

REASEM: Herramienta para la gestión de requisitos

REASEM: Requirement management tool

Jorge Eduardo Gómez Maldonado¹, Ing., German Urrego Giraldo², PhD. & Liliana González Palacio³, MSc

1. Ingeniero de Sistemas, Universidad de Antioquia

2. Profesor de la Universidad de Antioquia

3. Docente tiempo completo Universidad de Medellín

koled2@yahoo.com; gaurrego@udea.edu.co; ligonzalez@udem.edu.co

Recibido para revisión 26 de Noviembre de 2008, aceptado 25 de Agosto de 2009, versión final 2 de Septiembre de 2009

Resumen— El proyecto del cual se deriva este artículo está enmarcado en dos áreas principales: la Ingeniería de Requisitos (IR) y los Sistemas Embebidos (SE), y tiene como objetivo el desarrollo de una herramienta que apoye la metodología ABC-Besoins-SEM, concebida para la gestión de requisitos de Sistemas Embebidos (SE), en la perspectiva de la conceptualización de sistemas ubicuos, con uso intensivo de componentes de hardware/software.

Los aplicativos de IR existentes tienen algunas carencias, tales como: no ofrecen flexibilidad para permitir cambios de acuerdo al modelo de requisitos que plantea cada metodología; no soportan la fase de elaboración del modelo conceptual y su posterior conversión a un modelo que guíe el diseño. Adicional a lo anterior, no se conocen herramientas centradas en la gestión de requisitos en el campo de sistemas embebidos, ni en su extensión hacia los sistemas ubicuos.

En este artículo se presenta REASEM, herramienta para la gestión de requisitos. Esta aplicación contiene las características básicas para apoyar las fases de la Ingeniería de Requisitos, es decir: permite la identificación, clasificación y documentación de requisitos. Adicionalmente permite la generación de un modelo conceptual y su transformación en un modelo de diseño.

Palabras Clave— Sistemas Embebidos, Ingeniería de Requisitos, Metodología ABC-Besoins-SEM, Herramientas de Gestión de Requisitos.

Abstract— This Article is derived from a project which contains two topics: Requirements Engineering (RE), and Embedded Systems (ES). The objective of the article aims to develop a tool for supporting the ABC-Besoins-SEM Methodology, conceived for the Requirements management of Embedded Systems (ES) in the perspective of ubiquitous system conceptualization, using intensively hardware/software components.

At present, RE applications lack of special features, such as: a) flexibility for changing or customizing according to needs of

different methodologies: b) to support the initial phase of conceptual model, and its conversion to a model to guide the design.

Additionally, it is not known the existence of tools for supporting the process requirements nor in embedded systems neither in ubiquitous systems.

This article is presented REASEM, requirements management tool. REASEM contains the basic features for supporting the stages of Requirements Engineering, that is, the tool lets the identification, classification and documentation of requirements. Furthermore, REASEM lets to create a conceptual model and its transformation to design model.

Keywords— Embedded systems, Requirements Engineering, ABC-Besoins-SEM methodology, requirements management tool.

I. INTRODUCCIÓN

La Ingeniería de Requisitos comprende las etapas fundamentales en el desarrollo de los productos de la ingeniería de software y se centra en descubrir, analizar, escribir y verificar los servicios y restricciones del sistema. Su importancia radica en que, de la definición de los requisitos depende el éxito de las etapas posteriores del desarrollo [36]. Si los requisitos no se descubren o son encontrados en una etapa avanzada del desarrollo, esto provocará atrasos en el cronograma, aumento en el presupuesto, y el riesgo de que el producto no satisfaga las necesidades del cliente [15], [19]. De ahí la necesidad de proponer metodologías que permitan la captura y tratamiento de requisitos de una manera sistemática, oportuna y confiable soportadas por herramientas adecuadas.

La parte restante de este artículo se estructura de la manera siguiente: en la sección 2 se hace una breve descripción de los

términos que serán usados en la descripción de solución. En la sección 3 se plantea la situación problemática a resolver. La solución propuesta se incluye en la sección 4. Los resultados hacen parte de la sección 5. El proceso de validación se enuncia en la sección 6. Las tres últimas secciones corresponden a las conclusiones, trabajos futuros y bibliografía.

II. DEFINICIÓN DE LOS ELEMENTOS DE LA SOLUCIÓN

A continuación se presentan las definiciones fundamentales utilizadas en la descripción de la situación problemática y su solución:

- Requisito: Es una necesidad, funcionalidad o restricción que debe satisfacer un sistema [27].
- Ingeniería de Requisitos (IR): Es una rama de la Ingeniería de Software que apoya al analista de sistemas en su tarea de traducir las necesidades de los agentes interesados a funciones y restricciones de un sistema para garantizar resultados de buena calidad [36]. Comprende dos fases: definición y análisis. [19].
- Modelo de requisitos: Representación de los elementos utilizados para la obtención y representación de requisitos. Generalmente está referido a una taxonomía [18].
- Modelo conceptual: Es la representación gráfica de las funcionalidades de un sistema. Constituye el modelo lógico de dicho sistema. Es el producto final de la fase de análisis [32], [33]. Un modelo conceptual puede tener diferentes niveles de abstracción de acuerdo al grado de detalle que se desee [17], [21].
- Modelo de diseño: Representación arquitectural del sistema incluyendo detalles de la plataforma de desarrollo y detalles de implementación [12], [24].
- Herramienta de gestión de requisitos: Aplicación de soporte a la realización de algunas tareas de la Ingeniería de Requisitos [8].
- Sistema Embebido (SE): Es un sistema de procesamiento de información de uso específico, conformado por componentes hardware y software e integrado en otro sistema de mayor tamaño. [16], [23].
- SystemC: Es un lenguaje de especificación de sistemas embebidos, basado en C++, que soporta la codificación en diversos niveles de abstracción. Entre las ventajas de SystemC se encuentra el establecimiento de un ambiente de diseño común para el software y el hardware consistente en un conjunto de librerías de C++ [20].

III. PROBLEMÁTICA

A menudo los diseñadores de sistemas cometen el error de comenzar a diseñar e implementar soluciones que no han sido completamente especificadas y que corresponden a problemas a los que les falta delimitación, lo cual conduce a la construcción

de sistemas que no satisfacen las necesidades de los clientes y que incurren en el aumento de los costos y en el incumplimiento de los plazos establecidos [9], [35]. Todo lo anterior refleja las carencias que existen en cuanto a la definición de requisitos como se describe en las estadísticas del Standish Group [28], [29] y en la literatura de Ingeniería de Requisitos desde los trabajos pioneros de Ross [27].

Esta problemática sigue existiendo tal como se ha mostrado en diferentes trabajos desde hace tres décadas [9], [36]. En el campo de los sistemas embebidos tratados en [4], [5], [16] dicha problemática general de los sistemas de información, no solo se conserva sino que se agrava hasta el punto que el 60% de las componentes que integran hardware (HW) y Software (SW) deben ser rediseñadas luego de haber sido programadas [20]. La problemática expuesta para los sistemas en general tiene mayor impacto en el caso de los sistemas embebidos [5], [11], [20].

Algunas de las dificultades adicionales que se presentan en el caso de los sistemas embebidos surgen generalmente por las razones siguientes [3], [4], [16]: a) Estos sistemas son construidos por expertos en electrónica, pero alejados del uso de las metodologías de análisis y diseño de sistemas. b) El uso de metodologías de propósito general que no se ajustan completamente a las características de los sistemas embebidos. c) Las metodologías existentes en el campo de los sistemas embebidos no cubren todas las fases de la IR, centrándose en la fase de diseño con poco apoyo a las fases previas del ciclo de vida de los sistemas.

Para apoyar la IR de sistemas embebidos se propuso la metodología ABC-Besoins-SEM [11], en la cual se plantea un modelo de requisitos para este dominio específico, un modelo conceptual y un modelo de diseño, que permite dar continuidad hacia la fase de diseño de un sistema embebido. Para que el uso de una metodología sea eficiente, es ideal contar con una herramienta que la soporte. En la actualidad existen muchas [2], [6], [7], [13], [25], [26], [30] que apoyan la gestión de requisitos pero para sistemas diferentes a los sistemas embebidos. A continuación se mencionan y caracterizan algunas de estas herramientas (CALIBER-RM, DOORS, REQUISITEPRO y REM) estableciendo un comparativo que permite observar sus bondades y limitaciones [15], [31]:

En la tabla anterior la columna denominada "Características" presenta 13 grandes funcionalidades que corresponden a las herramientas de gestión de requisitos. Estas características fueron en general calificadas con sí o no indicando para cada herramienta si posee o no dicha característica. El carácter amigable de estas aplicaciones fue medido en escala de alto, medio o bajo.

Luego de realizar este comparativo fue posible detectar que ninguna de las herramientas existentes logra apoyar por completo la metodología ABC-Besoins-SEM, por las siguientes razones [10], [15], [20], [31]:

Tabla 1. Comparativo de herramientas para gestión de requisitos

HERRAMIENTAS ANALIZADAS				
CARACTERÍSTICAS	CALIBER-RM	DOORS	REQUISITEPRO	REM
• Facilidad para importación/exportar requisitos	Sí (formato Word)	Sí (formato Word)	Sí (Formato Word y CSV)	Sí (Formato HTML)
• Posibilidad de clasificar requisitos	Sí, se definen tipos de requisitos	Sí, se definen objetos (requisitos como objetos).	Sí, se definen tipos de requisitos	Sí, requisitos estándar
• Posibilidad de personalizar el modelo de requisitos	No	No	No	No
• Soporte selección y reutilización de requisitos en otros proyectos	No. Sólo copiar y pegar entre proyectos.	No. Sólo copiar y pegar módulos.	No, sólo copiar y pegar dentro del mismo proyecto.	No, sólo copiar y pegar dentro del mismo proyecto
• Creación de nuevos tipos de requisitos y atributos	Sí	No	Sí	No
• Manejo de versiones	Sí, automático.	Sí, automático.	Sí, automático.	No.
• Control de acceso	Sí, grupos y usuarios.	Sí, grupos y usuarios.	Sí, grupos y usuarios.	No
• Práctica y sencilla de utilizar. Interfaz amigable	Alta	Baja	Alta	Alta
• Licenciamiento Libre	No	No	No	Si
• Capacidad de generación de un modelo conceptual	No	Sí, diagramas UML	No	No
• Facilidad para generación de modelo de diseño	No	No	No	No
• Relación directa con el dominio de sistemas embebidos	No	No	No	No

- No aceptan el uso de diferentes clasificaciones de requisitos, y la mayoría de ellas trabaja con una taxonomía en la cual los requisitos se clasifican solamente como: funcionales, no funcionales.

