

Derivar casos de uso de un glosario

Graciela D.S. Hadad

Facultad de Ciencias Fisicomatemáticas e Ingeniería, Pontificia Universidad Católica Argentina,
Ciudad de Buenos Aires, Argentina
Departamento de Ingeniería e Investigaciones Tecnológicas, Universidad Nacional de La Matanza,
San Justo, Prov. Buenos Aires, Argentina
gracielahadad@gmail.com

Andrés Migliaro

Facultad de Ciencias Fisicomatemáticas e Ingeniería, Pontificia Universidad Católica Argentina,
Ciudad de Buenos Aires, Argentina
amigliaro@fibertel.com.ar

Nicolás Grieco

Facultad de Ciencias Fisicomatemáticas e Ingeniería, Pontificia Universidad Católica Argentina,
Ciudad de Buenos Aires, Argentina
ngrieco@gmail.com

Abstract

Business modeling helps detecting weaknesses in business processes enabling the envisagement of efficient improvements involving software systems. The use of natural language-based models reduces the communication gap between users and software engineers, and when those models are scenarios or use cases, they stimulate the imagination of software engineers and users to project improvements in business processes applying software systems. We show in the article an approach to business modeling building use cases derived from a specific glossary called language extended lexicon. This approach encourages a middle-out fashion while modeling use cases and gives a bootstrap advantage against other proposals.

Keywords: use cases, glossary, business modeling.

Resumen

El modelado del negocio permite detectar debilidades en los procesos del negocio dando pie a proyectar mejoras eficientes involucrando sistemas de software. Utilizando modelos en lenguaje natural se achica la brecha de comunicación entre los usuarios y los ingenieros de software, y cuando además se trata de modelos como los casos de uso o los escenarios, éstos estimulan la imaginación para proyectar mejoras en los procesos del negocio con sistemas de software, tanto por parte de los ingenieros de software como de los propios usuarios. Se presenta en este artículo un enfoque para el modelado del negocio construyendo casos de uso a partir de un glosario específico, denominado léxico extendido del lenguaje. Este enfoque aborda una modalidad middle-out para el modelado de los casos de uso, ofreciendo ventajas de arranque frente a otras propuestas.

Palabras clave: casos de uso, glosario, modelado del negocio.

1 INTRODUCCIÓN

Los modelos de lenguaje natural son de uso mucho más reciente frente a los modelos de especificación formal, modelos estructurados, o modelos orientados a objetos, y se han incorporado en varios métodos de la ingeniería de requisitos. Están entre otros: los Casos de Uso [8] [13] descripciones narrativas de la interacción entre un actor y el sistema; los Escenarios [20] [27] [15] descripciones de situaciones del universo de discurso en diversos formatos según los autores; glosarios con definiciones de términos del universo de discurso [14] [26] [11]; glosarios que describen la terminología utilizada en otros modelos de requisitos [23] [22] [1]; modelos de Reglas de Negocio [24] [6]; User-Stories [3] versión simplificada de Casos de Uso para Programación eXtrema, entre otros.

Los modelos en lenguaje natural facilitan la comunicación entre los involucrados y la validación de dichos modelos, aunque su verificación requiere en general procesos manuales o semi-automáticos frente a otros modelos que pueden verificarse automáticamente. Este inconveniente menor no opaca sus fortalezas, y es por ello que se han difundido ampliamente en el proceso de la ingeniería de requisitos. En particular, los casos de uso (CU) están siendo fuertemente utilizados en la práctica real aunque en un modo de construcción frecuentemente ad-hoc o con algunos lineamientos generales que poco ayudan a desarrolladores inexpertos.

En la literatura sobre CU, existen autores que proponen un proceso de construcción basado en un enfoque top-down, es decir, partir de uno o más CU de alto nivel de detalle y construir los siguientes CU en una forma de refinamiento por pasos. Por otro lado, otros autores consideran que los CU deberían construirse desde lo particular a lo general. Algunos de ellos no recomiendan un enfoque específico pero generalmente existe un principio subyacente top-down o bottom-up (ver el estudio realizado en [15]).

