

Propuesta de Incorporación Temprana de Usabilidad en el Modelado Conceptual de Sistemas

Proposed Early Incorporation of Usability in the Conceptual Modeling of Systems.

Presentación: 11/10/2019

Doctorando:

Juan Carlos Moreno

Centro de Investigación, Desarrollo y Transferencia de Sistemas de Información (CIDS)
Universidad Tecnológica Nacional – Facultad Regional Córdoba
Jmoreno33@gmail.com

Director/es:

Dr. Marcelo Martín Marciszack – Dr. Mario Alberto Groppo

Resumen

La industria del software actual le exige a la Ingeniería de Sistemas el desarrollo de métodos para la construcción veloz, correcta y sobre todo de aplicaciones usables para los sistemas de información. En el presente trabajo, se presenta una propuesta metodológica que permite definir y mantener atributos de usabilidad, integrándolos al ciclo de vida de desarrollo del software. En etapas tempranas de desarrollo del software, los atributos de usabilidad son identificados, partiendo del Modelado de Procesos de Negocios. Luego se realiza un conjunto de diferentes transformaciones para integrarlos a una estructura denominada “Requirements Baseline”, que emplea el léxico extendido del lenguaje (LEL) y escenarios, donde se permite definir métricas de evaluación de los atributos y la relación que existe entre los diferentes atributos y subatributos, tomando como fundamento la estructura conceptual propuesta por la norma ISO/IEC 25000 (SQUARE). Los subatributos permitirán definir métricas que se emplean posteriormente para evaluar distintos criterios de usabilidad relacionados con la calidad del producto. Además, permite incorporar nuevas especificaciones de usabilidad y sus correspondientes métricas de evaluación en forma integrada al proceso de desarrollo de software, lo que le otorga cierto grado de trazabilidad a todo el proceso.

Palabras claves: Usabilidad, MDA, Modelado Conceptual, Escenarios, Patrones.

Abstract

The current software industry requires Systems Engineering to develop methods for rapid, correct construction and, above all, for usable applications for information systems. In this paper, a methodological proposal is presented that allows defining and maintaining usability attributes, integrating them into the software development life cycle. In the early stages of software development, usability attributes are identified, based on Business Process Modeling. Then a set of different transformations is carried out to integrate them into a structure called “Requirements Baseline”, which uses the extended language lexicon (LEL) and scenarios, where it is possible to define metrics for evaluating the attributes and the relationship that exists between the different attributes and sub-attributes, based on the conceptual structure proposed by ISO / IEC 25000 (SQUARE). Sub-attributes will allow metrics to be defined that are subsequently used to evaluate different usability criteria related to product quality. In addition, it allows incorporating new usability specifications and their corresponding evaluation metrics in an integrated way to the software development process, which gives a certain degree of traceability to the entire process.

Keywords: Usability, MDA, Conceptual Modeling, Scenarios, Patterns.

1. Introducción

La norma ISO/IEC 9126-1 (2001), define a la usabilidad como “la capacidad en que un producto de software puede ser entendido, aprendido y usado por determinados usuarios bajo ciertas condiciones en un contexto de uso específico”. Para estudiar la usabilidad, los criterios abstractos que la definen se descomponen en subatributos medibles (Piattini et al., 2007). La Ingeniería de Sistemas debe construir software de calidad cuyos resultados deben ser útiles a sus usuarios. El problema se suscita porque la calidad de las aplicaciones se mide generalmente basándose en el sentido común del desarrollador (Abraham et al., 2004), quien hace énfasis en aspectos de arquitectura, funcionalidad y persistencia de cada proceso, y no trata adecuadamente la facilidad de uso.

1.1 Hipótesis

Partiendo del contexto analizado, se puede establecer qué: “la construcción de un nuevo marco metodológico conceptual, que incorpore criterios de usabilidad en etapas tempranas del ciclo de vida del software, permite desarrollar software de calidad y optimizar los procesos”.

1.2 Objetivo general

Proponer un marco metodológico para la construcción de modelos conceptuales, que permitan incorporar y evaluar la usabilidad en forma temprana a través de procesos estandarizados, empleando patrones para incorporar la usabilidad, y modelos abstractos para verificar la presencia de la misma.

