jerarquía.
 - **Diagramas de Transición:** Se trata de la misma técnica que proponen Hix y Hartson con el nombre de Diagrama de Transición de Estados de la Interfaz.
 - **Diagramas de Estados de Harel:** Este tipo de diagrama de estados permite agrupar estados, para factorizar transiciones comunes, y pueden extenderse con especificaciones de flujo de datos y de restricciones. Representan mejor que los diagramas de transición la concurrencia y la sincronización.
 - **UAN (*User Action Notation*):** Shneiderman coincide con Hix y Hartson en utilizar esta técnica.

[Constantine, 99]

De las fuentes consideradas, [Constantine, 99] es la que más difiere del resto, puesto sigue un esquema particular en su enfoque del desarrollo. Constantine y Lockwood proponen un método que, aunque emplea algunas técnicas de usabilidad básicas, la mayoría de los modelos

propuestos son adaptaciones de modelos utilizados en el campo. Los modelos son la línea a lo largo de la cual el libro detalla el método propio de los autores, por lo que las técnicas están de alguna forma dispersas a lo largo de todo el trabajo y son difíciles de extraer del contexto del método y los modelos particulares utilizados.

Las técnicas propuestas por Constantine y Lockwood son como sigue:

- **Modelo Estructurado de Roles:** Un modelo de roles es una lista de roles de usuarios que va a soportar por un sistema, que describe cada rol en términos de sus necesidades, intereses, expectativas, comportamientos y responsabilidades que caracterizan y distinguen ese rol. Algunos roles se destacan como roles focales, y son los que se juzgan como los más comunes o típicos o que se consideran particularmente importantes desde una perspectiva de negocio o desde el punto de vista del riesgo o de algún otro contenido técnico. El modelo estructurado de roles ofrece una forma metódica de capturar el máximo posible de información relevante sobre la relaciones de los usuarios del sistema. Se organiza en una serie de perfiles como descripción (*incumbents*), habilidad (*proficiency*), interacción, información, criterios de usabilidad o soporte funcional.
 - **Mapa de Roles de Usuario:** Los roles de usuarios y sus relaciones se representan por medio de un Mapa de Roles de Usuario. Captura la visión general de los usuarios del sistema. Las relaciones que se representan pueden ser de afinidad, clasificación o de composición.
- **Card Sorting:** Esta técnica es la misma que la descrita por Nielsen más arriba. Se trata de una técnica simple para ordenar y organizar datos. Los autores describen las siguientes variantes:
 - **Agrupación por Afinidad:** Las tarjetas se agrupan por su similitud aparente o por la estrecha relación que guardan.
 - **Ordenación por Criterio:** Cuando se ordenan los elementos por algún criterio (por ej. importancia relativa). Cuando se tienen muchos elementos a ordenar, se puede aplicar la clasificación por extremos anclados, en la que se va separando la más alta y la más baja según el criterio escogido, en cada paso de ordenación, hasta que se tiene un conjunto de los elementos medios suficientemente pequeño como para ordenarlos directamente.
 - **Voto por Umbral:** Cada participante selecciona de la colección de tarjetas un número predefinido de tarjetas, que se marcan con una trazo por detrás. Cuando todos han tomado su decisión las cartas que superan un cierto umbral de trazos son las seleccionadas.
- **Escenarios:** Se trata de la misma técnica presente entre las propuestas por Preece, et al.
- **Modelado de Tareas - Casos de Uso:** Los autores emplean los casos de uso como su modo preferido de realizar el modelado de tareas. Un caso de uso es un tipo de uso al cual un sistema se puede someter. Otras explicaciones de un caso de uso incluyen: funcionalidad proporcionada, una vista externa de "caja negra", una narración descriptiva, interacción entre un usuario (en algún rol de usuario) y un sistema, y un uso del sistema que es completo y con significado para el usuario. Los casos de uso se describen de una forma estructurada, en la cual la narrativa se divide en dos partes: el modelo de acciones del usuario y el modelo de respuesta del sistema. Hay un modo abstracto de expresar los casos de uso como sigue:

- **Casos de Uso Esenciales:** A un nivel más alto de abstracción, los casos de uso se definen en términos de las intenciones de los usuarios y las responsabilidades del sistema, manteniendo un enfoque libre de tecnología e independiente de la implementación. Pueden utilizarse para trabajar con casos de uso al principio del proceso de desarrollo, sin tener que tomar demasiadas decisiones sobre los detalles de la IU. El mapa de casos de uso particiona la funcionalidad total del sistema en una colección de casos de uso esenciales interrelacionados. Nótese que esencial se refiere al foco abstracto utilizado para la descripción de casos de uso, no para especificar detalles de la IU, y es aplicable a todos los casos de uso, no se refiere a un conjunto particular de casos de uso especialmente importantes.
- **Modelo del Contenido de la Interfaz:** Es una representación abstracta de los contenidos de los distintos espacios de interacción de un sistema. El contenido se puede modelar por medio de papel (una hoja por cada espacio de interacción) y Post-Its que representan las herramientas y materiales que se van a ofrecer al usuario.
- **Mapa de Navegación entre Contextos:** Representa la arquitectura general de la IU modelando las relaciones entre contextos de interacción. Cada contexto se representa con un rectángulo y las flechas que los conectan representan posibles transiciones entre un espacio de interacción y otro.
- **Diseño Integrador (*Both-And Design*):** Se trata de una técnica para la ingeniería creativa de interfaces. La manera de pensar *both-and* busca una síntesis creativa de ideas aparentemente opuestas o alternativas supuestamente exclusivas. Más que un compromiso, busca incorporar lo mejor de ambos mundos y satisfacer metas conflictivas simultáneamente. Los autores describen un proceso para ayudar a encontrar soluciones de este tipo.
- **Prototipado:** Hay diferentes variantes de prototipos, a continuación se clasifican según su naturaleza activa o pasiva:
 - **Prototipos Pasivos:** Incluyen los prototipos de papel, gráficos por ordenador preparados con software de gráficos, y maquetas no funcionales creadas utilizando herramientas de programación.
 - **Prototipos Activos:** También se denominan prototipos funcionales. Pueden ser maquetas, simulaciones o implementaciones limitadas. El término maqueta (*mock-up*) normalmente se refiere a un prototipo de fidelidad y rendimiento limitados construido en un medio software que no se trata de un entorno de programación completo.
- **Técnicas de Diseño de la Ayuda:** Una técnica basada en casos de uso se describe para el diseño del subsistema de ayuda:
 - **Organización de la Ayuda según Casos de Uso:** Si los casos de uso esenciales han sido bien construidos, entonces reflejarán cómo los usuarios piensan y llevan a cabo su trabajo. Cada caso de uso se convierte en una entrada en el fichero de ayuda.
- **Modelo Operacional:** El modelo operacional es una colección de varias influencias operacionales y contextuales que pueden jugar un rol en usabilidad. Estas colecciones se refieren como perfiles. Los factores operacionales que pueden afectar a la arquitectura y detalles de la IU son: características de los usuarios y roles de usuario, aspectos del entorno físico de trabajo, características y limitaciones del equipo y de

los dispositivos de interfaz, y factores operacionales de riesgo genéricos y específicos.

- **Evaluación por Expertos:** Se trata de una evaluación subjetiva de la usabilidad de un producto basada en la experiencia y juicio de un experto o expertos en usabilidad.
- **Inspecciones y Recorridos (*Walk-Throughs*):** Las inspecciones se refieren a cualquiera de varias formas de procesos sistemáticos, con mayor o menor formalidad, para localizar problemas de usabilidad. Las inspecciones de usabilidad emplean desarrolladores y/o especialistas en usabilidad, algunas veces en conjunción con usuarios, para identificar defectos de usabilidad. El autor presenta las siguientes variantes de estas técnicas:
 - **Evaluación Heurística:** Esta técnica es la misma que han propuesto anteriores autores.
 - **Recorrido Cognitivo:** En un recorrido cognitivo, el grupo va paso a paso a través de un escenario de tarea, llevando a cabo un detallado análisis de la intención, conocimiento, procesos de pensamiento e interpretaciones del usuario para cada acción. Se centra en un sólo atributo de usabilidad: Facilidad de aprendizaje.
 - **Recorrido de Usabilidad Pluralístico:** Se trata de un proceso colaborativo que implica a usuarios finales, desarrolladores y expertos en usabilidad, con todos los participantes representando el rol de usuario. Su objetivo es conseguir "empatías coordinadas".
 - **Inspecciones de Usabilidad Colaborativas:** Se trata de un examen sistemático de un producto finalizado o un prototipo, desde el punto de vista de su usabilidad última por los usuarios previstos. El proceso de revisión es un esfuerzo de equipo que incluye desarrolladores software, usuarios finales, expertos de la aplicación o del dominio y especialistas en usabilidad, colaborando para realizar una inspección completa y eficiente. Algunas variaciones de una inspección de usabilidad colaborativa se llaman inspecciones centradas (*focused*). Pueden ser de dos tipos:
 - **Inspecciones de Consistencia:** En las inspecciones de consistencia, el objetivo es identificar inconsistencias entre contextos de interacción y sus contenidos.
 - **Inspecciones de Conformidad:** En las inspecciones de conformidad, el objetivo es identificar desviaciones de los estándares de IU o de las guías de estilo en vigor en la organización. Todos los participantes deben estar familiarizados con los estándares o guías de estilo aplicables, y no se suele incluir a usuarios.
- **Métricas de Usabilidad:** Las métricas de usabilidad son análogas a las métricas de calidad del software. Ofrecen una forma de valorar la usabilidad del diseño que se está elaborando. El autor clasifica las métricas de usabilidad en tres categorías:
 - **Métricas de Preferencia:** Cuantifican la evaluación subjetiva y las preferencias de los usuarios.
 - **Métricas de Rendimiento:** Cuantifican importantes aspectos del uso real, bien en un entorno controlado de laboratorio, bien en el entorno habitual de trabajo.