Se propone en este artículo describir un proceso de construcción de casos de uso con la visión del negocio, de manera de centrarse en el dominio del problema y no en la solución tecnológica. Para ello, en la siguiente sección se presentan estrategias en el modelado del negocio, en especial el del Proceso Unificado, que utilizan casos de uso; en la sección 3 se describen los modelos propuestos para describir el negocio; en la sección 4 se presenta una estrategia en el modelado de CU; en la sección 5 se describe la herramienta que da apoyo a la estrategia y se muestran los datos observados, y finalmente se exponen conclusiones.

2 ENFOQUES EN EL MODELADO DEL NEGOCIO CON CASOS DE USO

En [21] se presentan dos enfoques para construir CU: top-down y bottom-up. En el enfoque bottom-up, comienzan describiendo ejemplos de escenarios, identifican episodios comunes, generalizan los escenarios y sintetizan en CU que los abarcan, mientras que en la modalidad top-down comienzan identificando actores y CU, construyen la estructura de los CU, describen sus eventos, generan los escenarios y validan. Los autores sugieren el uso de ambos enfoques en forma iterativa en un mismo método.

El Proceso Unificado [9] [12] es un proceso completo de desarrollo de software basado en el modelo iterativo e incremental, dirigido por CU y centrado en la arquitectura. El Proceso Unificado describe flujos de trabajo, los que se ejecutan concurrentemente e interactúan entre sí usando los artefactos de los otros flujos. Se describe a continuación el flujo de trabajo que se encarga de modelar el negocio.

El flujo de Modelado del Negocio, de ejecución opcional, tiene por objetivo comprender el contexto del sistema. Maneja dos alternativas de modelado: el modelado del dominio y el modelado del negocio. El primero describe conceptos importantes del contexto como objetos del dominio y sus relaciones, capturados a través de expertos del dominio¹, y representados en un diagrama de clases o un glosario de términos del dominio. El modelado del negocio se utiliza para comprender los procesos del negocio de la organización, manejando dos tipos de representaciones: el modelo de casos de uso del negocio y el modelo de objetos del negocio. Los autores del Proceso Unificado [7, p.118] utilizan los términos “clases del dominio” y “entidades del negocio” en forma indistinta, y aclaran que el modelado del dominio puede considerarse como una variante simplificada del modelado del negocio², ya que este último presenta un nivel de detalle mayor, facilitando la traza desde él hacia los CU del sistema.

En [9] no se especifica un método ni heurísticas para identificar y describir los CU del negocio. Sólo se expresan pautas para construir CU que describen el sistema en el flujo de trabajo de Requisitos. La estrategia para la construcción de CU comienza: *«Primero el analista de sistemas ejecuta la actividad de Encontrar Actores y Casos de Uso para preparar una primera versión del modelo de casos de uso, con los actores y casos de uso identificados»*. En esta actividad una vez identificados actores y CU, éstos se describen brevemente y se construye el diagrama de casos de uso en su totalidad, que muestra las relaciones entre los CU y los actores. Posteriormente, en el avance del proceso, se van describiendo todos los CU, y se reestructura el modelo de CU definiendo generalizaciones entre los CU. Esta última actividad comprende detectar funcionalidades compartidas entre CU, funcionalidades adicionales y opcionales, y otras relaciones entre CU.

En la literatura, se puede observar que no hay posturas concordantes sobre cómo construir casos de uso. Por ejemplo, se proponen enfoques top-down para construir CU en [18] [25] [19] [10] mientras estrategias bottom-up en [20] [5]. Se puede decir que, si la funcionalidad del sistema es bien conocida, una aproximación top-down podría utilizarse, mientras que, si no hay suficiente conocimiento, éste debería capturarse mediante una estrategia diferente.

3 MODELOS UTILIZADOS

3.1 Caso de Uso

Se presenta a continuación la definición más difundida de Caso de Uso [9] [4]:

“Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo y que producen un resultado observable de valor para un actor concreto”

“Un escenario es una secuencia específica de acciones que ilustran un comportamiento”

Los CU en su definición inicial dada por Ivar Jacobson [8] describían la interacción entre un actor y

¹ En la literatura, los modelos generados en el análisis del dominio representan el conocimiento elicitado principalmente de la literatura técnica, implementaciones de sistemas existentes y el consejo de expertos [2] [17].