2. Metodología y Herramientas

Generalmente, la usabilidad es considerada en etapas finales de la construcción del software, donde una modificación afecta la arquitectura y su costo es alto (Bass et al., 2003), (Folmer et al., 2004). Una solución posible sería incluir el análisis de la usabilidad en etapas tempranas de construcción del modelo conceptual, empleando el paradigma de Desarrollo de Software Dirigido por Modelos (MDA) (Mellor et al., 2002), (Miller et al., 2003), y posteriormente modelos abstractos para su verificación. Ver Figura 1.

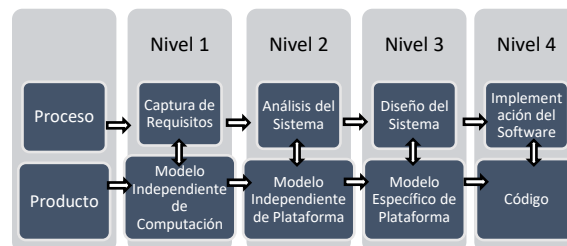


Figura 1. Esquema de Proceso de Model-Driven Architecture (MDA)

Para poder aplicar el esquema conceptual propuesto, planteado como una estrategia en etapas, es necesario tener conocimiento sobre conceptos relacionados con: Modelado de Procesos de Negocios (BPM), Desarrollo de Software Dirigido por Modelos, Escenarios, Patrones y Modelos Abstractos. Esto le permitirá al lector comprender mejor como se diseña y especifica cada etapa, los elementos y artefactos construidos relacionados al modelo conceptual del sistema y como incorporar criterios de calidad (Usabilidad) al modelo.

2.1 Estrategia Propuesta

La estrategia propuesta, que aquí se presenta, se conforma de tres etapas con distintas actividades que se llevan a cabo en cada una de las mismas (ver Tabla 1). Posteriormente se desarrollará un ejemplo práctico:

Tabla 1. Especificación de la Propuesta Metodológica

Proceso	Actividades
a) Etapa 1	1) Diseñar el modelo de negocio con “Business Process Model and Notation” (BPMN) o en español “Modelo y Notación de Procesos de Negocio” (OMG). 2) Indicar qué actividades están vinculadas con la gestión de información. Incorporar estereotipos (Reglas de Negocio) a aquellas actividades automatizadas con Requerimientos No Funcionales (R.N.F.) de Usabilidad.
b) Etapa 2	3) Construir los Escenarios a partir de las actividades y el diccionario de datos, empleando Léxico Extendido del Lenguaje.
c) Etapa 3	4) Trasformar los Procesos de Negocios y Escenarios en máquinas abstractas. 5) Verificar la consistencia de los requerimientos y de las definiciones de atributos de Usabilidad a través de Redes de Petri Coloreadas y Autómatas Finitos resultantes del proceso anterior.

a) La primera etapa comienza con el modelado de negocio en BPMN (OMG), en donde las actividades resultantes serán mapeadas dentro de una estructura denominada Requirements Baseline (do Prado Leite et al, 1997), propuesta por Leite, que permite representar el lenguaje de la aplicación mediante un conjunto de símbolos y su comportamiento, en un momento específico, a través de escenarios.

Para realizar el modelado se deberá cumplir lo siguiente:

- Cada especificación de usabilidad deberá definirse como una actividad con estereotipo “Regla de Negocio”, y asociarse a la actividad del modelo de negocio que deba satisfacer la especificación.
- Las especificaciones deberán definirse en términos de palabras reservadas que ya se encuentren contenidas en el vocabulario del LEL, cada palabra reservada corresponderá a un atributo de usabilidad de bajo nivel que contendrá su definición y métrica asociada.
- Dado el caso que la especificación no se encuentre definida como atributo de bajo nivel dentro del LEL, deberá definirla el especialista como se irá describiendo en la segunda y tercera etapa.
- Todas las actividades serán mapeadas a la Requirements Baseline como escenarios salvo las definidas con el estereotipo “Manual”.
- Las actividades de “Regla de Negocio” asociadas a una actividad con el estereotipo “Manual” no serán mapeadas.

