



## PRÁCTICAS DIRIGIDAS A LA MEJORA DE LA USABILIDAD DEL SOFTWARE

GABRIEL ALBERTO GARCÍA MIRELES\*,  
MA. ÁNGELES MORAGA DE LA RUBIA

*La usabilidad es un atributo del software (y sistema) que considera la experiencia de las personas cuando interactúan con las interfaces de un producto software. Una meta potencial de las organizaciones de desarrollo de software es mejorar la usabilidad de sus productos. Por otra parte, el software se desarrolla aplicando un proceso, el cual tiene un impacto significativo en la calidad final del producto, incluyendo su usabilidad. Teniendo esto en cuenta, en este trabajo se presenta un enfoque basado en el proceso de software en el cual nos interesa conocer las actividades, prácticas y métodos que podrían mejorar la usabilidad del software. Para ello, a partir de una revisión sistemática de literatura, identificamos los procesos de software que podrían ser complementados con prácticas de usabilidad. Adicionalmente, los métodos y técnicas de soporte a la usabilidad los relacionamos con cada proceso. Este es un primer paso en el desarrollo de una estrategia que permita introducir cambios en el proceso de software de cara a mejorar la usabilidad del producto.*

M. C. GABRIEL ALBERTO GARCÍA MIRELES  
Departamento de Matemáticas,  
Universidad de Sonora, México  
Correo: mireles@gauss.mat.uson.mx

DRA. MA. ÁNGELES MORAGA DE LA RUBIA  
Instituto de Tecnologías y Sistemas de Información,  
Universidad de Castilla – La Mancha, España  
Correo: MariaAngeles.Moraga@uclm.es

\*Autor para correspondencia: Gabriel Alberto García Mireles  
Correo electrónico: mireles@gauss.mat.uson.mx  
Recibido: 30 de septiembre de 2013  
Aceptado: 05 de mayo de 2014  
ISSN: 2007-4530

## INTRODUCCIÓN

La diversidad de sitios y aplicaciones web, redes sociales, software para dispositivos móviles, entre otros, llevan a nuevas formas de interactuar con el software. En muchas ocasiones, los resultados que se obtienen de dicha interacción condicionan la disposición del individuo a continuar usando un software particular. Dado que una interacción deficiente puede provocar el fracaso del producto software, las organizaciones que desarrollan software se están dando cuenta de la importancia que tiene la usabilidad para que el software sea aceptado por los destinatarios a los que está dirigido (1).

El diseño de sistemas usables tiene como beneficios una mejor aceptación del producto por los usuarios, promueve una respuesta positiva en los clientes, reduce el tiempo de capacitación e incrementa la productividad de los usuarios (2). De hecho, la usabilidad está relacionada con la experiencia que tiene una persona cuando usa un software particular para realizar una tarea específica. Esa experiencia se puede evaluar en términos de lograr el objetivo de la tarea, el uso eficiente de recursos y la actitud del usuario de cara a dicha experiencia (3). En consecuencia, el software deberá tener mecanismos apropiados de interacción según las tareas que realice el usuario.

El desarrollo de sistemas usables requiere de la aplicación de métodos y herramientas que se han estado desarrollando en dos campos: el Diseño Centrado en el Usuario (UCD) y la Ingeniería de Software (4). El primero