- **Métricas Predictivas o de Diseño:** Son medidas objetivas de calidad que pueden calcularse a partir de artefactos de diseño, como diseños visuales para las pantallas. Hay tres tipos de métricas predictivas, y los autores detallan una suite esencial de métricas de usabilidad, que se muestran a continuación, organizadas según los tres tipos:
 - **Métricas Procedurales:** Son sensibles a la tarea. Los autores proponen la Eficiencia Esencial, la Concordancia de Tareas, y la Visibilidad de Tareas en esta categoría.
 - **Métricas Estructurales:** Se basan en propiedades superficiales (visibles). La métrica propuesta en esta categoría es la Uniformidad de la Disposición.
 - **Métricas Semánticas:** Son sensibles al contenido. En esta categoría los autores encuadran la métrica de Coherencia Visual.
- **Test en Laboratorio:** Los test de laboratorio implican llevar a cabo test en un lugar específico configurado especialmente para la realización de test de usabilidad. La principal ventaja de este tipo de test de usabilidad es que proporciona un entorno controlado y consistente en el que evaluar software. Comparar los resultados de diferentes test, distintos usuarios o distintos sistemas es más fácil y tiene mejor defensa bajo estas condiciones. Aunque varias de las técnicas descritas a continuación son para test de usabilidad en general (tanto en laboratorios de usabilidad como en estudios de campo), las describimos aquí:
 - **Háblame (pensar en voz alta):** Se trata de la técnica Pensar en Voz Alta nombrada en autores previos.
 - **Información Post-Test:** La mayor parte de planes de test incluyen una entrevista post-test con cada sujeto. Normalmente se agradece a lo sujetos por su participación y se les reafirma acerca de su rendimiento.
 - **Reflexión Diferida:** Se trata de la misma técnica que Nielsen describe como Test Retrospectivo.
- **Test de Campo:** Los test de campo toman los test de usabilidad en el lugar de trabajo. Se lleva a cabo en la organización del usuario, aunque no necesariamente en la estación de trabajo y oficina del usuario.
- **Participación de Usuarios:** La participación de usuarios en el método se limita a donde pueden actuar como expertos, expertos en su trabajo y en su propio uso del sistema. Los usuarios se pueden implicar intensamente en el diálogo colaborativo de requisitos, modelado de tareas, modelado del dominio, definición de estándares y estilo e inspección de usabilidad. Una técnica particular para la participación de usuarios es la siguiente:
 - **JEM (Joint Essential Modelling):** Es un proceso estructurado para colaborar con usuarios para desarrollar especificaciones de requisitos centrados en el uso mediante el modelado concurrente. Se parece de alguna forma a su antecesor JAD (*Joint Application Design*). Las actividades que incluye JEM son las siguientes:
 1. Premodelado y consolidación
 2. Modelado de roles
 3. Modelado de tareas
 4. Evaluación de modelos
 5. Asignación de funcionalidades

[Mayhew, 99]