- No permiten usar un modelo conceptual diferente a los sugeridos por la notación UML.

- No ofrecen la posibilidad de generar modelos orientados al diseño.

- Ninguna de ellas está orientada al dominio de sistemas embebidos, que posee características particulares a tener en cuenta.

- De las herramientas analizadas, sólo REM es de uso libre. Sólo a esta se le pueden hacer adaptaciones y modificaciones en el código.

Al mejoramiento de estas deficiencias se orienta la herramienta REASEM propuesta en este artículo.

IV. SOLUCIÓN PROPUESTA

REASEM es una herramienta que ayuda al analista de sistemas en la captura, clasificación y documentación de requisitos en el dominio de los sistemas embebidos. Igualmente la herramienta ofrece definiciones de Ingeniería de Requisitos, sistemas embebidos y su modelamiento conceptual, así como también proporciona el soporte para hacer repercutir los requisitos en el modelo conceptual del sistema [10].

Enseguida se hace una breve descripción de la metodología ABC-Besoins-SEM [11] para cuyo soporte se desarrolló la herramienta REASEM presentada en este trabajo. La explicación completa sobre los modelos y estructura de ABC-Besoins-SEM no hace parte del alcance de este artículo. Luego de tener un panorama general de la metodología se describirá en detalle la herramienta REASEM.

Metodología ABC-Besoins-SEM

Es una metodología de Ingeniería de Requisitos para el análisis de sistemas embebidos que permite [11]:

- Adquirir un conocimiento básico del sistema, mediante un modelo de interacciones típicas de un sistema embebido con el ambiente y otros sistemas embebidos.
- Capturar los requisitos de este tipo de sistemas, ofreciendo para ello un modelo de requisitos propio.
- Desarrollar modelos conceptuales con base en una plantilla general que contiene el modelo lógico típico de los sistemas

embebidos, y que facilita la trazabilidad entre requisitos y elementos del modelo conceptual.

- Facilitar el paso a la etapa de diseño, mediante el uso de un modelo de diseño genérico.

La metodología apoya las etapas básicas del estudio de requisitos y ofrece elementos de soporte para cada una de estas etapas [34] (ver ilustración 1).

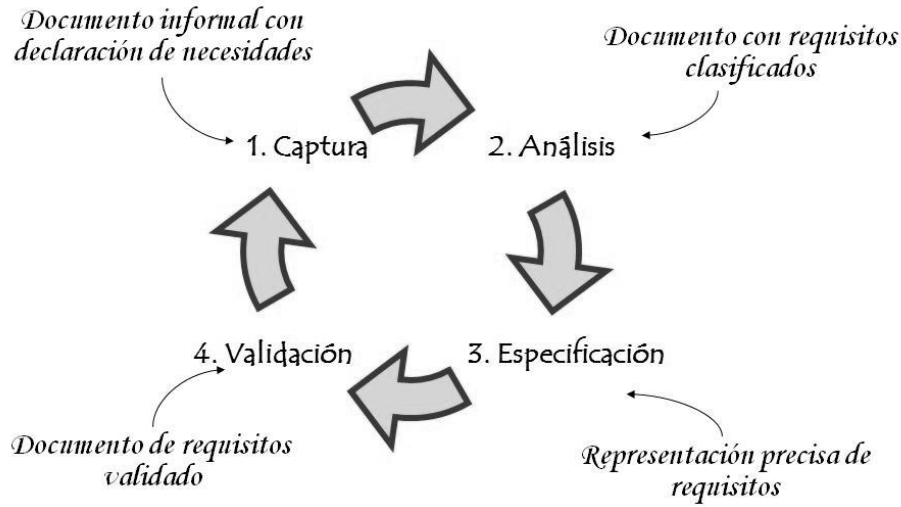


Ilustración 1. Etapas básicas del estudio de requisitos. Fuente [11]

Con ayuda de la metodología, en la fase 2 se clarifican y clasifican los requisitos, usando categorías de ABC-Besoins [34] adaptadas a las características de los sistemas embebidos.

Para la fase 3 la metodología propone una plantilla de modelo conceptual en la cual se involucran todos los elementos y relaciones típicas de un sistema embebido. Esto le permite al analista establecer correspondencias entre el modelo y los requisitos previamente recopilados. La metodología permite incluir en el modelado conceptual la interacción del sistema embebido con su ambiente, con el sistema contenedor y con otros sistemas embebidos.

En la fase 4, aunque no se propone una forma particular de validación se elaboran los requisitos y se organizan en el documento de requisitos.

Con base en la estructura de la metodología ABC-Besoins-SEM, se hará una descripción de REASEM, herramienta que soporta dicha metodología.