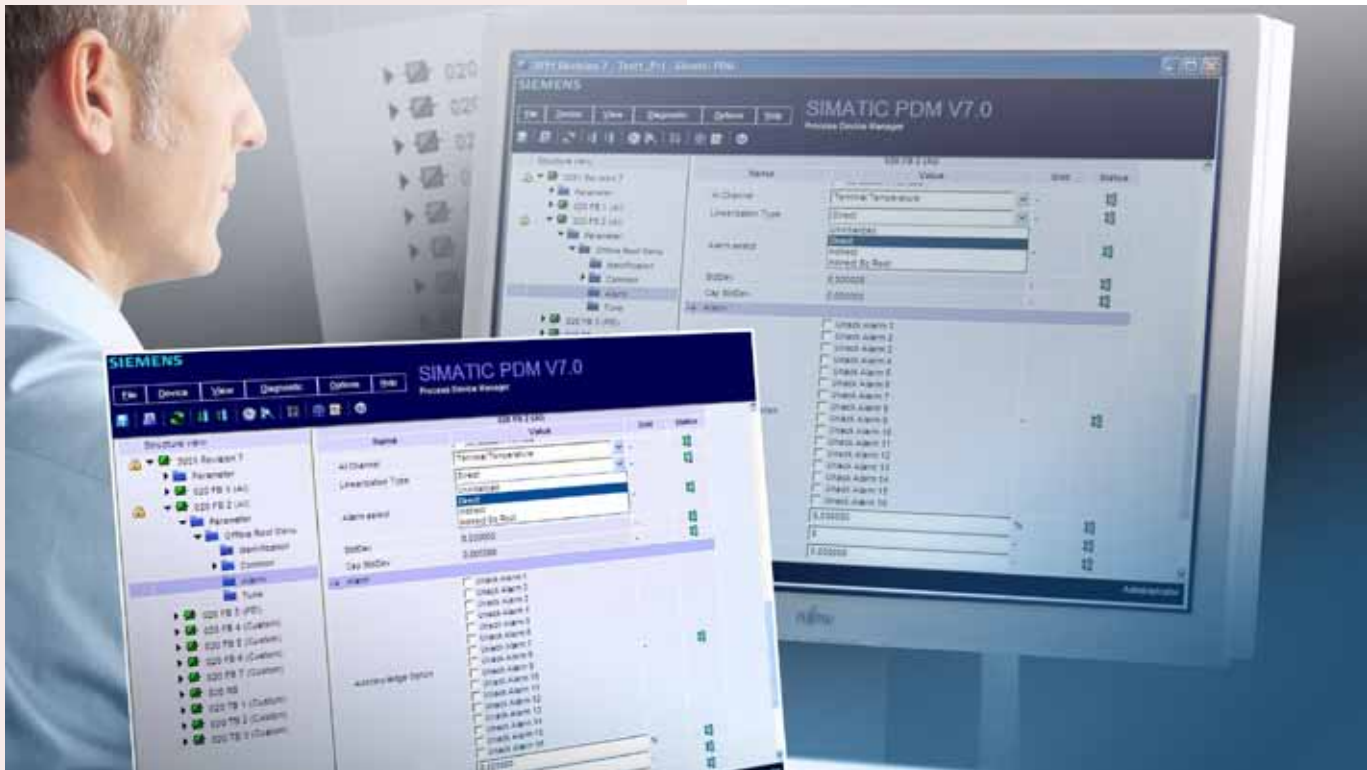


se enfoca en la perspectiva del usuario, en donde la meta es comprender el papel que desempeña cada uno de los distintos tipos de usuario en el contexto de operación del sistema y las tareas requeridas en ese contexto particular. El segundo se centra en el sistema y las funciones que debe proporcionar. Aunque hay diferencias en las perspectivas, ambas disciplinas comparten objetivos a la hora de desarrollar un sistema usable, por ejemplo, elegir las prácticas y métodos más apropiados que se usarán durante el desarrollo del producto software.

En una situación ideal, se esperaría que las organizaciones que desarrollan software interactivo cuenten con un equipo multidisciplinario de expertos. Sin embargo, la mayoría de las empresas que desarrollan software son pequeñas empresas, cuyo personal tiene, principalmente, estudios en las áreas de computación, y son ellos quienes se deben encargar del sistema completo (5).

La usabilidad de aplicaciones web, software para dispositivos móviles, y programas de juegos, entre otros, continuamente está bajo examen por los usuarios. Las experiencias negativas con el software podrían alejar a los usuarios. Dada esta situación, el objetivo de la investigación que realizamos es encontrar artículos científicos en los cuales se señale cómo mejorar la usabilidad del software considerando prácticas de usabilidad, es decir, actividades, prácticas y métodos que podrían usarse en el proceso de desarrollo enfocadas particularmente en esta característica de calidad.





La identificación de prácticas para mejorar la usabilidad del software que se presenta se basa en un estudio de literatura existente en el tema de combinación de actividades de UCD y de proceso de software. Se describe brevemente la metodología empleada y los principales resultados encontrados. Para armonizar los resultados con el enfoque de proceso de software, relacionamos las actividades y/o métodos sugeridos en la literatura revisada con cada uno de los procesos de desarrollo de software del ISO/IEC 12207 (6).

## MÉTODO

Con la finalidad de comprender cómo se ha abordado la usabilidad en el proceso de desarrollo de software, aplicamos un método que se ha propuesto en el campo de la ingeniería de software, denominado Revisión Sistemática de Literatura (RSL) que se diferencia de las revisiones convencionales por su rigurosidad (7). El método está dirigido principalmente a investigadores y tiene como meta identificar, evaluar e interpretar las evidencias, publicadas en artículos científicos, que se refieren a un tema o pregunta de investigación particular. Los artículos seleccionados que responden a la pregunta de investigación se denominan artículos primarios. El procedimiento para realizar las RSL contiene tres actividades principales: *planeación*, *ejecución de la revisión* y *generación de informes*.