² La definición de modelado del dominio que utiliza el Proceso Unificado no se corresponde con la expresada por autores que han estudiado el tema de análisis del dominio [2] [17], pues para éstos, un modelo del dominio representa el conjunto de objetos, relaciones y reglas comunes en un dominio y que, por ende, pueden reusarse en diferentes aplicaciones relacionadas a dicho dominio, mientras que un modelo del negocio representa conceptos específicos del dominio de la aplicación.

el sistema. Esta definición es muy utilizada en Human Computer Interaction. Desde una óptica más amplia, los CU permiten modelar, además de la funcionalidad del sistema, el comportamiento del entorno en el cual éste operará. Esta última postura brinda una visión más acabada del problema a resolver, pues mira aspectos sociales del negocio y permite encarar el impacto de incorporar el sistema en el universo de discurso (UdeD).

Caso de Uso:	<Nombre del caso de uso>
Alcance:	<Sistema bajo estudio>
Nivel:	<Grado de detalle del caso de uso>
Actor Principal:	<Nombre del actor que inicia el caso de uso>
Involucrados e intereses:	<Objetivos e intereses de los involucrados>
Precondición:	<Condición que debe cumplirse para iniciar el caso de uso>
Disparador:	<Evento que inicia el caso de uso>
Garantías Mínimas:	<Afirmaciones de lo que será verdad al finalizar el caso de uso por éxito o por fracaso>
Garantías de Éxito:	<Afirmaciones de lo que será verdad al finalizar el caso de uso por éxito>
Escenario Principal de éxito:	
	1. <Pasos del Escenario Principal>
	2. ...
	3. ...
Extensiones:	
	3a. <Condición de la Extensión>
	3a1. <Pasos de la Condición>
	3a2. ...

Figura 1. Plantilla del Caso de Uso

La plantilla de caso de uso propuesto por Cockburn en [4] se utilizará en este artículo por ser una de las más ilustrativas en la literatura (ver Figura 1). Con ella se pueden identificar tres tipos de CU según su alcance:

- i) Caso de uso del negocio, que describe las actividades del negocio. Se estudia la organización en sí misma.
- ii) Caso de uso del sistema, que lo describe externamente, como caja negra. El actor principal es un usuario frente a la computadora.
- iii) Caso de uso de diseño, que describe al sistema internamente, como caja blanca. El actor principal es un objeto usuario.

Cabe mencionar que los casos de uso del sistema fue la primera versión de casos de uso, interacción entre actor-sistema, presentada por Jacobson [8], versión que luego avanzó hacia el diseño por los métodos orientados a objetos, y muy posteriormente se incluyó la visión del negocio. En este artículo se pretende utilizar los CU para la descripción del negocio por lo tanto se estará siempre haciendo referencia a los CU de alcance del negocio.

Por otro lado, los CU se distinguen por su nivel de detalle en [4]:

- i) Nivel Resumen, que describen múltiples sesiones de casos de uso de objetivo de usuario.
- ii) Nivel Objetivo de Usuario, que describen un objetivo particular e inmediato de un actor principal.
- iii) Nivel Sub-función, que describe objetivos parciales de un caso de uso de objetivo de usuario o de otra sub-función.

Este nivel en los CU establece una relación jerárquica entre los mismos, pues los pasos de un CU nivel resumen son a su vez CU de objetivo de usuario o CU nivel resumen. Los pasos en un CU de

nivel objetivo de usuario pueden ser directamente acciones o CU de nivel sub-función.

Los CU de nivel resumen dan una visión integradora del problema bajo estudio. Los CU de nivel objetivo de usuario brindan una descripción de cada actividad del negocio mientras que los detalles importantes pueden describirse mediante CU de nivel sub-función. La Figura 2 muestra mediante un ejemplo la relación jerárquica entre CU indicando la visión que presenta cada tipo de CU. La Figura 2-A muestra un CU nivel Resumen, donde el paso 11 “Informar Sentencia” se describe como un CU en la Figura 2-B, donde a su vez el paso 5 “Notificar a las Partes” se describe como otro CU en la Figura 2-C.