V. REASEM

REASEM (Requisitos apoyados por la metodología ABC-Besoins-SEM) es una herramienta de soporte a la metodología ABC-Besoins-SEM [11], que permite la interacción con el modelo de requisitos, conceptual y de diseño de dicha metodología. REASEM ofrece las funciones básicas de un software que apoya la fase de IR, tales como [15], [2]: permitir la identificación, clasificación y documentación de requisitos, y ayudar a la generación de un modelo conceptual y a su transformación en un modelo de diseño. La estructura funcional de la herramienta se presenta en la ilustración 2:

Consecuente con su filosofía de asistente, la aplicación guía al usuario a través del proceso de obtención y posterior representación de los requisitos de un sistema embebido. Inicialmente el usuario tiene la posibilidad de adquirir los conceptos básicos para entender cómo funciona un sistema embebido y qué es la ingeniería de requisitos. Luego de esta

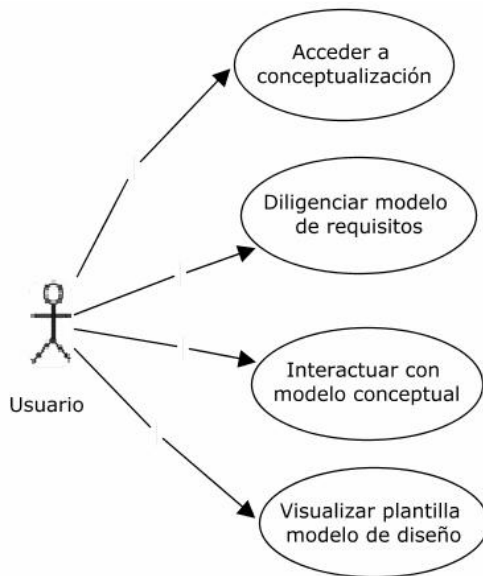


Ilustración 2. Estructura funcional REASEM. Fuente: [10]

fase de contextualización, se pasa a ingresar los requisitos del sistema. Estos se encuentran clasificados en 13 categorías con sus correspondientes sub-categorías, y luego son asociados a elementos del modelo conceptual. Por último se presenta la plantilla del modelo de diseño.

En la interacción con el prototipo, se logran tres productos: un documento con los requisitos correspondientes a cada una de las categorías y sub-categorías; un segundo documento que contiene el modelo conceptual referido a los requisitos, y por último, un documento que contiene el modelo de diseño [10].

Los modelos conceptual y de diseño propuestos en la metodología ABC-Besoins-SEM [11] incluyen una base estándar de las funcionalidades de un sistema embebido. El usuario podrá realizar cambios en los nombres y descripciones de los elementos que constituyen cada uno de estos modelos, pero no podrá adicionar ni eliminar componentes.

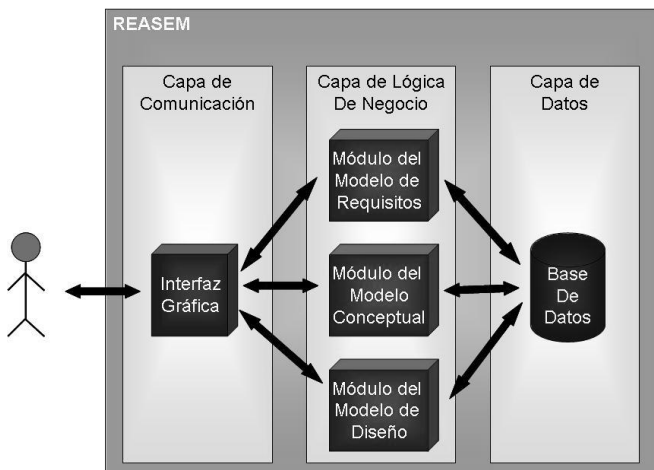


Ilustración 3. Arquitectura básica de REASEM. Fuente [10]

Arquitectura Básica de REASEM

Como se observa en la ilustración 3 la arquitectura de la herramienta está conformada por las tres capas siguientes:

Capa de comunicación: Contiene la interfaz gráfica, en la cual se establece la comunicación usuario-herramienta.

Capa de lógica de negocio: Se compone de todas las funciones y métodos de la metodología ABC-Besoins-SEM. Los tres módulos de esta capa son: el módulo del modelo requisitos, en donde será posible ingresar requisitos pertenecientes a las diferentes categorías y sub-categorías por dicho modelo; el módulo del modelo conceptual que permite generar el modelo y relacionar sus componentes con los requisitos obtenidos. Por último, dentro de la segunda capa, está el módulo del modelo de diseño el cual guiará al usuario en la construcción de un modelo de diseño especificado en términos básicos del lenguaje SystemC.

Capa de datos: Almacena información referente a las fases de desarrollo de los sistemas embebidos, desde su definición, pasando por la definición de requisitos, la elaboración del modelo conceptual, hasta el modelo de diseño.