Mayhew presenta su método, llamado Ciclo de Vida de la Ingeniería de Usabilidad, de una forma muy completa. El libro está estructurado en torno a las actividades de su ciclo de vida, y para cada actividad propone un procedimiento base (denominado por la autora "técnica base") y nombra alguna técnica adicional que se puede aplicar. En este anexo se van a detallar las técnicas que aparecen descritas en los procedimientos base y aquellas técnicas alternativas que la autora describe como útiles, y para las que detalla su forma de aplicación. Las técnicas propuestas son las siguientes:

- **Cuestionarios de Perfiles de Usuarios:** Los perfiles de usuarios describen a los usuarios previstos del sistema, según las siguientes características:
 - Características psicológicas (por ej. actitud, motivación)
 - Conocimiento y experiencia (por ej. velocidad de mecanografiado, experiencia en la tarea)
 - Características del puesto y de las tareas (por ej. frecuencia de uso, estructura de tareas)
 - Características físicas (por ej. daltonismo)

Para cada usuario se incluye una descripción general, una descripción de las características de los usuarios, y un apartado sobre los requisitos de usabilidad para ese tipo de usuario.

- **Entrevistas Contextuales:** Tras reunir toda la información relacionada con el trabajo a realizar (especificaciones de requisitos, reunión con miembros del equipo, reunión con usuarios), y haber identificado los actores y casos de uso clave, se realizan las observaciones o entrevistas contextuales propiamente dichas. En las entrevistas el usuario realiza sus tareas habituales, y el entrevistador le interroga sobre el por qué de sus decisiones y acciones.
- **Diagramas de Afinidad:** Esta técnica se propone para el caso de tener que organizar las notas obtenidas de una serie de entrevistas contextuales, realizadas por varios observadores en distintas sesiones. La autora describe la dinámica de la técnica, por lo cual se ha decidido incluirla entre las técnicas que propone. La técnica consiste en que cada observador anota en un Post-It cada una de las observaciones que va recogiendo de la observación de los usuarios en su entorno habitual de trabajo. Cuando se reúnen todos los observadores, van poniendo sus notas en una pared blanca grande, de una en una, agrupando juntas las notas que parecen estar relacionadas. Según se van añadiendo notas el grupo va reagrupando las notas según criterios en los que esté de acuerdo todo el grupo. Se presenta como paso de organización de ideas, previo a sesiones tipo tormenta de ideas (*brainstorming*).
- **Escenarios de Tareas:** Los escenarios de tareas son instancias de casos de uso que representan las tareas del trabajo en la vida real. Se elaboran estos escenarios para las tareas más representativas de cada tipo de usuario. No hace falta que correspondan exactamente con una tarea concreta que se haya observado en las entrevistas contextuales, pues un escenario puede incorporar los aspectos más interesantes de varias tareas reales combinados.
- **Task Sorting:** Esta técnica sirve para obtener el Modelo de Organización de Tareas directamente de los usuarios. Se les presentan las tareas de bajo nivel escritas en una tarjeta, y se les pide que las agrupen del modo que tenga más sentido para ellos, dada la manera en la que suelen pensar y actuar en su trabajo. Una vez están agrupadas se les pide que den un nombre a cada grupo. Una vez se tienen las jerarquías propuestas

por cada usuario participante en el ejercicio, se intenta extraer de ellas una jerarquía que exprese las características comunes a todas ellas.