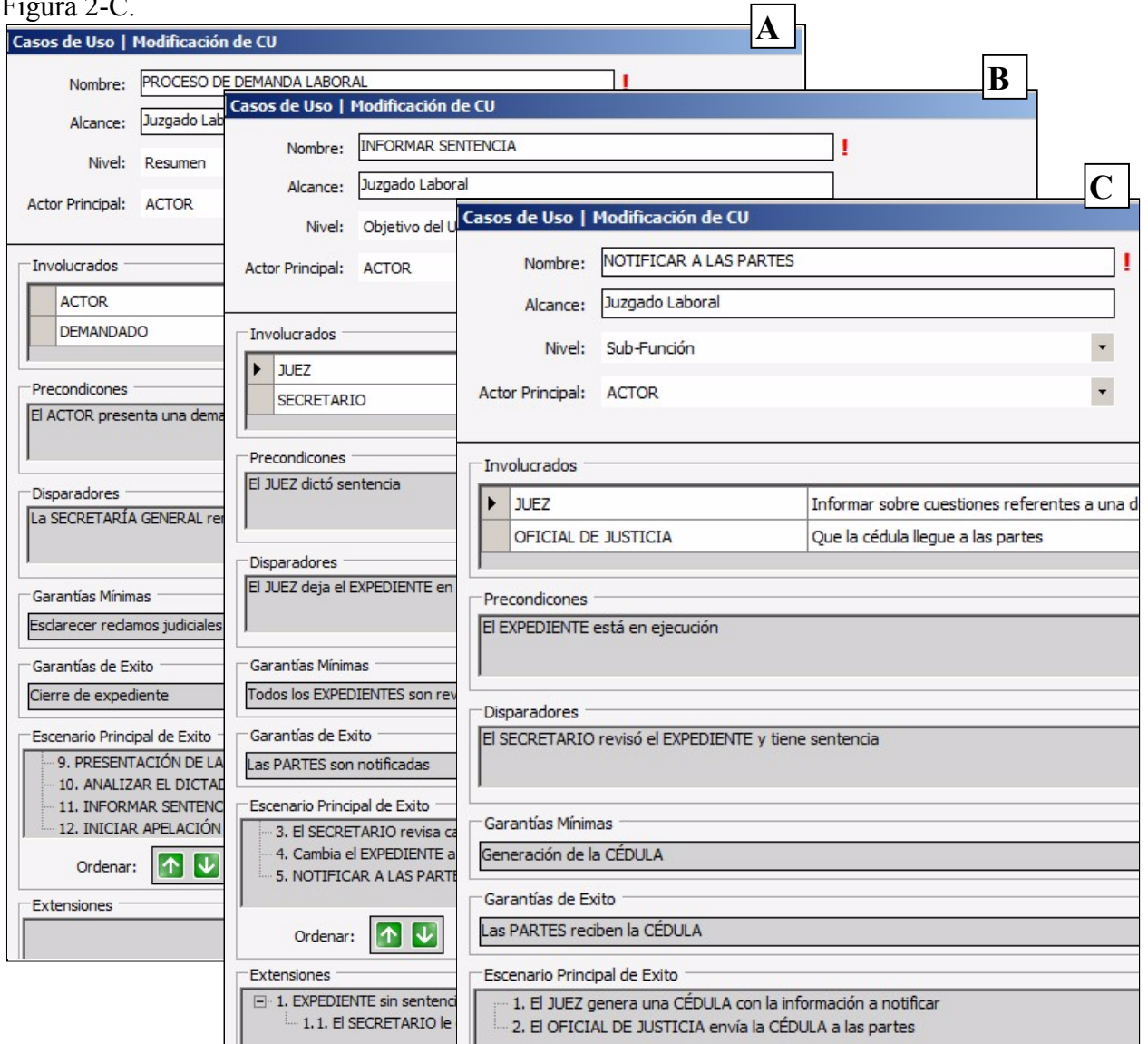


Figura 2. Ejemplo de niveles de detalle en los CU³

Los componentes que dan contexto al CU son: Alcance, Nivel, Actor Principal, Involucrados e Intereses y Precondición. Ellos representan aspectos importantes en el modelado del negocio. En el componente Alcance, siendo que se desea describir el negocio, puede identificarse la organización o departamento o área de la empresa bajo estudio en dicho CU.

³ Todos los ejemplos que se presentan en este artículo corresponden a un caso real realizado en un juzgado laboral de la Ciudad de Buenos Aires.

Un CU alberga todas las posibles variantes de una situación, ya sea para la obtención de un objetivo o para su fracaso. En los CU los pasos nunca llevan una condición en la variante normal (denominada Escenario Principal de Éxito), el resto de las variantes y excepciones se describen en el componente Extensiones, donde sí se aplica una condición a un conjunto de pasos que representan una determinada alternativa o excepción (denominados Escenarios Secundarios).