Para el modelado de negocio se utilizará la herramienta Bizagi. El modelo resultante se exportará en formato XPD L para ser utilizada en la siguiente etapa.

b) En la segunda etapa se utilizará la herramienta Baseline Mentor Workbench (BMW), que contiene todas las funcionalidades necesarias para introducir en la Requirements Baseline las definiciones contenidas en el archivo XPD L obtenido en la etapa anterior. Como se verá más adelante en un ejemplo de aplicación de la metodología, las especificaciones de usabilidad serán agregadas como restricciones en los escenarios y deberán definirse reglas en la herramienta BMW que permitan asociarlas a un atributo en función de si se encuentra definido o no dentro del vocabulario del LEL.

c) Finalmente, en la tercera etapa del proceso es donde se deben definir dentro del LEL los atributos de usabilidad y las métricas con la que se evaluarán para luego ser utilizados como palabra reservada en el proceso de modelado de la primera etapa. Cada atributo que se defina deberá asociarse a una sub-característica, y ésta a su vez, con una característica de alto nivel, estableciendo relaciones jerárquicas entre sí formando un árbol de requerimientos de calidad. La descomposición de características en sub-características estará precargada en el LEL y se basará en la estructura conceptual propuesta en ISO/IEC 25000 (SQUARE) (2005). También se ofrecerá un conjunto de atributos precargados con métricas definidas, ofreciendo la posibilidad al especialista de modificarlos, o bien, de generar nuevos atributos con sus métricas y asociaciones. Las métricas definidas en los atributos serán utilizadas posteriormente para evaluar la calidad del producto. Las etapas del proceso pueden observarse en la Figura 2.

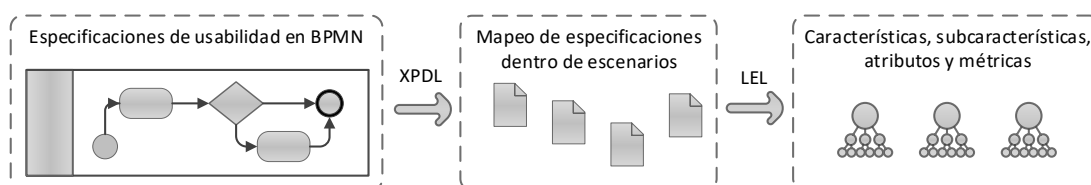


Figura 2. Etapas del proceso de la metodología.

3. Descripción del proceso metodológico con un ejemplo.

3.1 Diseñar el Modelado de Negocios con BPMN.

En BPMN, los Procesos de Negocio se construyen como una secuencia ordenada de actividades y de información que emplea el proceso, para representar cómo una organización realiza sus objetivos centrales del negocio. BPMN es gráficamente más rico, con menos símbolos, fundamentales lo que facilita su comprensión por parte de personas no expertas. A continuación, se plantea un ejemplo de un proceso de producción (ver Figura 3). Teniendo en cuenta las reglas de modelado descritas para la primera etapa se mapearán todas las actividades exceptuando aquellas que tengan el estereotipo “Manual”.

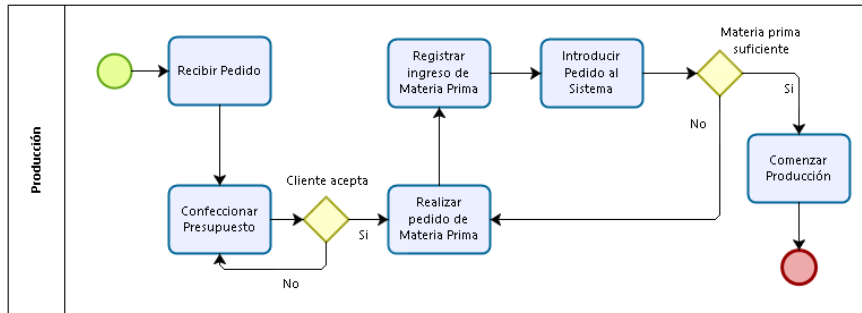


Figura 3. Ejemplo de un Modelo de Proceso de Negocio de Producción modelado en BPMN.