- **Objetivos de Usabilidad:** Los objetivos de usabilidad se establecen al comienzo de un proyecto con el fin de que dirijan todas las posteriores decisiones de diseño. Pueden ser de varios tipos:
 - **Objetivos Cualitativos:** Este tipo de objetivos o requisitos describen metas no cuantificables. Son útiles para guiar los esfuerzos iniciales de diseño.
 - **Objetivos Cuantitativos:** Se pueden cuantificar. Hay tres tipos:
 - **Objetivos de Rendimiento:** Cuantifican el rendimiento real de un usuario utilizando un sistema.
 - **Objetivos de Satisfacción:** Cuantifican el nivel de satisfacción del usuario con una interfaz concreta.
 - **Objetivos de Preferencia:** Cuantifican la preferencia de un usuario entre interfaces alternativas, basada en cierto grado de conocimiento de las mismas.
- **Capacidades y Restricciones de Plataforma:** Esta técnica presenta una forma de describir los aspectos relevantes de las plataformas hardware/software sobre las que va a funcionar el sistema.
- **Maquetas de Alta Fidelidad:** Son prototipos que se pueden ejecutar en una máquina. La autora aconseja su uso cuando se dispone de una herramienta de creación de prototipos, y se cuenta con experiencia en su uso.
- **Maquetas de Baja Fidelidad:** Se trata de prototipos en papel. Únicamente incluyen suficiente detalle en aquellos aspectos esenciales para dar al usuario el contexto necesario para poder comprender cada paso en la navegación por la IU.
- **Guía de Estilo del Producto:** Este documento recoge el Modelo Conceptual (esto es, reglas de presentación) y los estándares de diseño de pantallas. Organiza en un único documento todas las decisiones de diseño de la IU con el objetivo principal de conseguir la consistencia en la IU de todo el producto. Puede establecerse a varios niveles: Plataforma, organización, familia de productos o para un producto particular. Es útil que además de las decisiones de diseño incluya la lógica que ha motivado dichas decisiones para permitir futuros cambios.
- **Test Formales de Usabilidad:** Se trata de los test de usabilidad más habituales en el campo de la IPO, en los que se le pide al usuario que realice una serie de tareas con el prototipo del sistema para poder observar qué problemas experimenta el usuario, qué errores comete y cómo se recupera de los mismos. Cuando se realiza en la etapa de evaluación iterativa del Modelo Conceptual, entonces se trata más de sugerir al usuario que piense en voz alta que de medir tiempos con exactitud. Por el contrario, en niveles posteriores de diseño y pruebas, no se interviene tanto porque se prueban objetivos de rendimiento específicos.
- Otras técnicas de evaluación de la usabilidad, que han sido nombradas anteriormente entre las propuestas en [Preece, 94] y [Shneiderman, 98]:
 - **Evaluación Heurística**
 - **Revisiones de Guías**
 - **Inspecciones de Consistencia**
 - **Inspecciones de Estándares**

- **Recorridos Cognitivos**
- **Recorridos Pluralísticos**
- **Técnicas de Test Remoto:** Permiten realizar el test de usabilidad en el entorno de trabajo habitual del usuario, y ahorran posibles desplazamientos y problemas con la agenda. La autora describe cuatro variantes de técnicas de test remoto:
 - **Evaluación por Control Remoto:** El ordenador del usuario se conecta vía Internet o vía línea telefónica con el ordenador del evaluador. El evaluador puede ver (y grabar) la interacción del usuario con el sistema en tiempo real y comunicarse con el usuario, de forma similar a cómo se haría en un test de usabilidad tradicional.
 - **Videoconferencia:** El evaluador se comunica con el usuario y puede ver su interacción con el sistema por medio de un sistema de videoconferencia.
 - **Evaluación Remota Instrumentada:** Consiste en la instalación de monitores software que recogen datos de uso en el ordenador del usuario. Los datos se empaquetan y envían a los evaluadores para su análisis.
 - **Evaluación Remota Semi-Instrumentada:** Los usuarios identifican los problemas de usabilidad en su uso normal de una aplicación. La aplicación tiene una función que permite al usuario describir un problema de usabilidad que está experimentando en un momento dado. La aplicación registra el estado del sistema en el momento en el que el usuario invoca dicha función y le pide la descripción del problema. Se empaqueta la información de estado junto con la descripción del usuario y se envía a los evaluadores.
- **Retroalimentación del Usuario:** La autora nos propone una serie de técnicas para obtener retroalimentación de los usuarios una vez el sistema ha sido ya instalado:
 - **Test de Usabilidad:** Los mismos que se han descrito más arriba.
 - **Entrevistas:** Llevar a cabo entrevistas para obtener las reacciones subjetivas de los usuarios.
 - **Focus Groups:** Es la misma técnica descrita por Nielsen.
 - **Cuestionario:** Para obtener información de un número más amplio de usuarios se pueden elaborar, distribuir y analizar cuestionarios estructurados que incluyen preguntas sobre la usabilidad del producto.
 - **Estudios de Uso:** Está centrado en cómo se usa realmente el sistema. Un estudio de este tipo puede revelar, por ejemplo, que hay funcionalidades que se usan muy poco. Entonces hay que estudiar por qué, porque puede ser debido a que son difíciles de encontrar, de aprender o de recordar, y entonces hay un problema de usabilidad identificado. Hay dos formas de llevar a cabo estudios de uso:
 - **Observación aleatoria:** Se avisa a los usuarios que el observador va a venir en algún momento de una franja horaria. El observador llega en cualquier momento y observa la interacción del usuario con el sistema.
 - **Monitores software:** Un monitor software se activa de forma aleatoria y registra qué funcionalidades se usan y cuándo.

