

Integración de la IPO en el Proceso de Desarrollo de la Ingeniería del Software: Propuestas Existentes y Temas a Resolver

Xavier Ferré
Universidad Politécnica de Madrid
Facultad de Informática
Campus de Montegancedo
28660 - Boadilla del Monte (Madrid) - España
+34-913366923
xavier@fi.upm.es

Ana M. Moreno
Universidad Politécnica de Madrid
Facultad de Informática
Campus de Montegancedo
28660 - Boadilla del Monte (Madrid) - España
+34-913366929
ammoreno@fi.upm.es

RESUMEN

La usabilidad del producto software se ha convertido en un tema altamente relevante para la industria del software. Sin embargo, existe una falta de integración de las actividades y técnicas relacionadas con la usabilidad en los procesos de desarrollo software de la IS (Ingeniería del Software). El artículo realiza una revisión de la evolución que ha sufrido la consideración de la usabilidad por parte de la IS, y un análisis de las aproximaciones existentes a la integración de las actividades y técnicas de la IPO (Interacción Persona-Ordenador) en la IS. Además, se detallan los principales problemas que deben solucionarse para que se pueda producir tal integración, y se proponen posibles soluciones.

Palabras Clave

Interacción Persona-Ordenador, Ingeniería del Software, proceso de desarrollo, usabilidad

1. INTRODUCCIÓN

La usabilidad se ha convertido en un factor crítico de calidad en los sistemas software que se desarrollan actualmente, y está recibiendo una atención creciente en los últimos años por parte de la IS [13]. Las técnicas IPO tienen como objetivo incrementar el nivel de usabilidad del producto software, sin embargo hay un gran desconocimiento entre los desarrolladores acerca de las técnicas IPO [8] [31]. Dicha situación puede deberse a la percepción existente en la comunidad de desarrollo software de que los temas relacionados con la usabilidad son más bien ajenos a la IS, y que tienen que aplicarse únicamente para el desarrollo de la IU (Interfaz de Usuario). Se da por hecho que este desarrollo de la IU se lleva a cabo una vez que la parte funcional del sistema, la parte interna, ha sido diseñada.

A pesar de este desconocimiento, la IPO es un campo maduro, en el que utilizan métodos de desarrollo que aseguran que el nivel de usabilidad del producto software desarrollado alcance el mínimo

Se concede el permiso para la reproducción digital o impreso total o parcial de este trabajo sin contraprestación económica únicamente para la utilización personal o en clase. En ningún caso se podrán hacer o distribuir copias de para su explotación comercial. Todas las copias deben de llevar esta nota y la información completa de la primera página. Para cualquier otro uso, publicación, publicación en servidores, o listas de distribución de esta información necesitara de un permiso específico y/o el pago correspondiente.

Interacción 2004, 3-7 mayo, 2004, Lleida (España).

esperado. Estos métodos son particulares a la IPO y se rigen por principios distintos a los de la IS. Sin embargo, ambas disciplinas abordan el producto software desde perspectivas distintas. Muchas organizaciones de desarrollo de software quieren aumentar el nivel de usabilidad de sus productos, pero no están dispuestas a cambiar completamente su proceso de desarrollo por un método de IPO, aunque tal cambio proporcione mejoras en algún atributo de calidad (por ejemplo, la usabilidad).

La divergencia de visión entre ambas disciplinas, sin embargo, se ha ido reduciendo en los últimos años. Por un lado, la IPO ha buscado una mayor formalización en sus métodos, similar a la empleada en la IS, y han surgido propuestas como el ciclo de vida de la ingeniería de usabilidad [24], que define con precisión las actividades del desarrollo, los roles que participan en ellas, y los productos que elaboran, al estilo de la IS. Por otro lado, la IS está dando cada vez mayor importancia a la usabilidad, debido a que la base de usuarios de productos software es hoy en día mucho más amplia que hace unos años. Este acercamiento ha hecho que la integración entre ambas haya sido planteada en más de una ocasión, respondiendo a la demanda que en este sentido se produce desde las organizaciones de desarrollo de software.

El cambio de orientación producido en la IS va a ser analizado en la siguiente sección. En la sección 3 se presentan las propuestas de integración de la IPO en el proceso de desarrollo. Los temas a resolver que se han identificado tras el análisis de las distintas propuestas se detallan en la sección 4. Finalmente, en la sección 5 se recogen las conclusiones alcanzadas.

2. USABILIDAD EN EL PROCESO DE LA IS

En el glosario estándar de terminología de IS de IEEE (IEEE Std 610.12-1990) [11] se define IS como “la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software”. Esta preocupación por parte de los ingenieros software por los procedimientos sistemáticos y repetibles, ha hecho que queden en un segundo plano otros temas, como la manera en la que el producto software potencia la creación de modelos mentales adecuados en el usuario, o cómo se puede dar mejor soporte a las tareas que el usuario desea realizar. Por el contrario, uno de los temas que sí ha centrado la atención de la IS es el proceso software. El término “proceso de desarrollo software” se refiere al esquema de desarrollo seguido por una

organización productora de software, esto es, la serie de actividades llevadas a cabo para crear y mantener sistemas software. Los desarrolladores típicamente siguen el proceso software de su organización, con la suposición implícita de que el empleo de un buen proceso de desarrollo conlleva la obtención de un buen producto software.