3.2 Indicar qué actividades tienen manejo de información.

En la Figura 4, se muestra cómo se procede para seleccionar las actividades de negocio que son automatizadas y formarán parte del Sistema de Información. Las especificaciones de usabilidad deberán agregarse como actividades utilizando el estereotipo “Regla de Negocio” y la descripción a utilizar deberá corresponder a un atributo cargados previamente en el LEL. Para el caso donde la especificación no se encuentre cargada como atributo en el LEL, la descripción de la actividad será utilizada para generar una nueva entrada que posteriormente deberá definirse. Una vez definida estará disponible para ser usada como palabra reservada. Aquí, el analista deberá identificar en los diagramas de procesos aquellas actividades que utilicen / generen información, diferenciándolas de aquellas que son puramente manuales. Las especificaciones de usabilidad deberán agregarse como actividades utilizando el estereotipo “Regla de Negocio”. En la Figura 4, también se muestra un ejemplo de la actividad “Registrar ingreso de materia prima”, a la que se le asociaron los atributos “Completado de las tareas” y “Exactitud de las tareas” correspondientes a la subcaracterística “Rendimiento de las tareas del usuario”. El proceso completo puede observarse a continuación.

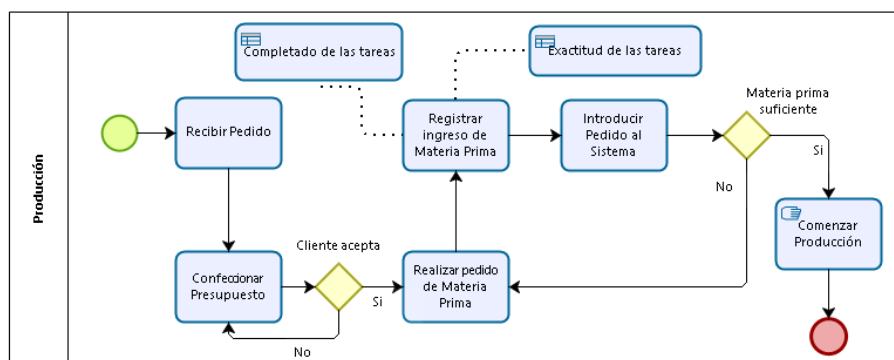


Figura 4. Modelado de atributos de usabilidad en BPMN para actividades de un proceso de producción.

Una vez finalizado el modelado se debe exportar el modelo en formato XPD, para luego ser importado dentro de la herramienta BMW.

3.3 Construir los escenarios a partir de las actividades de negocio.

En este paso, utilizando como guía las actividades marcadas como automatizadas (no manuales) en el punto anterior, es preciso identificar los Casos de Uso del sistema que darían soporte "informático" a las actividades.

Habiendo finalizado el modelado del sistema de negocio y cumplimentado con todos los aspectos a considerar especificados en las dos etapas anteriores, se procede a aplicar transformaciones mediante un proceso automatizado, donde las actividades identificadas como de soporte del sistema de información se convierten en escenarios.

Una vez finalizado el modelado del sistema de negocios se debe exportar el modelo en formato XPD desde Bizagi para luego ser importado dentro de la herramienta BMW. La herramienta verificara todas las actividades generando un escenario para cada una de ellas exceptuando las del tipo “Manual”. Para las del tipo “Regla de negocio” buscará una asociación dentro del vocabulario de LEL que corresponda a un atributo de usabilidad ya definido, una vez encontrada la asociación se agregará como restricción al escenario. En caso de no encontrar ninguna asociación con atributos contenidos en del LEL se creará automáticamente una entrada que deberá definir el diseñador del sistema. Las entradas de LEL son denominadas símbolos y se definen por medio de nociones e impactos que indican la repercusión que tendrá el símbolo en el sistema.