El componente Escenario Principal de Éxito incluye todos los pasos que representan el flujo normal de eventos, y en el componente Extensiones se incluyen los pasos que representan flujos alternativos ya sea de éxito o de fracaso del CU. Los pasos muestran el orden secuencial en que se realizan, y también admiten indicar un orden paralelo y repetición de pasos. Un paso puede ser a su vez el nombre de un CU.

3.2 Léxico Extendido del Lenguaje

El contenido de cada CU se escribirá utilizando el vocabulario que se emplea en el UdeD. Esto facilita la lectura de los CU por parte de los usuarios, promoviendo una mejor validación de los mismos. Para que los ingenieros de software puedan conocer ese vocabulario y utilizarlo en la comunicación con los usuarios, ya sea oralmente o por escrito a través de documentos y modelos, no alcanza con escuchar y tomar notas de los términos que emplean los usuarios, sino que es necesario generar un glosario con dichos términos.

Se propone aquí la creación de un léxico extendido del lenguaje (LEL), que es un glosario con términos del UdeD, donde además de la definición del término (denominada Noción) se incluye una descripción de cómo ese término repercute en el UdeD (denominada Impacto). La Figura 3 muestra el modelo del LEL, como un conjunto de términos, denominados símbolos, que son palabras o frases relevantes en el UdeD. Estos símbolos se describen mediante la noción y el impacto haciendo referencia a otros símbolos, formando una red de conexiones semánticas.

LEL: representación de los símbolos en el lenguaje del dominio de la aplicación.
 Sintaxis:
 $\langle \text{LEL} \rangle \Rightarrow \{ \langle \text{Símbolo} \rangle \}_1^N$

Símbolo: entrada del léxico que tiene un significado especial en el dominio de la aplicación.
 Sintaxis:
 $\langle \text{Símbolo} \rangle \Rightarrow \{ \langle \text{Nombre} \rangle \}_1^N + \langle \text{Noción} \rangle + \langle \text{Impacto} \rangle$

Nombre: identificación del símbolo. Más de uno indica la presencia de sinónimos.
 Sintaxis:
 $\langle \text{Nombre} \rangle \Rightarrow \langle \text{Palabra} \rangle | \langle \text{Frase} \rangle | \langle \text{Acrónimo} \rangle$

Noción: denotación del símbolo. Debe ser expresada usando referencias a otros símbolos y usando el vocabulario mínimo.
 Sintaxis:
 $\langle \text{Noción} \rangle \Rightarrow \text{Noción:} + \{ \langle \text{Oración} \rangle \}_1^N$

Impacto: connotación del símbolo. Debe ser expresado usando referencias a otros símbolos y usando el vocabulario mínimo.
 Sintaxis:
 $\langle \text{Impacto} \rangle \Rightarrow \text{Impacto:} + \{ \langle \text{Oración} \rangle \}_1^N$

donde
 $\langle \text{Oración} \rangle$ está compuesta solamente por Símbolos y No Símbolos, éstos últimos pertenecientes al vocabulario mínimo;
 $\langle \text{Palabra} \rangle$, $\langle \text{Frase} \rangle$ o $\langle \text{Acrónimo} \rangle$ tienen el significado común.

+ significa composición, {x} significa cero o más ocurrencias de x, | representa **or**
 Figura 3. El Modelo del Léxico Extendido del Lenguaje

Para definir cada término se lo clasifica previamente en sujeto, objeto, verbo o estado; de acuerdo a la clase a la que pertenece corresponderá el tipo de información que debe incluirse en la noción y en

el impacto. El proceso de creación del léxico se detalla en [16] y [7]. La Figura 4 muestra 2 ejemplos de símbolos, uno de tipo sujeto y el otro de tipo verbo.

Cabe mencionar que todo documento o modelo que se genere sería aconsejable escribirlo usando el vocabulario del UdeD, es decir, haciendo referencia a los símbolos del LEL. En particular, en este enfoque se propone describir los CU utilizando los términos del LEL.

The image displays two overlapping windows from a software application, both titled "Símbolos | Modificar".