Entre los atributos de calidad del software, la usabilidad ha estado presente desde los años 70 en las descomposiciones de calidad utilizadas en la IS [2] [23]. Sin embargo, este criterio de calidad ha recibido tradicionalmente menos atención que otros, como la eficiencia, la fiabilidad o la validez. Mientras estos últimos atributos pueden tratarse sin intervención de los usuarios y/o clientes, afrontar la usabilidad requiere una fuerte interacción con usuarios. En el objetivo de la IS de convertir el desarrollo en un proceso sistemático, intentaba reducirse al mínimo la imprevisibilidad que introduce el factor humano al tener que tratar con usuarios. El objetivo tradicional ha sido obtener unos requisitos del sistema, los cuales quedan congelados para poder desarrollar el sistema sin interacción con elementos externos perturbadores al equipo de desarrollo, hasta que el cliente realiza la evaluación del sistema en las pruebas de aceptación. Este planteamiento está en consonancia con el enfoque de desarrollo predominante en la IS hasta la década de los 90: El ciclo de vida en cascada, cuya característica fundamental consiste en unos requisitos congelados [21].

Esta relegación de la usabilidad puede observarse en la presencia marginal del diseño de la interacción en los principales estándares de proceso de desarrollo de la IS: Los estándares ISO/IEC 12207:1995 [15] e IEEE 1074-1997 [12]. Se trata de dos estándares que se complementan: Mientras el estándar ISO 12207 proporciona las condiciones que debe cumplir un proceso, el estándar IEEE 1074 detalla los bloques básicos con los que construirlo.

En el estándar ISO 12207:1995 no se mencionan directamente ni la usabilidad ni las actividades de IPO. Por el contrario, sí que se indica que la participación de usuarios debe ser planificada, pero dicha participación se circunscribe únicamente a ejercicios de establecimiento de requisitos, a demostraciones de prototipos y a evaluaciones. Los usuarios juegan un papel pasivo en las escasas actividades en las que se les incorpora. Otras cláusulas de la norma podrían interpretarse de tal forma que permitieran aplicar las técnicas IPO, pero el estándar no incluye ningún tipo de orientación acerca de cómo debería realizarse tal aplicación. Como resumen, podemos afirmar que la generalidad de la que hace gala la norma, junto con su flexibilidad para acomodar distintos enfoques de desarrollo de software permiten la integración de técnicas y actividades IPO, pero no incluye ninguna indicación que facilite dicha tarea de integración.

El estándar IEEE 1074-1997 [12] es más concreto que el ISO/IEC 12207:1995, puesto que detalla entradas y salidas para cada actividad, e incluye guías adicionales sobre cómo interpretar cada actividad. Respecto a la usabilidad, a pesar de que se menciona en relación con los requisitos de la IU y con la gestión de riesgos, el estándar no aborda ninguna de las actividades necesarias para gestionar la usabilidad del producto software. Algunas actividades de evaluación podrían interpretarse de forma que permitiera la aplicación de tareas de evaluación de la usabilidad. De forma similar, se podría interpretar la parte dedica al planteamiento de la necesidad (*statement of need*) para que incluyera actividades

relacionadas con la usabilidad. Sin embargo, el bajo nivel de granularidad de las actividades del estándar no encaja bien con las prácticas habituales de la IPO, donde algunas de sus actividades podrían muy bien corresponder a varias actividades de la norma de IEEE, requiriéndose un esfuerzo en la integración de todos los productos que deberían crearse según el estándar. Por lo tanto, la integración de técnicas y actividades de IPO en las actividades del estándar IEEE plantea importantes desafíos.

Esta poca presencia de actividades relacionadas con la usabilidad en estos dos estándares desarrollados en los años 90, está cambiando hacia una mayor presencia de la usabilidad en la IS, según se puede observar por cómo la tratan el SWEBOK (*Guide to the Software Engineering Body of Knowledge*) [14] y la primera enmienda al estándar ISO/IEC 12207 [17].

El SWEBOK [14] ha sido desarrollado por la IEEE Computer Society en un esfuerzo por recoger lo que se consideran conocimientos comúnmente aceptados en el campo de la IS. En la parte dedicada a requisitos software, aunque no se alude a la usabilidad directamente, se nombran como técnicas de educación, actividades y técnicas típicas de la IPO: Entrevistas, escenarios, prototipado y observación de usuarios. No sorprende esta convergencia de tipos de actividades entre IS e IPO, si se presta atención a la definición ofrecida para los ingenieros de requisitos: "Uno de los principios fundamentales de una buena ingeniería del software es que haya buena comunicación entre los usuarios del sistema y los desarrolladores. Es el ingeniero de requisitos quien sirve de interlocutor para esta vía de comunicación" [14]. En la parte de diseño software, se especifica que el diseño considerado de invención (el destinado a conceptualizar y especificar un sistema para satisfacer ciertas necesidades novedosas) no se va a tratar, pues en IS se conoce como diseño más bien al destinado a descomponer un sistema en partes. El diseño de invención, sin embargo, es muy importante en la IPO para poder proporcionar un modelo mental del sistema al usuario [26] [27], de tal forma que encaje al máximo con sus experiencias previas y sus expectativas. Sí se identifica en el SWEBOK como tema clave en el diseño de software, con implicaciones transversales a todo el diseño, el enfoque elegido para la interacción con los usuarios en aquellos sistemas que sean interactivos, aunque no profundiza en este tema. Por contra, el diseño de la IU (Interfaz de Usuario) se considera como perteneciente a una disciplina relacionada con la IS (la disciplina de Ciencia Cognitiva y Factores Humanos), pero no parte de ella. La usabilidad sí que se nombra como parte de los atributos de calidad, y se destaca su importancia en sistemas de alta confiabilidad (*high dependability systems*). Mientras estos temas relacionados con la usabilidad se citan con referencias de textos de la IS, en la parte de pruebas sí que se nombran los test de usabilidad con la referencia al trabajo de Rubin [30], perteneciente a la IPO. Por tanto, podemos afirmar que los temas relacionados con la usabilidad están presentes en varias de las áreas de conocimiento tratadas en el SWEBOK, aunque tímidamente comparado con lo que sería de esperar debido a la convergencia de los objetivos tratados por ambas disciplinas.