Durante la *planeación* se determina con claridad el tema de investigación, se establece la estrategia de búsqueda y se especifican los criterios de selección de los

artículos primarios. La información generada se documenta en el protocolo de revisión, el cual guiará todo el proceso de revisión sistemática. En la tabla 1 se presentan las preguntas de investigación, los términos considerados para realizar la búsqueda en bases de datos de artículos científicos, así como algunos elementos que deberán considerarse en el artículo para decidir su inclusión en para nuestra investigación.

Tabla 1. Algunos elementos considerados en el protocolo de la RSL de proceso de software y usabilidad.

Elemento del protocolo	Aplicación
Preguntas de investigación	<ul style="list-style-type: none"> <li>• ¿Cuál es el estado de la investigación con respecto a la relación entre procesos de software y la usabilidad?</li> <li>• ¿Cuáles son los procesos y actividades que deberían considerarse para mejorar la usabilidad del software?</li> </ul>
Términos de búsqueda	<ul style="list-style-type: none"> <li>• Usabilidad</li> <li>• Producto o software</li> <li>• Proceso de software, proceso de desarrollo, método de desarrollo</li> </ul>
Criterios de inclusión de artículos	<ul style="list-style-type: none"> <li>• Artículo publicado en inglés</li> <li>• Artículo que describa el proceso y/o actividades orientadas a mejorar la calidad</li> <li>• Artículo publicados hasta el año 2010</li> </ul>

Tras la revisión del protocolo de revisión, se procede con la *ejecución de la revisión* de la literatura. Aquí se realizan las consultas en las bases de datos descritas en el protocolo, usando los términos de búsqueda. Esta actividad da lugar a un conjunto de artículos recuperados. En nuestra investigación, recuperamos 504 artículos de cinco bases de datos: Scopus (<http://www.scopus.com/>), Science@Direct (<http://www.sciencedirect.com/>), IEEE (<http://ieeexplore.ieee.org/xpl/periodicals.jsp>), ACM (<http://dl.acm.org/>) y SpringerLink (<http://link.springer.com/>).

El proceso de selección de artículos requiere, al menos, dos etapas. En la primera se revisa el título, resumen y palabras clave con el objetivo de determinar si el artículo satisface los criterios de inclusión especificados. Si es así, el artículo se selecciona como artículo candidato. En la segunda etapa, se lee el artículo completo y se determina si el trabajo contiene los criterios de calidad de investigación especificados, como los de rigurosidad metodológica y relevancia en el sector industrial. Si el artículo candidato cumple con los criterios establecidos, entonces se selecciona como primario. En nuestra investigación, identificamos 168 estudios candidatos y 18 primarios. La lista de artículos primarios puede consultarse en García-Mireles et al (8).

La figura 1 muestra la cantidad de artículos publicados por año. Como se observa, la mayor parte de contribuciones se realiza a partir de 2005. Los artículos publicados muestran que es un tema presente en la



investigación, aunque la cantidad limitada de ellos sugiere que la línea de investigación aún no se ha consolidado. Por tanto, esta área de investigación ofrece oportunidades para investigar cómo mejorar la usabilidad del software tomando en cuenta las prácticas que los ingenieros de software deberían aplicar al momento de desarrollar el producto.

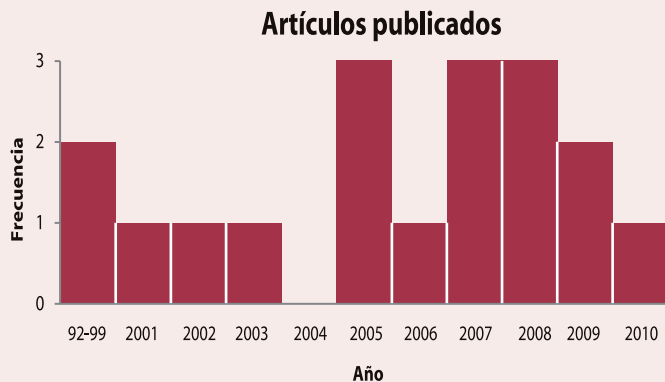


Figura 1. Artículos publicados por año.