En la Figura 5 puede observarse el escenario creado para la actividad “Registrar ingreso de materia prima”, y las asociaciones de los atributos de usabilidad “Completado de las tareas” y “Exactitud de las tareas” agregadas como restricciones. Las descripciones restantes de los escenarios deberán ser completadas en forma manual por el analista.

Figura 5. Plantilla de LEL para la definición de escenarios.

3.4 Transformar los Procesos de Negocios y Escenarios en Autómata Finito.

Las máquinas abstractas se emplean en el modelado de sistemas en diversas áreas. En este caso, se emplea un tipo máquina abstracta que son los Autómatas Finitos. Una vez que se cuenta con una especificación detallada de los Procesos de Negocios y los escenarios que satisfacen las necesidades del negocio, se realiza nuevamente una transformación de los mismos a máquinas abstractas. Cada una de las actividades de Negocio identificadas tendrá un mapeo directo con cada estado identificado del Autómata finito. Lo mismo ocurrirá con los estados de Inicio y de finalización, ya sea por el éxito del procedimiento o por el fracaso del mismo. Los arcos del autómata finito surgirán a través de los flujos de trabajo que vinculan las Actividades del proceso de Negocio.

3.5.1 Verificar la consistencia de las definiciones a través de los Autómatas Finitos resultantes.

Al disponer de la representación de cada proceso de negocio y de los escenarios, se obtiene un Autómata Finito, a partir de una transformación directa de la representación del modelo que debe ser validado, se verifican los modelos los cuales deberán satisfacer ciertos requisitos y condiciones, que se detallan a continuación: a) *Conjunto Conexo y accesibilidad de estados*: si un estado definido en el AF no es accesible desde el estado inicial, significa que el modelo que está siendo representado por el autómata no está correctamente planteado; b) *Autómata Finito Determinista*: La definición de los modelos de procesos puede resultar en procesos paralelos o simultáneos, se traducen en no determinismo. Se puede convertir el AF No Determinista en uno Determinista equivalente, de manera de brindar al analista la posibilidad de analizar si se reformula el modelo o se mantiene tal como está definido; c) *Minimización del Autómata Finito*: un AF no minimizado puede significar la presencia de estados equivalentes, los cuales mediante los autómatas pueden ser identificados y reemplazados. Esto contribuye a simplificar el Modelo que representa al Proceso de Negocio; d) *Simulación de Ejecución de Autómatas Finitos*: En cada modelo de proceso de Negocio y su representación mediante un Autómata Finito, se pueden simular diversas entradas, verificando si se producen los resultados esperados por el modelo. Todo esto permite determinar si en los modelos hay procesos con actividades que no se realizan o procesos con actividades irrelevantes o innecesarias.

3.5.2 Verificar la consistencia de las definiciones de las interfaces a través de Redes de Petri Coloreadas.

Se realiza una breve descripción del método empleado para modelar interfaces, mediante grafos. Para ello, se emplearán Las redes de Petri, que se las puede definir como un lenguaje gráfico, con un alto nivel de formalidad para la descripción y simulación de procesos. Se emplean generalmente para modelar sistemas distribuidos y concurrentes. Estos modelos son muy empleadas para el análisis de sistemas dinámicos discretos.

El análisis del comportamiento de los estados del grafo, empleando simulación, permite saber si es posible alcanzar una configuración a partir de otra (alcanzar cada estado del grafo), los estados que permiten alcanzar a otros, y si existen estados inaccesibles o inalcanzables dentro de la red diseñada.

Las redes de Petri coloreadas tienen la particularidad que permiten agregar datos estructurados a los tokens, cuyo valor es a lo que se le denomina color. El mismo es importante porque permite que pueda ser seleccionado por los arcos, evaluado en las transiciones, y cambiado por transición a otro estado. Además, brinda la ventaja de poder evaluar subredes, pertenecientes a diseños de redes mayores. Estos modelos de redes serán empleados para validar las interfaces o pantallas de modelos abstractos.