El cambio que puede suponer un hito en la integración de la usabilidad en el proceso de desarrollo software, es la inclusión de un nuevo proceso de usabilidad entre los procesos de soporte del ciclo de vida del estándar ISO/IEC 12207, en la enmienda publicada a dicho estándar en el 2002 [17]. El nuevo proceso de usabilidad está basado en la información descrita en el estándar ISO 13407:1999 [16], que es el estándar de proceso de desarrollo

de referencia en la IPO. El proceso de usabilidad 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 en este sentido podemos decir que la descripción del proceso de usabilidad es bastante completa, incluyendo una mención a las principales actividades en un proceso centrado en el usuario. Con esta adición al estándar de proceso ISO/IEC 12207 la gestión de la usabilidad pasa de ser una labor realizada en organizaciones de desarrollo de software donde se produce un interés especial hacia la usabilidad, a ser una parte de cualquier proceso que alegue conformidad con el estándar. Tomando como base el proceso de usabilidad definido en este estándar se puede abordar la integración de técnicas y actividades de la IPO en el proceso de desarrollo con una base normativa. 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 gestionarse en cada proyecto.

Hasta hace pocos años se podía observar que la IPO y la IS habían recorrido caminos separados, a pesar de tener preocupaciones y fines comunes. Pero podemos afirmar que dicha separación está destinada a desaparecer, puesto que la IS está prestando una atención creciente a los temas relacionados con la usabilidad.

3. PROPUESTAS DE INTEGRACIÓN DE LA IPO EN EL PROCESO DE DESARROLLO

Las aproximaciones de integración de la IPO en el proceso de desarrollo de la IS analizadas en esta sección se van a clasificar en aquellas provenientes del campo de la IPO y aquellas provenientes del campo de la IS.

3.1 Procesos IPO y su Relación con el Proceso de Desarrollo

Entre los procesos IPO, encontramos referencias a su posible integración en el proceso de desarrollo más amplio en el Ciclo de Vida en Estrella de Hix y Hartson [10], en el Diseño Centrado en el Uso de Constantine y Lockwood [6], y en el Ciclo de Vida de la Ingeniería de Usabilidad de Mayhew [24].

3.1.1 Ciclo de Vida en Estrella (Star Life Cycle)

El Ciclo de Vida en Estrella de Hix y Hartson [10] es un proceso centrado en el usuario que establece las principales actividades IPO que deben llevarse a cabo en el desarrollo. No prescribe un orden particular para las actividades, pero da un papel predominante a la evaluación de la usabilidad, que es ubicada en el centro de la estrella que representa las actividades del desarrollo. El modelo incluye dos actividades, Despliegue y Producción de Software, que los autores consideran como parte del proceso de desarrollo del dominio constructivo (*constructional domain*) y que no se discuten en el libro. Por tanto, la parte del desarrollo no relacionada con la usabilidad (principalmente implementación según los autores) es muy pequeña comparada con lo que los autores consideran como actividades relacionadas con la usabilidad.

Hix y Hartson describen las vías de comunicación que deberían existir entre las actividades de usabilidad (de diseño de la interacción con el usuario) y el diseño del software. Separan estrictamente el diseño de la IU del desarrollo del resto del

sistema software, con dos actividades que las conectan: Análisis de Sistemas y Pruebas/Evaluación. El grupo de análisis de sistemas alimenta requisitos tanto al grupo de diseño del dominio como al grupo de diseño de la interacción con el usuario. Es un enfoque relativamente simplista a la integración IPO-IS, y los autores admiten que "es preciso investigar como se pueden comprender y soportar mejor las necesidades de comunicación reales de este complejo proceso" [10].