La identificación de los artículos primarios permite la extracción de información relevante para contestar las preguntas de investigación. Para ello se elabora una plantilla en la cual se establecen los datos de interés. En nuestro caso consideramos los siguientes elementos: definición de usabilidad empleada, estándares usados en el proceso de desarrollo de software, actividades agregadas o modificadas del proceso de desarrollo que, según los autores del artículo, mejorará la usabilidad del software y el método de investigación empleado para validar la propuesta. Finalmente, se establece un enfoque de análisis de los datos recuperados en las plantillas, en el cual se clasifica y sintetiza la información con el objetivo de contestar las preguntas de investigación. En la siguiente sección se describirán las propuestas encontrados en los artículos evaluados.

Finalmente se realiza la actividad de *generación de informes* de la RSL, en donde se establece la estrategia de difusión de resultados de la investigación. El objetivo es presentar los resultados a las audiencias que pueden ser beneficiarias de esta información. La estrategia puede incluir revistas especializadas en el tema de investigación, revista de divulgación general, páginas web, carteles, y otros medios. En nuestro caso, la RSL fue publicada en una revista especializada del área de informática (García-Mireles et al. 2013).

### ANÁLISIS DE LOS ARTÍCULOS PRIMARIOS

Los artículos primarios pueden agruparse en cuatro temas principales (Figura 2), según el contexto en el cual fueron aplicadas las actividades que tienden a mejorar la usabilidad: procesos convencionales de desarrollo de software (33%), adaptación en procesos de desarrollo

específicos a dominios tecnológicos particulares (17%), enfoques de desarrollo ágil (22%) y mejora de la especificación de requisitos de calidad (28%). Usamos esta clasificación porque facilita el análisis de los artículos primarios.

### Clasificación de artículos primarios

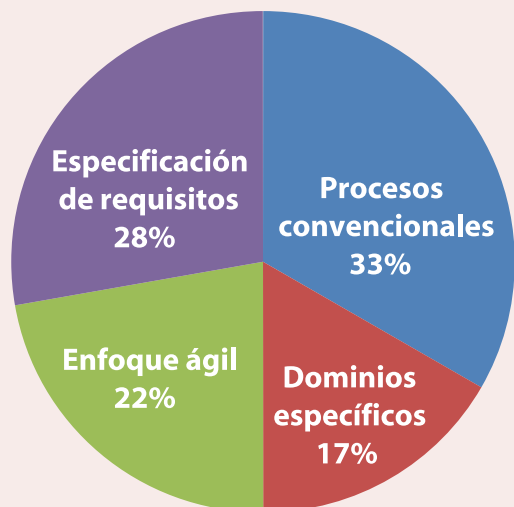


Figura 2. Porcentajes de clasificación.

### Procesos convencionales de desarrollo de software

En esta categoría se encuentran diversas propuestas que usan como referencias los ciclos de vida tradicionales

de desarrollo de software, donde estos se caracterizan por una planeación inicial de todo el proyecto de software y elaboración de productos (documentos o software) en cada una de las etapas metodológicas. El modelo al que se hace referencia en los artículos primarios es el Proceso Unificado. Las propuestas revisadas recomiendan involucrar al usuario, entender el contexto de uso del sistema, aplicar un enfoque de desarrollo iterativo y evaluar nuevos prototipos. Además, se sugiere incluir modelos de tarea, requisitos de usabilidad, patrones de diseño de usabilidad, y evaluación de prototipos para determinar la usabilidad de interfaces.

Algunas propuestas señalan la incorporación de roles específicos, como el ingeniero de usabilidad, el diseñador de interfaz de usuario y el examinador de usabilidad. Otras apuntan que en el proceso de desarrollo también deberían producirse productos como diseño del producto, modelos de usuarios, modelos de tarea, escenarios de uso, diseño de pantallas, prototipos de baja fidelidad (en papel) y prototipos de alta fidelidad (en software).

### Procesos orientados a dominios particulares

En esta categoría se encuentran las propuestas de proceso que consideran las características particulares de la tecnología que soporta al software. Algunas propuestas han señalado que uno de los principales requisitos es el establecimiento de metas de usabilidad y señalan que se deberían considerar la identificación de necesidades, establecer requisitos, elaborar el diseño conceptual, el diseño físico y la implementación. En el caso de aplicaciones web, se recomienda identificar el propósito





esencial, modelizar los roles, modelizar tareas, agrupar tareas, realizar el diseño preliminar de interfaces, modelizar la interacción, generar el prototipo, elaborar el diseño detallado y construir el software. Además, se recomienda entrenamiento especial y coordinación cercana entre diseñadores de interfaz de usuario y programadores.