4. Resultados

Partiendo del modelado de negocio de un dominio de ejemplo fue posible utilizar el Léxico Extendido del Lenguaje para definir características, subcaracterísticas, y métricas de usabilidad entre distintos modelos, estableciendo relaciones entre los mismos, garantizando cierto grado de trazabilidad. Además, se crea un mecanismo flexible que permita al especialista modificar especificaciones de usabilidad precargadas en el LEL y crear nuevas conforme avanza el ciclo de desarrollo. En función de los resultados obtenidos, se puede crear un procedimiento sistematizado como una metodología, que permita detectar, definir, mantener e incorporar en forma flexible y escalable: características, subcaracterísticas, atributos y métricas de usabilidad asociadas, basados en los estándares vigentes de calidad.

Se continuará trabajando sobre la metodología para incorporar otros estándares de calidad de software disponibles y poder establecer ventajas y desventajas de su aplicación. Si bien todos estos procesos son realizados y verificados por distintas herramientas, se plantea la posibilidad de diseñar un software con los procesos descriptos integrados.

5. Conclusiones

La metodología planteada en este trabajo permite al diseñador definir especificaciones de usabilidad en etapas tempranas del desarrollo de software en forma organizada, flexible, escalable y acorde a estándares de calidad vigentes. Esto queda demostrado a través del desarrollo metodológico propuesto como proceso donde se emplean metodologías de Modelado de Negocios y de Escenarios, vinculadas a través del uso de conceptos y prácticas del paradigma del Desarrollo de Software Dirigido por Modelos. A través de esta propuesta metodológica y sus herramientas de soporte, que se sintetizan como un conjunto de transformaciones aplicadas sobre el modelo conceptual primario, es posible generar nuevos modelos que sirvan para representar las máquinas abstractas necesarias para la verificación y validación de los requerimientos funcionales iniciales como así también de sus interfaces, garantizando de esta forma que los modelos reflejen fielmente la realidad, sin ambigüedades, manteniendo la coherencia y asegurando la trazabilidad a lo largo de todo el proceso de gestión de requerimientos.

Estas validaciones y simulaciones a las Máquinas Abstractas generadas a través de un proceso automatizado de transformaciones, ya sean sobre Procesos de Negocios, o Escenarios, nos permiten confirmar las características deseables sobre las especificaciones de los requisitos funcionales del sistema a construir. De este modo, el proceso de verificación y validación propuestos resultan útiles durante la etapa de diseño el modelo conceptual y de sus interfaces, para posteriormente poder construir el sistema de software que será soporte del sistema de información.

6. Referencias.

Abrahão, S., Condori-Fernández, N., Olsina, L., & Pastor, O. (2004, September). Defining and validating metrics for navigational models. In *Proceedings. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry (IEEE Cat. No. 03EX717)* (pp. 200-210). IEEE.

Bass, L., & John, B. E. (2003). Linking usability to software architecture patterns through general scenarios. *Journal of Systems and Software*, 66(3), 187-197.

do Prado Leite, J. C. S., Rossi, G., Balaguer, F., Maiorana, V., Kaplan, G., Hadad, G., & Oliveros, A. (1997). Enhancing a requirements baseline with scenarios. *Requirements Engineering*, 2(4), 184-198.

Folmer, E., & Bosch, J. (2004). Architecting for usability: a survey. *Journal of systems and software*, 70(1-2), 61-78.

ISO/IEC 25000:2005. Software Engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE.

Mellor, S. J., Scott, K., Uhl, A., & Weise, D. (2002, September). Model-driven architecture. In *International Conference on Object-Oriented Information Systems* (pp. 290-297). Springer, Berlin, Heidelberg.

Miller, J., Mukerji, J. "MDA Guide Version 1.0" (2003), http://www.omg.org/mda/mda_files/MDA_Guide_Version1-0.pdf, Consultado en junio de 2019.

Norma ISO/IEC ISO9126-1, "Software Engineering -Product Quality - Part 1," 2001.

Object Management Group. Business Process Modeling Notation (BPMN).

Piattini Velthuis, M. G., Garcia Rubio, F., & Caballero Muñoz-Reja, I. (2007). Calidad de sistemas informáticos (No. 004.05). Alfaomega Ra-Ma.