3.1.2 Diseño Centrado en el Uso (Usage-Centered Design)

Constantine y Lockwood [6] proponen una colección de actividades coordinadas que contribuyen a la usabilidad, agrupadas en el método de Diseño Centrado en el Uso. Su modelo de actividades de diseño centradas en el uso incluye algunas actividades que corresponden al proceso de desarrollo más amplio (diseño de la Estructura de Objetos, Construcción Concéntrica e Iteración Arquitectural), junto con actividades de usabilidad puras, como Modelado de Tareas o Modelado del Contenido de la Interfaz. Los modelos que proponen Constantine y Lockwood son atractivos para los ingenieros software, puesto que el tipo de modelado es más cercano al utilizado en la IS que en otros métodos IPO. No en vano Larry Constantine es una figura reconocida de la IS en el tema del desarrollo estructurado, que en los últimos 15 años ha derivado su trabajo hacia el desarrollo de software usable, y por tanto a la IPO. En concreto, la técnica de casos de uso esenciales, que son un pilar del enfoque centrado en el uso, son una reinterpretación de la técnica de casos de uso, la cual es muy popular en el desarrollo orientado a objetos de la IS. Los casos de uso esenciales pueden, por tanto, servir de puente para la integración de la IPO en el proceso de desarrollo. De hecho, dos populares libros de referencia de la IS ([4] y [21]) recomiendan el trabajo de Constantine y Lockwood acerca de los casos de uso esenciales.

En cuanto a la integración de la usabilidad y el diseño de la IU en el proceso de desarrollo, Constantine y Lockwood ofrecen una serie de consejos, admitiendo que no hay una única forma de enfocar dicha integración. Por tanto, dejan el tema de la integración para ser resuelto de forma particular en cada caso. Indican que "buenas estrategias de integración de la usabilidad en el ciclo de vida acomodan de forma conjunta las nuevas prácticas con las antiguas, modificando las practicas actuales para incorporar la usabilidad en los procesos de diseño y análisis, al tiempo que se particulariza el diseño centrado en el uso a la organización y sus prácticas" [6]. Aunque los autores describen algunas técnicas que son más cercanas al modelado practicado en la IS, esta propuesta no está formalizada en términos de proceso de desarrollo (según se entiende en la IS), y su trabajo se dedica más bien a detallar las técnicas que a especificar el proceso en términos de dependencias, productos y roles.

3.1.3 Ciclo de Vida de la Ingeniería de Usabilidad (Usability Engineering Lifecycle)

Deborah Mayhew [24] propone el Ciclo de Vida de la Ingeniería de Usabilidad para el desarrollo de IUs usables. El ciclo de vida estructura las actividades en tres fases: Análisis de Requisitos, Diseño/Pruebas/Desarrollo, e Instalación. Este proceso de desarrollo sigue un enfoque de ciclo de vida en cascada [29]: Una fase inicial de Análisis de Requisitos, seguida de la fase de Diseño/Pruebas/Desarrollo, y finalmente la de Instalación. Únicamente se vuelve a la fase de Análisis de Requisitos en caso

de no haberse tratado toda la funcionalidad, por tanto no se trata de un enfoque verdaderamente iterativo.

A pesar de esta semejanza con el ciclo de vida en cascada, se trata de 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. Aunque Mayhew afirma que el método está destinado sólo al desarrollo de la IU, se incluyen actividades íntimamente relacionadas con la ingeniería de requisitos, como por ejemplo el Análisis Contextual de Tareas. La autora indica cómo se debe integrar su propuesta con el método OOSE (*Object-Oriented Software Engineering*) [18] y con los métodos de prototipado rápido, pero admite que la integración de la ingeniería de la usabilidad con la IS debe ser particularizada y que el solapamiento entre actividades de usabilidad y de IS no está completamente claro. Por otra parte, las indicaciones de cómo se integra el método en OOSE y en los métodos de prototipado rápido están descritas a un nivel muy general, y el desarrollo de la IU se presenta como una actividad que es bastante independiente del desarrollo del resto del sistema.

De forma adicional, choca encontrar la siguiente definición que Mayhew da de la IS como "un enfoque al desarrollo de software que implica definir los requisitos de la aplicación, establecer objetivos y probar en ciclos iterativos hasta que se cumplen los objetivos" [24]. 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 el resto del libro.

3.2 Propuestas de Integración Basadas en Procesos de IS

En esta sección se presentan las propuestas de integración que han tomado como base modelos de proceso de la IS. En primer lugar se trata el método MUSE (*Method for Usability Engineering*) [22], a continuación la propuesta de Costabile [7], y finalmente el plug-in de Experiencia del Usuario de RUP (*Rational Unified Process*).

3.2.1 MUSE

Lim y Long proponen el método MUSE [22] para diseñar la IU. Se trata de uno de los métodos de usabilidad mejor estructurados en la fecha de su publicación (1994). MUSE se divide en tres fases: Educación y Análisis de Información, Síntesis de Diseño y Especificación de Diseño. El método tiene como objetivo la integración con métodos de ingeniería del software, y se describe con tal fin cómo se integraría con el método JSD (*Jackson System Development*) [3]. Se centra principalmente en la especificación del diseño, puesto que los autores identifican como deficitaria la integración en dicha fase, mientras que, siempre según sus autores, etapas posteriores como la evaluación de la usabilidad están bien cubiertas en la literatura existente.

MUSE está basado en el principio de retrasar el compromiso con el diseño, para asegurarse de que el diseño detallado es precedido por un análisis del diseño y una definición conceptual apropiados. Por otra parte, comparando MUSE con el enfoque de prototipado rápido, Lim y Long 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*). Por lo tanto, MUSE sigue un

ciclo de vida en cascada, el cual es un obstáculo a la aplicación de un enfoque verdaderamente iterativo típico de la IPO.