Implementación de REASEM

La aplicación fue implementada en el lenguaje de programación Visual Basic.Net 2003 y cuenta con una base de datos en Microsoft Access 2007. REASEM funciona con filosofía de asistente, es decir, proporciona al usuario un proceso guiado en su interacción con la herramienta. Su estructura funcional se divide en cuatro fases [10], [11]: Contextualización, modelo de requisitos, modelo conceptual y modelo de diseño. La primera fase es una introducción al tema de los sistemas embebidos, a la ingeniería de requisitos y a los modelos conceptual y de diseño, en donde se presenta información pertinente para entender los conceptos manejados en la herramienta. La fase de modelo de requisitos, se compone de una serie de formularios que le permiten al usuario ingresar uno a uno los requisitos relacionados al proyecto del sistema embebido que este desarrollando. La tercera fase presenta el modelo conceptual y posibilita observar los requisitos relacionados con cada componente del modelo conceptual y por último se encuentra el modelo de diseño, que se encuentra basado en el modelo conceptual anterior y permite de igual forma verificar los requisitos relacionados con cada uno de sus componentes.

A continuación se presentan algunas ilustraciones que permiten observar la interfaz de la herramienta [10]:

La ilustración 4 muestra una de las primeras pantallas de la herramienta en la cual se presentan conceptos básicos requeridos para su uso. La ilustración 5, presenta uno de los formularios para la definición de los requisitos:

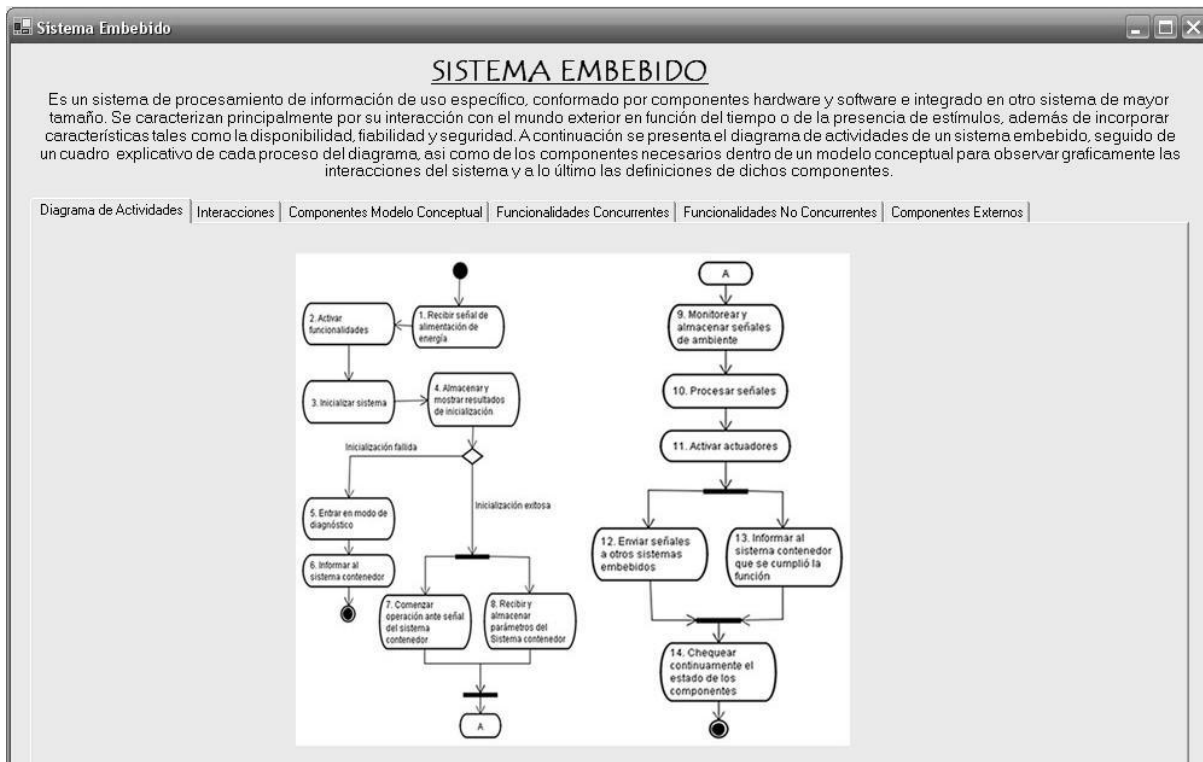


Ilustración 4. Fase de contextualización en REASEM. Fuente [10]

Modos de Operación

Categoría: Selección de Alternativas
 Son requisitos relacionados con la forma en que el sistema debe desplegar al usuario las posibles alternativas de uso. Este tipo de requisitos da paso a los modos y submodos de operación.

SubCategoría: Modos de Operación
 Un Modo de Operación es un conjunto de funcionalidades del sistema que se activan o desactivan con la ocurrencia de un evento, y posterior envío de una señal. Los modos de operación típicos de un sistema embebido son: Encendido, apagado o dicho de otra forma, activo e inactivo.

Definición de los Modos de Operación
 Ingrese uno a uno los modos de operación y seleccione por cada uno los elementos del modelos conce...