### Propuesta orientadas a métodos ágiles

A diferencia de los métodos convencionales, en los métodos ágiles se recomiendan prácticas en las cuales se genere una versión operable (que pueda ser ejecutada en la computadora) del producto software desde las primeras semanas de desarrollo. Entre las prácticas recomendadas, se propone el modelado de usuario, modelado del diseño de interfaz por implementar y la evaluación de usabilidad. El cliente podría participar en la evaluación de prototipos de alta y baja fidelidad.

### Propuestas orientadas a los requisitos de calidad

En esta categoría se trata de obtener una especificación de requisitos de usabilidad más completa y exacta. Algunas propuestas sugieren incorporar aspectos de usabilidad y utilizar la evaluación de expertos (heurísticas) con base en guías de inspección de usabilidad. Otras propuestas recomiendan realizar las actividades de descripción del contexto, determinación de los escenarios de análisis, evaluación de escenarios y la interpretación de resultados. Finalmente, otros trabajos proponen la creación de perfiles de usuario y la participación activa de los usuarios durante el análisis de requisitos.

La clasificación y análisis de los artículos primarios arroja dos resultados importantes. Primero, si una organización que desarrolla software desea mejorar la usabilidad de sus productos necesita considerar prácticas específicas. Segundo, la identificación de las prácticas de usabilidad, proporciona un conjunto inicial de prácticas con la cual una organización de software podría iniciar un proyecto de mejora de su proceso de desarrollo de software.

## INTEGRACIÓN DE LAS ACTIVIDADES DE SOPORTE A LA USABILIDAD EN EL PROCESO DE DESARROLLO DE SOFTWARE

Para facilitar el uso de las prácticas de usabilidad, éstas se analizan considerando su propósito y etapa del ciclo de vida del desarrollo de software en la cual deberían aplicarse. Por otra parte, se requiere un conjunto de procesos aplicables en cada una de las etapas del ciclo de vida del software, con los cuales podamos establecer una asociación entre las prácticas de usabilidad y los procesos correspondientes. Una referencia de procesos consensuada y usada en el sector industrial es el estándar ISO/IEC 12207 (ISO 2008). Este estándar describe los procesos principales que una organización debería definir para mejorar la calidad del software que desarrolla.





Tabla 2. Relación entre los procesos de software y las actividades métodos y técnicas de soporte a la usabilidad.

Proceso	Actividad de usabilidad sugerida	Métodos y técnicas de soporte
<b>Implementación de software</b> Planeación del desarrollo de software	Planeación de tareas de usabilidad	Elaboración de calendario para ciclo de prototipos, planificación del diseño y asignación de recursos
<b>Análisis de requisitos de software</b> Establecer características de calidad. Verificación de requisitos	Diseñar modelos conceptuales	Casos de uso con escenarios alternativos, prototipos de baja fidelidad, modelo de dominio, evaluación de productos de la competencia para establecer metas de usabilidad, modelado de tareas de usuario, modelado conceptual y validación con usuarios representativos
<b>Diseño arquitectónico de software</b> Construir la estructura de alto nivel de software y verificarla	Estandarizar el diseño de interfaces de usuario	Elaborar modelo de navegación, presentación, prototipo de interfaz gráfica, usar patrón MVC para definir la arquitectura, modelo de datos, de componentes y de procesamiento/ modelo de interacción. Evaluaciones heurísticas y pruebas de usuario sobre prototipos
<b>Diseño detallado de software</b> Especificar y verificar las características de los componentes	Diseño detallado de interfaces	Determinar objetos visuales y patrones de la interfaz de usuario, prototipos de alta fidelidad, definir las características de elementos visuales y de interacción de las pantallas, elaborar diseño detallado para contextos de interacción seleccionados, validación con usuarios y expertos
<b>Construcción de software</b> Desarrollar, documentar y probar cada elemento de software	Construir las interfaces de usuario	Implementación del prototipo, presentación de contenidos en interfaz web, refactorización de código, uso de herramientas automáticas, uso de patrones de diseño y bibliotecas de componentes, reutilizar código y evaluación por expertos y usuarios
<b>Integración de software</b> Desarrollar y ejecutar un plan de integración y pruebas	Integrar interfaces de usuario	Integrar los componentes
<b>Prueba de calidad del software</b> Realizar pruebas de calidad con base en los requisitos especificados	Verificar la usabilidad del software	Evaluaciones heurísticas y caminatas, pruebas de usabilidad guiadas por expertos, establecer meta, método y criterio de evaluación, inspecciones de usabilidad colaborativas

La tabla 2 muestra la relación que establecimos entre los procesos del ISO/IEC 12207 dirigidos al desarrollo de software y las recomendaciones para mejorar la usabilidad del producto.