Puesto que MUSE es un método estructurado, sus autores afirman que resulta fácil su integración con cualquier método estructurado, presentando el detalle de la integración con JSD como ejemplo. El método JSD es etiquetado por Lim y Long como un método que se utiliza principalmente para el desarrollo de sistemas de tiempo real. Este tipo de sistemas son una parte muy pequeña del conjunto de sistemas interactivos, por lo tanto el detalle de la integración de MUSE con JSD no resulta muy útil desde un punto de vista general. En cuanto a su posible integración con otros métodos de IS, el uso de técnicas tales como diagramas de estructuras y redes semánticas, cuyo uso cada vez es menor en la práctica de la IS, hace difícil su integración con los métodos actuales de la IS, especialmente en aquellos basados en el desarrollo orientado a objetos.

El proyecto BIUSEM [1] aplicó el método MUSE a tres proyectos de desarrollo en distintos campos, con distintos métodos de IS, con el fin de evaluar la aplicabilidad de MUSE. El proyecto produjo resultados positivos en cuanto a que la aplicación de MUSE mejoró la usabilidad del producto, y en cuanto a que compartir con el equipo de IS la visión de usabilidad ayudó a la educación de requisitos centrados en el usuario. Pero, a pesar de estos resultados positivos, el equipo del proyecto afirmó que "el libro y los artículos publicados describiendo MUSE son innecesariamente complicados y actúan de forma disuasoria para un uso más amplio" [1].

3.2.2 Propuesta de Costabile

Costabile [7] 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 software. En concreto, propone una modificación del ciclo de vida software para incluir las actividades relacionadas con la usabilidad, y toma como base para dicha modificación el ciclo de vida en cascada, al que denomina ciclo de vida estándar. La modificación consiste en añadir dos actividades puras de usabilidad (análisis de usuarios y de tareas por un lado, y escenarios y especificaciones de la IU por el otro), y otra actividad intermedia (prototipado y pruebas) que aparece dos veces entre actividades del ciclo de vida en cascada. Es posible volver a una fase anterior desde cualquier fase del ciclo de vida, y estas vueltas hacia atrás en la cascada junto con la actividad de prototipado y pruebas que aparece dos veces, son presentadas por Costabile como la forma de enfatizar el carácter iterativo del desarrollo. Así justifica la autora la condición de iteratividad necesaria en cualquier enfoque centrado en el usuario.

La propuesta de Costabile parte de una concepción errónea desde el punto de vista de la IS al considerar el ciclo de vida en cascada como ciclo de vida estándar. El enfoque de la cascada está prácticamente superado en la IS, proponiéndose únicamente para problemas de complejidad baja o media, en 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" [9], y afirma que el ciclo de vida en cascada está obsoleto. Larman identifica los siguientes problemas con el ciclo de vida en cascada [21]: Mitigación tardía de los riesgos, especulación e inflexibilidad de los requisitos y el diseño, alta complejidad y baja adaptabilidad. 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 [10] esta realidad al afirmar que, a pesar de lo que las metodologías les indican en cuanto a fases separadas y/o enfoques *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.

3.2.3 Plug-in de Experiencia del Usuario para RUP

El Proceso Unificado [19] es el proceso que está recibiendo en la actualidad la mayor atención en la IS, puesto que está impulsado por los principales metodólogos del desarrollo orientado a objetos de la década de los 90: James Rumbaugh, Ivar Jacobson y Grady Booch. El Proceso Unificado abraza 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 perniciosa identificación actividad-etapa que constituye la base del ciclo de vida en cascada. De los procesos 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 más arriba, puesto que representa una integración mucho menor de la usabilidad, se ha incluido por el papel preponderante del que disfruta el Proceso Unificado entre los modelos de proceso de la IS.

El Proceso Unificado no considera directamente la usabilidad, 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 de IPO del análisis de tareas. Por tanto, podría servir de punto de partida para la integración de la usabilidad en el proceso. 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 que gozan desde un punto de vista centrado en el usuario.

RUP (*Rational Unified Process*) es una particularización del modelo de proceso representado por el Proceso Unificado, y es vendido por la empresa Rational 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 su versión del 2002 [28], RUP cuenta con un *plug-in* de Experiencia del Usuario (*User Experience*), que trata de incorporar al proceso las actividades y técnicas habituales

en el campo del desarrollo de sistemas web para el desarrollo del concepto del sitio web (el cual normalmente dirige el desarrollo completo). Se basa en el trabajo sobre modelado para la web de Conallen [5], y existen grandes semejanzas entre los objetivos de la Experiencia del Usuario y preocupaciones clásicas de la IPO. 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" [5]. A pesar de lo prometedora que resulta esta definición desde una perspectiva IPO, el trabajo de Conallen se dedica únicamente al modelado. Los productos en los que se centra el equipo de Experiencia del Usuario son los siguientes según Conallen: Descripciones de ventanas y contenidos, escenarios *storyboard*, y caminos navegacionales a través de las pantallas.

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. Sin embargo, es un indicador del creciente interés que se puede observar en el campo del desarrollo para la web hacia el tema que nos ocupa: la integración de una perspectiva basada en la usabilidad en el proceso de desarrollo.