Ingreso de Requisitos

- El sistema debe contar con los siguientes modos de operación (ingrese uno a uno los modos de operación en el campo inferior):
- Seleccione los componentes del modelo conceptual relacionados con el requisito a ingresar. (Clic en el boton seleccionar componentes)

Requisitos Capturados

Código	Requisito
R1A-1	El sistema embebido a desarrollar debe contar con el modo de operación: prueba
R1A-2	El sistema embebido a desarrollar debe contar con el modo de operación: otra prueba
R1A-3	El sistema embebido a desarrollar debe contar con el modo de operación: modo

Información del proyecto
 * Proyecto: 72-prueba

Ilustración 5. Fase del modelo de requisitos en REASEM. Fuente [10]

Tal como se muestra en la ilustración anterior, todos los formularios para definición de requisitos cuentan con los siguientes elementos [10]:

- **Definiciones:** En este espacio se despliega una explicación de la categoría y sub-categoría del modelo de requisitos a la cual se le adicionarán requisitos si se considera conveniente.
- **Formulario de Ingreso:** La aplicación provee una sección para ingreso de los requisitos relacionados con la categoría y subcategoría que fue explicada en la sección previa de "Definiciones".
- **Lista de requisitos:** En este espacio del formulario es posible observar un listado de los requisitos ingresados en el proyecto y categoría/subcategoría activas.

• **Información del proyecto:** La aplicación permite al usuario recordar el proyecto sobre el cual está adicionando requisitos, desplegando su identificador único y nombre corto.

• **Botones de Control:** Esta sección cuenta con cinco botones: Guardar (permite almacenar en la base de datos el requisito ingresado); Nuevo (borra los datos que no se hayan guardado en la base de datos y deja en blanco los campos susceptibles de diligenciamiento en el formulario); Eliminar (Borra definitivamente un requisito de la base de datos); Anterior (Facilita la devolución entre formularios); Siguiente (Permite avanzar entre formularios).

La ilustración 6 presenta el modelo conceptual planteado en la metodología ABC-Besoins-SEM y con el cual se puede interactuar a través de REASEM.

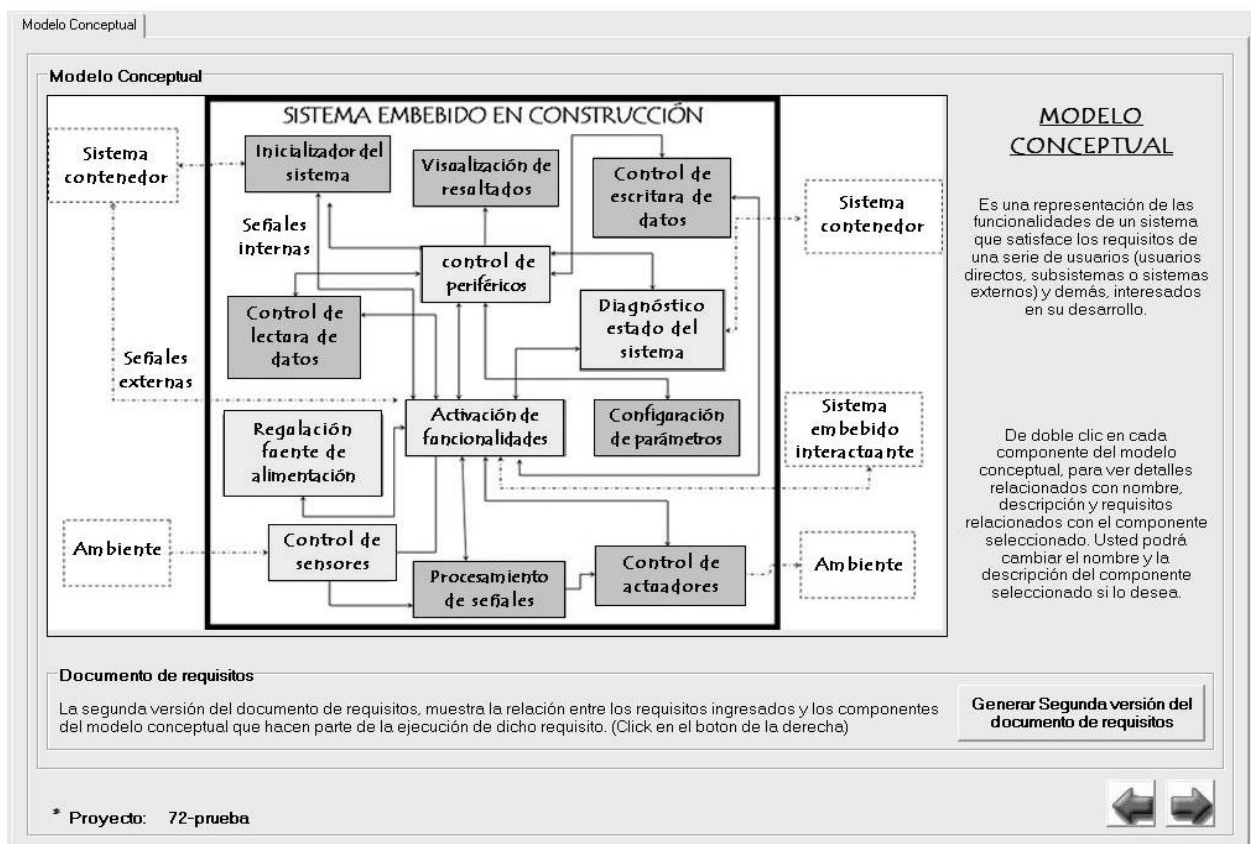


Ilustración 6. Fase del modelo conceptual en REASEM. Fuente [10]

Este formulario facilita el enlace entre los requisitos y los componentes típicos de un modelo conceptual para sistemas embebidos de acuerdo a los hallazgos de la metodología ABC-Besoins-SEM [11]. El usuario tendrá la posibilidad en este punto de modificar los nombres de los componentes típicos del modelo conceptual y conectar los requisitos ingresados con cada uno de los elementos de dicho modelo, además de obtener un documento en donde se hacen expresas estas relaciones.