La primera columna de la tabla presenta el nombre de los procesos de software del estándar en negritas, seguido de una breve descripción. La segunda columna presenta el nombre de actividad sugerida y que podría servir de enlace con las actividades contenidas en los procesos de software. La tercera columna señala métodos y técnicas que podrían usarse para tomar en cuenta a la usabilidad en cada proceso. Por ejemplo, en el proceso de análisis de requisitos se incluyen una gran cantidad de técnicas para identificar al usuario, establecer el perfil del usuario, definir las tareas que realiza, especificar el contexto en el cual opera, realizar propuestas de diseño de interfaz y su correspondiente validación para definir los requisitos de usabilidad.

## CONCLUSIONES Y TRABAJO FUTURO

La usabilidad es un atributo del software que las organizaciones deberían considerar cuando implementan sistema interactivos. Los resultados de la revisión sistemática de literatura permiten identificar las prácticas de usabilidad que podrían ser aplicadas durante el desarrollo del software. Además, el análisis de las prácticas de usabilidad permitió identificar aquellos procesos, descritos en el estándar ISO/IEC 12207, que requieren agregar o modificar actividades para mejorar la usabilidad del producto software.

La mejora de la usabilidad del software es un proceso iterativo, en el cual en diferentes tiempos del proyecto de desarrollo del producto, los ingenieros de software pueden visitar a los usuarios para determinar los requisitos de usabilidad, validar interfaces y comprobar que el software efectivamente mejoró la usabilidad.

A partir de este trabajo podrían desarrollarse un conjunto de metodologías y procesos de soporte a la usabilidad del software. Por ejemplo, podría desarrollarse un método de evaluación cuya meta sea determinar el grado en el cual las organizaciones de desarrollo de software implementan prácticas de usabilidad. Otra aplicación de estos resultados es el desarrollo de un método de adaptación del proceso de desarrollo de software para que incluya prácticas de usabilidad. Si consideramos, adicionalmente, mediciones de usabilidad aplicadas tanto al software como a las prácticas sugeridas, podrían desarrollarse métodos para determinar el grado de satisfacción de los requisitos de usabilidad y la eficiencia de las prácticas de usabilidad implementadas. Además, estas futuras aportaciones deberían ser validadas a través de estudios empíricos en organizaciones que desarrollan software.

## AGRADECIMIENTOS

Este trabajo ha sido realizado gracias al financiamiento del proyecto GEODAS-BC (Ministerio de Economía y Competitividad y Fondo Europeo de Desarrollo Regional FEDER, TIN2012-37493-C03-01).

## BIBLIOGRAFÍA

- 1) Berntsson Svensson, R., Gorschek, T., Regnell, B., Torkar, R., Shahrokni, A. and Feldt, R. (2012). Quality requirements in industrial practice-an extended interview study at eleven companies. *IEEE Transactions on Software Engineering* 38(4): 923-935.
- 2) Maguire, M. (2001). Methods to support human-centred design. *International Journal of Human Computer Studies* 55(4): 587-634.
- 3) Bevan, N. (2010). Los nuevos modelos de ISO para la calidad y la calidad en uso del software. *Calidad del producto y proceso software*. C. Calero, M. Á. Moraga and M. Piattini, Ra-Ma: 55-77.
- 4) Seffah, A., Desmarais, M. and Metzker, E. (2005). HCI, *Usability and Software Engineering Integration: Present and Future. Human-Centered Software Engineering — Integrating Usability in the Software Development Lifecycle*. A. Seffah, J. Gulliksen and M. Desmarais, Springer Netherlands. 8: 37-57.
- 5) Seffah, A. and Metzker, E. (2004). The obstacles and myths of usability and software engineering. *Comm. Of the ACM* 47(12): 71-76.
- 6) ISO (2008). ISO/IEC 12207 Systems and software engineering — Software life cycle processes
- 7) Kitchenham, B. and Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. *Technical Report EBSE/EPIC-2007-01*.
- 8) García-Mireles, G. A., Moraga, M.A., García, F. and Piattini, M. (2013). The Influence of Process Quality on Product Usability: A Systematic Review. *CLEI Electronic Journal* 16(2): 1-13 en: <http://www.clei.org/cleiej/paper.php?id=278..>