3.3 Resumen de las Propuestas

Para poder comparar las propuestas analizadas, se muestra en la Tabla 1 un resumen de los principales puntos fuertes y carencias de cada propuesta según cinco criterios concernientes a la integración de técnicas y actividades de usabilidad en el proceso de desarrollo software. La fila Iteratividad detalla si la propuesta es verdaderamente iterativa o si, por el contrario, conlleva un enfoque de desarrollo basado en el ciclo de vida en cascada. La fila etiquetada Formalidad indica si el proceso es descrito únicamente por medio de actividades de alto nivel, o bien describe actividades de grano fino, junto con sus dependencias a un nivel detallado, las técnicas a aplicar en cada actividad y los productos que generan. La fila Cobertura se refiere al objetivo de la propuesta según sus autores: Si está dirigido al desarrollo del sistema completo, o bien se presenta como un método para el diseño de la IU. La fila Integración indica cómo se aborda la integración de la usabilidad en el proceso de desarrollo, a qué nivel de detalle se trata dicha integración. Finalmente, la fila Enfoque recoge si la propuesta recoge la perspectiva IPO del desarrollo, y por tanto puede denominarse centrada en el usuario. Para este último valor no se ha considerado el hecho de tener o no un enfoque verdaderamente iterativo a pesar de ser necesaria dicha condición para un enfoque centrado en el usuario, por haberse tratado ya en la fila Iteratividad.

Se puede observar en la Tabla 1 que las principales carencias de las propuestas consisten en su no iteratividad, la falta de formalización de los procesos que definen (según el punto de vista de la IS), la cobertura únicamente del diseño de la IU, la falta de detalle en cuanto a cómo realizar la integración, o no incorporar un enfoque centrado en el usuario. Ninguna de las propuestas existentes da solución a todas estas carencias.

Tabla 1. Comparación de Propuestas de Integración de la Usabilidad en el Proceso de Desarrollo

	Ciclo de Vida en Estrella	Diseño Centrado en el Uso	Ciclo de Vida de la Ingeniería de Usabilidad	MUSE	Propuesta de Costabile	Plug-in de Experiencia del Usuario para RUP
Iteratividad	Iterativo	Iterativo	Nominalmente iterativo, pero basado en la cascada	Nominalmente iterativo, pero basado en la cascada	Nominalmente iterativo, pero basado en la cascada	Iterativo
Formalidad	Actividades de alto nivel	Actividades de alto nivel	Actividades de grano fino	Actividades de grano fino	Actividades de alto nivel	Actividades de grano fino
Cobertura	Diseño de la IU	Sistema completo	Diseño de la IU	Diseño de la IU	Sistema completo	Sistema completo
Integración	Indicaciones generales	Indicaciones generales	Describe dependencias con actividades de OOSE y con fases de un método de prototipado rápido	Detallada con el método JSD	Integración con el ciclo de vida en cascada (en el que se basa)	Detallada (se trata de un proceso de la IS con una pequeña ampliación de usabilidad)
Enfoque	Centrado en el usuario	Centrado en el usuario	Centrado en el usuario	Centrado en el usuario	Centrado en el usuario	Parcialmente centrado en el usuario

4. TEMAS A RESOLVER

Una vez analizadas las propuestas de integración vamos a identificar los principales obstáculos que podemos observar para que la integración de la usabilidad en el proceso de desarrollo pueda llevarse a cabo de forma satisfactoria.

4.1 Diseño de la IU vs. Diseño de la Interacción

En la IPO se entiende por Diseño de la IU algo más amplio de lo que éste término significa en un contexto de IS.

En IS únicamente se refiere al diseño de los elementos concretos que van a formar parte de la IU y su comportamiento asociado. No incluye ningún tipo de actividad que tenga que ver con la ingeniería de requisitos. Además, en la IS es un principio ampliamente aceptado 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.

Por otra parte, 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" [10]. Se puede observar que esta definición tiene una cobertura mucho mayor de actividades de todo el proceso de desarrollo.

Uno de los principales obstáculos para la integración de la IS y de la IPO es la diferencia de conceptos y terminología entre ambas

disciplinas. Esta diferencia se hace especialmente patente en la denominación de lo que la IPO se atribuye como su especialidad: El diseño de la IU. Sugerimos utilizar el término diseño de la interacción, entendiendo por interacción la coordinación del intercambio de información entre el usuario y el sistema [8], para transmitir a los ingenieros software que la usabilidad no tiene que ver únicamente con el diseño de la parte visible de la IU. Para el diseño de la interacción se tienen que desarrollar actividades y técnicas que tienen mucho más que ver con la concepción del sistema y con los requisitos del mismo tal y como se detalla en el siguiente punto.

4.2 Usabilidad en la Ingeniería de Requisitos

Las propuestas de integración provenientes de la IPO, tal y como se ha detallado más arriba, parten de la idea de dos procesos de desarrollo paralelos con puntos de comunicación, por un lado el proceso de diseño de la interacción (proceso IPO) y por otro el del resto del sistema (proceso de la IS). Esta hipótesis se basa en una supuesta independencia de los temas que trata cada uno de los procesos, que requieren una cierta coordinación pero que pueden ser realizados con relativa independencia.

Sin embargo, 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 la subdisciplina conocida como 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 resultados positivos en cuanto a la usabilidad del producto final, enfatizando dicha perspectiva en todo el desarrollo.