Anexo B

Cuestionario

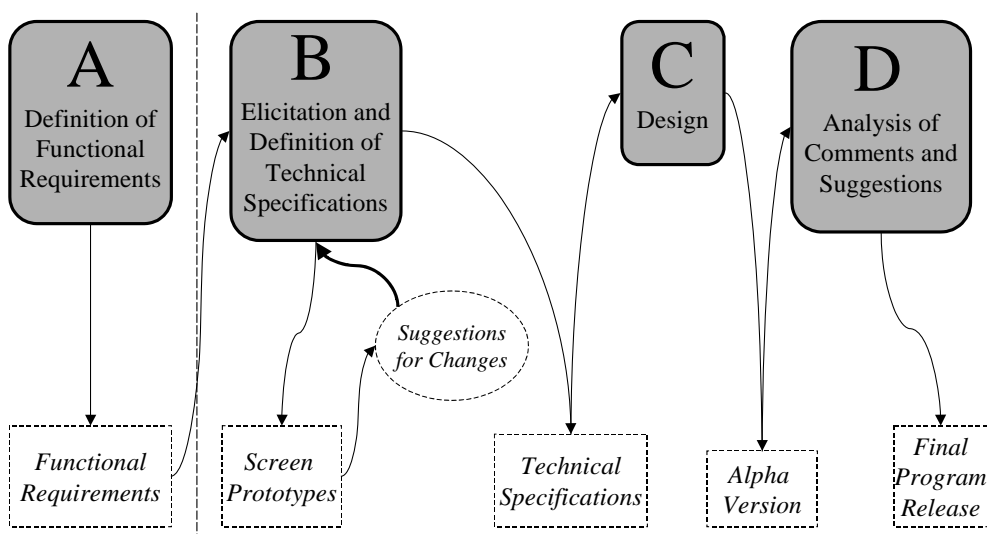
LogicDIS

STATUS Usability Framework Questionnaire

Name: Date:.....

1. You can find below a figure representing the process used at LOGICDIS with its four activities tagged with letters (A to D). The following table contains the usability techniques we saw in the course. For the techniques you have actually used, please mark with a cross the activity in which you have used them. Multiple activities may be marked where applicable.

Usability Technique	Activities			
	A	B	C	D
Observational Techniques				
Visual Brainstorming				
Scenarios				
Affinity Diagrams				
Competitive Analysis				
Personas				
Essential Use Cases				
Paper Prototypes				
Usability Specifications				
Pluralistic Walkthrough				
Card Sorting				
Website Tree				
Navigation Paths				
Product Style Guide				
Heuristic Evaluation				
Usability Testing with Thinking Aloud				
Usability Testing with Performance Measurement				
Usability Questionnaires				



2. Please, indicate for each technique you have used the extent of your agreement or disagreement with each sentence by placing a cross in the appropriate box. Leave blank the rows corresponding to techniques whose usage you have not considered for the project.

		Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree
Observational Techniques	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Visual Brainstorming	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Scenarios	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Affinity Diagrams	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Competitive Analysis	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Personas	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Essential Use Cases	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					

		Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree
Paper Prototypes	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Usability Specifications	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Pluralistic Walkthrough	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Card Sorting	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Website Tree	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Navigation Paths	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Product Style Guide	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Heuristic Evaluation	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					

		Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree
Usability Testing with Thinking Aloud	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Usability Testing with Performance Measurement	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					
Usability Questionnaires	It is easy to understand					
	It is easy to apply correctly					
	It is easy to integrate with other (non-usability) techniques used in development					
	The benefits of applying the technique compensate the effort of its application					

3. For what purpose do you think usability techniques have helped during development:

4. Please, indicate how much the result of usability activities has affected to the following development activities (circle the number that more closely represents your agreement):

- | | | | | | | | |
|--|----------------------------|---|---|---|---|---|--------------------------|
| a) Development of the visible part of the graphical user interface | It has not affected at all | 1 | 2 | 3 | 4 | 5 | It has strongly affected |
| b) Development of the rest of the system | It has not affected at all | 1 | 2 | 3 | 4 | 5 | It has strongly affected |

b.1) Please, write down to which specific activities it has affected and in which way:

5. In which way the application of usability techniques has influenced in the understanding of requirements issues?

6. Please, indicate the extent of your agreement or disagreement with the following sentence by placing a cross in the appropriate box:

Globally, the effort employed in the application of usability techniques pays off, considering the results obtained.

Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree

7. Do you think that, generally speaking, usability techniques are applicable to a wide range of software projects?

We will now ask you some questions about yourself:

8. Developer information

- a) Age.....
- b) Gender (Male/Female).....
- c) Years of experience in the software development field.....
- d) Position (Job title).....

Anexo C

Cuestionario

Alumno de

Master

Cuestionario Técnicas IPO

Fecha:

Como desarrollador

- Indica cuáles de las siguientes técnicas se han aplicado en el desarrollo de vuestro proyecto indicando con una cruz en qué etapa se han empleado.

	Exploración Inicial	Elaboración 1	Elaboración 2
Análisis Competitivo			
Casos de Uso Esenciales			
Investigación Contextual			
Prototipos de Papel			
Especificaciones de Usabilidad			
Evaluación Heurística			
Test de Usabilidad			
Pensar en Voz Alta			
Cuestionarios			
Entrevistas			

- Indica para cada técnica utilizada el grado de acuerdo con la afirmación de cada columna, con un valor entre valor entre 1 y 5 (1 = nada de acuerdo, 5 = completamente de acuerdo). Hay que escribir en la primera columna el nombre de la técnica que se ha utilizado y en las restantes columnas escribir el grado de acuerdo con la afirmación de esa columna.

Técnica	La técnica es difícil de entender	La técnica es difícil de aplicar correctamente	La técnica es difícil de integrar con el resto de técnicas del desarrollo	El desarrollo se ha ralentizado innecesariamente al incluir la técnica
Análisis Competitivo				
Casos de Uso Esenciales				
Investigación Contextual				
Prototipos de Papel				
Especificaciones de Usabilidad				
Evaluación Heurística				
Test de Usabilidad				
Pensar en Voz Alta				
Cuestionarios				
Entrevistas				

3. Para qué crees que te han sido de ayuda las técnicas en el desarrollo del proyecto:

4. Da un valor entre 1 y 5 (1 = no ha afectado, 5 = ha afectado mucho) que refleje cuánto ha afectado el resultado de las actividades de usabilidad al:

a) Desarrollo de la parte visible interfaz gráfica de usuario

1 2 3 4 5

b) Desarrollo del resto del sistema

1 2 3 4 5

b.1) Indica a qué actividades concretas ha afectado y de que manera:

5. ¿Qué influencia ha tenido la aplicación de técnicas IPO en la comprensión de los requisitos?

6. Indica tu grado de acuerdo con la siguiente afirmación con un valor entre valor entre 1 y 5 (1 = nada de acuerdo, 5 = completamente de acuerdo):

El esfuerzo que conlleva aplicar las técnicas IPO compensa a la vista de los resultados obtenidos

1 2 3 4 5

7. ¿Crees que las técnicas IPO son aplicables para cualquier tipo de problema o específicamente para algún problema particular (o dominio particular)?

Cuestiones Sobre el Desarrollador

8. Información sobre el Desarrollador

- e) Edad.....
- f) Sexo (Varón/Mujer)
- g) Años de experiencia profesional en desarrollo de software.....
- h) Puesto de Trabajo (Cargo)

Anexo D

Cuestionario de

Satisfacción

Cuestionario de evaluación

Nombre:

Fecha:

Impresión General

Por favor, marca con un círculo tu opinión personal de los siguientes puntos relativos al sistema que acabas de probar:

	Insatisfecho			Satisfecho	
Satisfacción General	-2	-1	0	1	2

	Peor			Mejor	
Valoración respecto a la versión anterior	-2	-1	0	1	2

(rellenar únicamente si se ha probado anteriormente otra versión del sistema)

Opinión general del sistema	Muy malo			Muy bueno	
	-2	-1	0	1	2
	Desilusionante			Satisfactorio	
	-2	-1	0	1	2
	Complejo			Sencillo	
	-2	-1	0	1	2
	Rígido			Flexible	
	-2	-1	0	1	2

Facilidad de Aprendizaje y de Comprensión

Por favor, indica marcando con un círculo tu grado de acuerdo con las siguientes afirmaciones sobre el sistema que acabas de probar:

	En desacuerdo			Completamente de acuerdo	
El sistema es fácil de aprender	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
Cuando el sistema se utiliza por primera vez es fácil de comprender la forma de trabajar con él	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
El número de pasos para realizar cada tarea es el adecuado	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
Los pasos que se requieren para llevar a cabo una tarea siguen una secuencia lógica	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
El sistema ofrece suficiente información sobre cada paso de cada operación realizada	-2	-1	0	1	2

Por favor, marca con un círculo tu opinión acerca del siguiente punto relativos al sistema que acabas de probar:

	Siempre			Nunca	
El modo en el que funciona el sistema favorece que el usuario cometa errores	-2	-1	0	1	2

Sobre la Interfaz de Usuario

Por favor, indica marcando con un círculo tu grado de acuerdo con las siguientes afirmaciones sobre la interfaz de usuario del sistema que acabas de probar:

	En desacuerdo			Completamente de acuerdo	
La terminología usada en las pantallas del sistema es la adecuada para los usuarios del sistema	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
Se puede deducir qué hace cada función del sistema	-2	-1	0	1	2

	En desacuerdo			Completamente de acuerdo	
Los mensajes y respuestas que ofrece el sistema son consistentes en cuanto a la terminología y forma de expresar la información	-2	-1	0	1	2

Por favor, marca con un círculo tu opinión acerca de los siguientes puntos relativos a la interfaz de usuario del sistema que acabas de probar:

	Oscura				Clara
La lectura y comprensión de fuentes de letras y dibujos es	-2	-1	0	1	2
	Complejas				Intuitivas
Las pantallas en sentido general pueden ser consideradas como	-2	-1	0	1	2
	Escasa				Excesiva
La información mostrada por pantalla es	-2	-1	0	1	2
	Confusa				Clara
La secuencia entre pantallas es	-2	-1	0	1	2
	Nunca				Siempre
La interfaz muestra la situación actual del sistema	-2	-1	0	1	2
	Escasas				Excesivas
Las opciones en pantallas son	-2	-1	0	1	2
	Nada adecuada				Muy adecuada
La distribución de los componentes en las ventanas es	-2	-1	0	1	2

Comentarios:

Anexo E

Cuestionario

Alumno de

Grado

3. Para qué crees que te han sido de ayuda las técnicas en el desarrollo del proyecto:

4. Da un valor entre 1 y 5 (1 = no ha afectado, 5 = ha afectado mucho) que refleje cuánto ha afectado el resultado de las actividades de usabilidad al:

a) Desarrollo de la parte visible interfaz gráfica de usuario

1 2 3 4 5

b) Desarrollo del resto del sistema

1 2 3 4 5

b.1) Indica a qué actividades concretas ha afectado y de que manera:

5. ¿Qué influencia ha tenido la aplicación de técnicas IPO en la comprensión de los requisitos?

6. Indica tu grado de acuerdo con la siguiente afirmación con un valor entre valor entre 1 y 5 (1 = nada de acuerdo, 5 = completamente de acuerdo):

El esfuerzo que conlleva aplicar las técnicas IPO compensa a la vista de los resultados obtenidos

1 2 3 4 5

7. ¿Crees que las técnicas IPO son aplicables para cualquier tipo de problema o específicamente para algún problema particular (o dominio particular)?

Cuestiones Sobre el Desarrollador

8. Información sobre el Desarrollador

- i) Edad.....
- j) Sexo (Varón/Mujer)
- k) Años de experiencia profesional en desarrollo de software.....
- l) Puesto de Trabajo (Cargo)

Anexo F

Índice de

Acrónimos

Acrónimo	Significado
CASE	<i>Computer-Aided Software Engineering</i>
DCH	Diseño Centrado en el Humano
ERP	<i>Enterprise Resource Planning</i>
GOMS	<i>Goals, Operations, Methods and Selection rules</i>
HTA	<i>Hierarchical Task Analysis</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IPO	Interacción Persona-Ordenador
IS	Ingeniería del Software
ISO	<i>International Organization for Standardization</i>
IU	Interfaz de Usuario
JEM	<i>Joint Essential Modeling</i>
LUCID	<i>Logical User Centered Interaction Design</i>
OOSE	<i>Object Oriented Software Engineering</i>
RUP	<i>Rational Unified Process</i>
SWEBOK	<i>SoftWare Engineering Body Of Knowledge</i>
TAG	<i>Task-Action Grammars</i>
UAN	<i>User Action Notation</i>
UML	<i>Unified Modeling Language</i>
UX	<i>User eXperience</i>