V. VALIDACIÓN DE REASEM

El desarrollo y funcionamiento de la herramienta se validó mediante un caso de estudio, basado en un sistema de alarma contenido en un sistema domótico o sistema de hogar inteligente [20], [22]. Se representaron los requisitos, se elaboró el modelo conceptual referido a dichos requisitos y se constató el paso del modelo conceptual al de diseño. Para obtener mayor

- REASEM facilita la gestión de requisitos y la documentación de sistemas embebidos.
- REASEM se acopla perfectamente a las características de las metodología ABC-Besoins-SEM, ya que implementa los diferentes modelos (modelo de requisitos, modelo conceptual y modelo de diseño).
- REASEM posibilita a los diseñadores de sistemas embebidos su introducción en la cultura del análisis y diseño.

VII. CONCLUSIÓN

A pesar de los avances en la Ingeniería de Requisitos aún falta mucho trabajo por hacer [1], [14]. Las carencias son más notorias en los nuevos tipos de sistemas, en los cuales como en el caso de los sistemas embebidos, las metodologías no se ocupan de la definición de requisitos ni del modelo conceptual.

REASEM constituye una contribución al desarrollo de la Ingeniería de Requisitos en sistemas embebidos. Ahora los analistas y diseñadores de este tipo de sistemas podrán realizar un proceso guiado en el cual se descubran y traten, elaboren el modelo conceptual y el de diseño, facilitando la codificación e implementación del sistema [11].

El sistema REASEM es la base para el desarrollo de otras herramientas de gestión de requisitos en el campo de los sistemas embebidos.

VIII. TRABAJO FUTURO

Existe una línea de trabajo interesante si se piensa en proveer funcionalidades que permitan flexibilidad y dinamismo en la generación de los modelos conceptual y de diseño, pues en este momento no es posible adicionar ni eliminar elementos de ninguno de los modelos.

Por otro lado se puede pensar en ampliar las funcionalidades de la herramienta para permitir la adición y eliminación de categorías de requisitos, de modo que REASEM pueda ser usada en otros dominios diferentes al de sistemas embebidos.

Es pertinente darle una orientación web, a la aplicación, para que sea multiusuario, facilitar el trabajo en equipo, y haciéndola mas flexible.

BIBLIOGRAFÍA

- [1] Anton A., Goal-Based Requirements Analysis, ICRE'96, Colorado Springs, Colorado USA, IEEE, 1996. [Anton 1996]
- [2] Borland® Caliber® Analyst. Disponible en: <http://www.borland.com/us/products/caliber/index.html>. Último acceso: Julio de 2008.
- [3] Broy M., Requirements Engineering for Embedded Systems. Proceedings of the First Workshop Formal Design of Safety Critical Embedded Systems (FemSys). 1997.
- [4] Capel M., Holgado J., 2004, A New Design Procedure for a Real-Time Hybrid System Model. En: Memoria IV Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento, Vol. 1, p. 191- 204.
- [5] Cyre W., Capture, Integration, and Analysis of Digital System Requirements with Conceptual Graphs. IEEE Transactions on Knowledge and Data Engineering, Vol. 9(1), 1997.
- [6] Departamento de lenguajes y sistemas informáticos. REM- Herramienta experimental gratuita para la Gestión de Requisitos de proyectos de desarrollo software. Último acceso: octubre 20 de 2008. Disponible en: <http://www.lsi.us.es/descargas/descargas.php>
- [7] Díez A., 2001. IRqA y el desarrollo de proyectos: Experiencias Prácticas. I Jornadas de Ingeniería de Requisitos Aplicadas. JIRA 2001. Seville, Spain.
- [8] Durán A., 2000, Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información. Tesis Doctoral, Universidad de Sevilla, España.
- [9] Eric S., Yu K., 1997, Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering. En: Third IEEE International Symposium on Requirements Engineering.
- [10] Gómez J., 2008, REASEM: Herramienta para la gestión de requisitos para el dominio de sistemas embebidos. Tesis, Universidad de Antioquia, Colombia.
- [11] González L., 2008, Metodología de Ingeniería de Requisitos para el Análisis de Sistemas Embebidos. Tesis de Maestría, Universidad de Antioquia, Colombia.
- [12] Granollers, T., 2003. User Centred Design Process Model. Integration of Usability Engineering and Software Engineering. INTERACT 2003 (Doctoral Consortium), Zurich (Suiza).
- [13] IBM. Rational RequisitePro. Disponible en: <http://www-306.ibm.com/software/awdtools/reqpro/>. Último acceso: Julio de 2008.
- [14] IEEE, 1993. IEEE Recommended Practice for Software Requirements Specifications. IEEE/ANSI Standard 830-1993, Institute of Electrical and Electronics Engineers, [IEEE 1993]
- [15] INCOSE, International Council of Systems Engineering SE Tools Taxonomy - Requirements Traceability Tools. Fecha de acceso: octubre 2008. Disponible en: <http://www.incose.org/ProductsPubs/products/SEtools/eia632tax/reqdefine.html>
- [16] Ingham M., Rasmussen R., Matthew B., Moncada A., 2006, Generating requirements for complex embedded systems using State Analysis. En: Acta Astronáutica, Vol. 58, p. 648 - 661.
- [17] Insfrán, E., Pastor, O., Wieringa, R., 2002. Requirements Engineering-Based Conceptual Modeling. Requirements Engineering Journal, Vol 7 (1).
- [18] Jacobson I., Christenson M., Jonsson P. and Overgaard G., 1992. Object Oriented Software Engineering: A Use Case Driven Approach Addison-Wesley.
- [19] Lamsweerde A.V., 2000. Requirements Engineering in the Year 00: A Research Perspective. Proc. 22nd International Conference on software Engineering. Limerick, ACM Press.
- [20] Lavi J., Kudish J., 2005, Systems modelling & requirements specification using ECSAM: an analysis method for embedded & computer-based systems. En: Innovations Syst Softw Eng, Vol. 1m, pp. 100-115.
- [21] Loucopoulos, P., 1994. The F3 (From Fuzzy to Formal) View on Requirements Engineering. Ingénierie des systèmes d'information, Vol. 2, N° 6, pp.639-655.
- [22] Lowe, D., Hall, W., 1999. Hypermedia and the Web. An Engineering approach. John Wiley & Son.
- [23] Marwedel P., 2003. Embedded system design. En: Kluwer Academic Publishers University of Dortmund. Germany.
- [24] OMG Model Driven Architecture, MDA., 1997. <http://www.omg.org/mda/>
- [25] Quality Systems & Software, Inc. DOORS- Requirements traceability tool.