El enfoque ideal desde el punto de vista de los investigadores IPO sería aquel en el que los expertos en IPO realizan las tareas relacionadas con los requisitos, y todo el desarrollo gira en torno a su trabajo. Mayhew defiende este enfoque cuando afirma:

"Presento las tareas de ingeniería de usabilidad en 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 interfaz de usuario debería dirigir el proceso de desarrollo general, en vez de ser dirigidas por éste o ser incidentales al mismo" [24]. De todas formas, debido a que el desarrollo de software en la mayor parte de las organizaciones de desarrollo de software está dirigido por los conceptos de la IS, un cambio de este calibre resulta extremadamente difícil. Parece poco probable que los ingenieros software admitan que todo su trabajo debe estar dirigido por expertos en IPO, porque los requisitos están reconocidos en la IS como la piedra angular del desarrollo, y los ingenieros software no van a ceder el control sobre un tema de tamaño importancia en el desarrollo de software. Consideramos que, en las organizaciones donde el desarrollo viene realizándose en torno a la IS, y en las que se quiere introducir la usabilidad, será necesario definir una integración estrecha de actividades y técnicas de usabilidad en las actividades del desarrollo que tengan relación con la ingeniería de requisitos. Según nuestra visión, el enfoque alternativo, consistente en mantener dos procesos separados, uno de IPO y otro de IS, con canales definidos de comunicación, no es adecuado, debido a que la usabilidad es una característica de calidad del software tan importante que debe estar totalmente integrada en el desarrollo.

4.3 Desarrollo Iterativo

El enfoque iterativo de desarrollo es una de las prácticas básicas de la IPO, nombrado en [6], [10], [16], [25], [27] y [32]. La complejidad del lado humano en la IPO hace casi imposible la creación de un diseño correcto al primer intento. Temas cognitivos, sociológicos, formativos, físicos y emocionales pueden jugar un papel importante en cualquier interacción usuario-sistema, y un enfoque iterativo en el desarrollo parece la forma más sensata de tratar con la complejidad que se genera.

Por un lado, tenemos que la literatura de IS ha ido aceptando el enfoque iterativo como el mejor para problemas de complejidad media-alta, y para aquellos en los que no se tiene un elevado conocimiento del dominio, como alternativa a un enfoque basado en el ciclo de vida en cascada. Sin embargo, en la práctica se ha seguido manteniendo una mentalidad de desarrollo en cascada, puesto que se trata de un modelo muy atractivo desde el punto de vista estructural, por la ilusión de orden y sencillez que proporcionan a una tarea tan complicada como es el desarrollo de sistemas software. Kroll et al. [20] y Larman [21] indican que el error más común en la aplicación del Proceso Unificado es superponer la cascada a las etapas del mismo. Por tanto, a pesar de que la aceptación del enfoque iterativo por parte de la IS parece hacer más fácil la integración de una perspectiva centrada en el usuario, la no aplicación de dicho enfoque a fondo actúa como un obstáculo importante. Es más, en los intentos por integrar la usabilidad en los modelos de proceso de la IS, se ha caído a menudo en el error de tomar como punto de partida un enfoque del desarrollo basado en la cascada, seguramente por simplificación, como ocurre en la propuesta de Costabile analizada en la sección 3.2.2.

4.4 Perspectiva Centrada en el Usuario en Todo el Desarrollo

Si se llevan a cabo actividades de usabilidad de forma independiente de aquellas tradicionalmente realizadas por parte de

la IS, se corre el riesgo de perder la perspectiva centrada en el usuario. Dicha perspectiva se mantiene en todo el desarrollo en los procesos propuestos por la IPO, puesto que es necesaria para la producción de software usable, y es necesario transmitirla a los equipos de ingenieros software. De esta forma, la usabilidad se debe tener en cuenta en todo el proceso de desarrollo, de igual forma que es necesario considerar otros atributos de calidad como, por ejemplo, la fiabilidad.

Si nos centramos en el caso habitual de una organización de desarrollo de software cuyas prácticas están dirigidas por los conceptos y modelos de proceso de la IS mayoritariamente, donde la presencia de expertos en usabilidad es mínima o inexistente, nos encontramos que, como se mencionó más arriba, probablemente no estará dispuesta a cambiar su proceso de desarrollo por uno de la IPO. Se tendrá que buscar una integración más que un cambio completo, y entonces será necesario que la filosofía de usabilidad (la perspectiva centrada en el usuario) se transmita a todo el equipo de ingenieros software. La aplicación de algunas técnicas requerirá personal especializado en IPO, pero creemos que otras técnicas habituales de la IPO, como el *card sorting*, el modelado de usuarios, el análisis de tareas, o el diseño de la navegación, por ejemplo pueden aplicarlas ingenieros software con una formación adecuada en usabilidad. En algunos casos se puede tener en cada equipo un "campeón de usabilidad" que haya sido formado en temas de IPO, y que pueda actuar como enlace con el departamento de usabilidad.

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 este escenario pueda realizarse.

5. CONCLUSIONES

La usabilidad está cobrando cada vez mayor importancia como atributo de calidad en el desarrollo de software, y ello está motivando que la IS busque la integración de actividades y técnicas de la IPO en sus procesos de desarrollo.

Se han analizado las principales propuestas de integración de la usabilidad en el proceso de desarrollo, identificando las carencias de las que adolecen dichas propuestas. Principalmente, la falta de un enfoque verdaderamente iterativo, y la presentación del diseño de la interacción como un proceso independiente del llevado a cabo por la IS.

Finalmente, se han identificado como principales temas a resolver la visión contrapuesta entre IS y la IPO en cuanto al diseño de la IU, la necesidad de integración fuerte en las actividades de requisitos, la aplicación de un enfoque verdaderamente iterativo, y la adopción de un punto de vista centrado en el usuario en todo el proceso de desarrollo. Para abordar estos problemas, será necesario proveer a los ingenieros software de una caracterización de las técnicas y actividades de usabilidad según los conceptos y terminología de la IS, para que puedan decidir en qué puntos de su proceso de desarrollo pueden integrarlas.

6. AGRADECIMIENTOS

La investigación descrita en el presente trabajo ha sido financiada por el Ministerio de Ciencia y Tecnología de España en el marco del proyecto TIC2002-00320.

7. REFERENCIAS

- [1] BIUSEM Consortium. Final Report BIUSEM. Benefits of Integrating Usability and Software Engineering Methods. ESSI Project 10290, 1995.
- [2] Boehm, B. Characteristics of Software Quality. North Holland Publishing Co., 1978.
- [3] Cameron, J.R.. JSP and JSD: The Jackson Approach to Software Development, Second Edition. IEEE Computer Society Press, 1989.
- [4] Cockburn, A. Writing Effective Use Cases. Addison Wesley, 2001.
- [5] Conallen, J. "Building Web Applications with UML. Second Edition". Addison-Wesley, 2003.
- [6] Constantine, L. L., and Lockwood, L. A. D. Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design. Addison-Wesley, 1999.
- [7] Costabile, M.F. Usability in the Software Life Cycle. Handbook of Software Engineering and Knowledge Engineering. ed. S. K. Chang. World Scientific Publishing, Singapore, 2001. 179-192.
- [8] Ferré, X., Juristo, N., Windl, H., and Constantine, L. Usability Basics for Software Developers. IEEE Software18, 1 (January/February 2001). 22-29.
- [9] Glass, Robert L. Facts and Fallacies of Software Engineering. Boston, MA: Addison-Wesley, 2003.
- [10] Hix, D., and Hartson, H.R. Developing User Interfaces: Ensuring Usability Through Product and Process. John Wiley and Sons, 1993.
- [11] IEEE. IEEE Std 610.12-1990. IEEE Standard Glossary of Software Engineering Terminology. IEEE, 1990.
- [12] IEEE. IEEE Std 1074-1997. IEEE Standard 1074 for Developing Software Life Cycle Processes. IEEE, 1998.
- [13] IEEE. IEEE Software. Usability Engineering. vol 18, 1 (January/February 2001). (entire issue).
- [14] IEEE Software Engineering Coordinating Committee. Guide to the Software Engineering Body of Knowledge - Trial Version 1.00. IEEE Computer Society, Los Alamitos, California, May 2001.
- [15] ISO/IEC. ISO/IEC International Standard: Information Technology. Software Life Cycle Processes, ISO/IEC Standard 12207:1995. ISO, 1995.
- [16] ISO. International Standard: Human-Centered Design Processes for Interactive Systems, ISO Standard 13407: 1999. ISO, 1999.
- [17] ISO/IEC. ISO/IEC International Standard: Information Technology. Software Life Cycle Processes. Amendment 1. ISO/IEC 12207:1995/Amd. 1:2002. ISO/IEC, 2002.
- [18] Jacobson, I., Christerson, M., Jonsson, P., Övergaard, G. Object-Oriented Software Engineering. A Use-Case Driven Approach. Revised Printing. ACM Press - Addison-Wesley, 1993.
- [19] Jacobson, I., Booch, G., Rumbaugh, J. The Unified Software Development Process. Reading (MA): Addison Wesley, 1999.
- [20] Kroll, P., Kruchten, P. The Rational Unified Process Made Easy. A Practitioner's Guide to the RUP. Addison-Wesley, 2003.
- [21] Larman, C.: UML y Patrones, Segunda Edición: Prentice-Hall, 2002.
- [22] Lim, K.Y., and Long, J. The MUSE Method for Usability Engineering. Cambridge University Press, 1994.
- [23] McCall, J.A., Richards, P.K., and Walters, G.F. "Factors in Software Quality," vol. 1, 2, and 3, AD/A-049-014/015/055, National Tech. Information Service, 1977.
- [24] Mayhew, D.J. The Usability Engineering Lifecycle. Morgan Kaufmann, 1999.
- [25] Nielsen, J. Usability Engineering. AP Professional, 1993.
- [26] D. A. Norman, The Design of Everyday Things. Doubleday, 1990.
- [27] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., and Carey, T. Human-Computer Interaction. Addison Wesley, 1994.
- [28] Rational Software Corp. What's New in Rational Development Accelerators™ Version 2002. 2002. http://www.rational.com/rda/wn_2002.jsp?SMSESION=NO
- [29] Royce, W. W. Managing the Development of Large Software Systems. Proc. of IEEE WESCON, 1970.
- [30] Rubin, J. Handbook of Usability Testing. John Wiley and Sons, 1994.
- [31] Seffah, A., and Andreevskaja, A. Empowering Software Engineers in Human-Centered Design. in Proceedings of the ICSE'03 Conference (Portland, Oregon - USA, May 2003), IEEE Computer Society, 653-658.
- [32] B. Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison-Wesley, 1998.