Último acceso: octubre 20 de 2008. Disponible en: <http://www.qssinc.com>

- [26] Rational Software. RequisitePro- Requirements traceability tool. Último acceso: octubre 20 de 2008. Disponible en: <http://www.rational.com/products/reqpro/docs/datasheet.html>
- [27] Ross D, K. Schoman., 1977. Structured Analysis for Requirements Definition. En: Transactions on Software Engineering, IEEE, vol. 3(1) pp. 6-15.
- [28] Standish Group, 1994. CHAOS Report 1994. Disponible en: http://www.standishgroup.com/sample_research/updated.php
- [29] Standish Group. CHAOS Report, 2002. Disponible en: http://www.standishgroup.com/sample_research/updated.php
- [30] Technology Builders, Inc. Caliber-RM- Requirements traceability tool. Último acceso: octubre 20 de 2008. Disponible en: <http://www.tbi.com/products/caliber.html>
- [31] Telelogic. Improve Systems and Software Quality with Requirements definition and Requirements management. Disponible en: <http://www.telelogic.com/products/doors/>. Último acceso: Julio de 2008
- [32] Thayer, R., Dorfman, M., Foreword by Davis, A., 1997. Software Requirements Engineering . IEEE Computer Society Press.
- [33] Urrego G, Giraldo G, 2009. Del modelo de Requisitos al Modelo de diseño. Capítulo de libro: tendencias en Ingeniería de software e inteligencia artificial. Editorial Unal, Universidad Nacional sede Medellín.
- [34] Urrego G., 2005. ABC-Besoins: Une approche d'ingénierie de besoins fonctionnels et non-fonctionnels centrée sur les Agents, les Buts, et les Contextes. Tesis Doctoral, Universidad Paris 1, Pantéon Sorbona, Francia.
- [35] Urrego, G., 2004a. Language-Based Model for Requirements Engineering. 23 th IASTED International Conference on Software Engineering ~SE 2004~ Feb 17-19. Innsbruck, Austria.
- [36] Zave P., Jackson M., 1997. Four dark corners of requirements engineering. En: ACM Transactions on Software Engineering and Methodology. Vol. 6(1), pp. 1-30.

Universidad Nacional de Colombia Sede Medellín

Facultad de Minas

Escuela de Ingeniería de Sistemas

Misión

La misión de la Escuela de Ingeniería de Sistemas es fomentar y apoyar la generación o la apropiación de conocimiento, la innovación y el desarrollo tecnológico en el área de ingeniería de sistemas e informática sobre una base científica, tecnológica, ética y humanística.



Visión

La formación integral de profesionales desde el punto de vista científico, tecnológico y social que les permita adoptar, aplicar e innovar conocimiento en el campo de los sistemas e informática en sus diferentes aspectos, aportando con su organización, estructuración, gestión, planeación, modelamiento, desarrollo, procesamiento, validación, transferencia y comunicación; para lograr un desempeño profesional, investigativo y académico que contribuya al desarrollo social, económico, científico y tecnológico del país.



Escuela de Ingeniería de Sistemas
Dirección Postal:
Carrera 80 No. 65 - 223 Bloque M8A
Facultad de Minas. Medellín - Colombia
Tel: (574) 4255350 Fax: (574) 4255365
Email: esistema@unalmed.edu.co
<http://pisis.unalmed.edu.co/>