The top window shows the configuration for the symbol "JUEZ". It is classified as a "SUJETO" (Subject). The "Sinónimos" (Synonyms) field contains "SEÑORÍA". Under the "Impacto" (Impact) section, several actions are listed: "FIRMA los EXPEDIENTES", "DICTAR SENTENCIA", "Toma decisiones ante las distintas instancias de...", "Resuelve las DEMANDAS", and "FIRMA los PROVEIDOS".

The bottom window shows the configuration for the symbol "INICIAR DEMANDA". It is classified as a "VERBO" (Verb). The "Sinónimos" field is empty. Under the "Impacto" section, the following actions are listed: "Se genera un EXPEDIENTE", "Se envía el EXPEDIENTE al JUZGADO asignado", and "Se aporta la documentación necesaria para conformar el EXPEDIENTE".

Figura 4. Ejemplos de términos del LEL

4 ENFOQUE EN EL MODELADO DEL NEGOCIO

4.1 Proceso de construcción de Casos de Uso

Esta estrategia se basa en la creación del LEL y la construcción de un conjunto de casos de uso que muestran las actividades del negocio. Se inicia la generación de CU derivándolos del glosario. Este paso genera un conjunto incompleto de CU pero que sirven de punto de partida para elicitación de información que permita completarlos y encontrar otros CU. Luego, los CU son organizados aplicando operaciones y heurísticas de integración. Las operaciones de composición / descomposición de CU tienen por objetivo hacer más comprensibles los CU, debido a que se puede presentar información duplicada o información relacionada dispersa en distintos CU o información con distinto nivel de detalle en el mismo CU. La integración de CU permite generar CU de nivel resumen que agrupan CU relacionados temporalmente, con lo cual se logra mostrar una visión global del UdeD.

La Figura 5 muestra las cinco actividades para construir los CU; en ella se observa que una vez descritos los CU y también luego de organizados, ellos son verificados y validados; en caso de

detectar defectos se retorna a la actividad Describir para corregirlos.

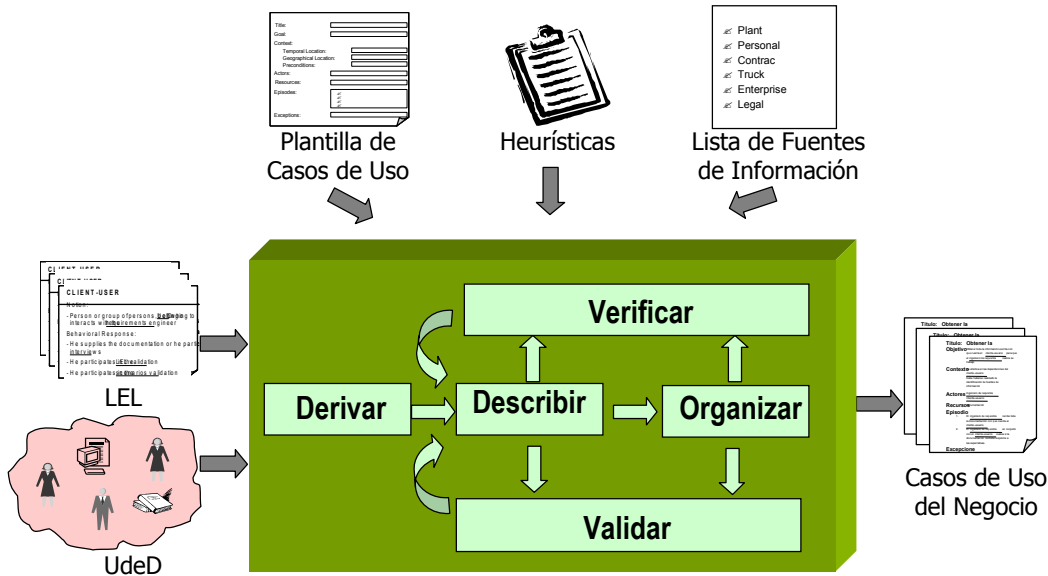


Figura 5. Construcción de Casos de Uso

La derivación de CU del LEL (ver Figura 6) consta de tres actividades secuenciales: i) identificar los actores de los CU, ii) identificar CU mediante la generación de una lista de posibles CU y iii) crear los CU completando la plantilla para cada CU identificado. Estas tres actividades se realizan utilizando exclusivamente información del LEL, siguiendo la heurística que se detalla en la subsección siguiente.

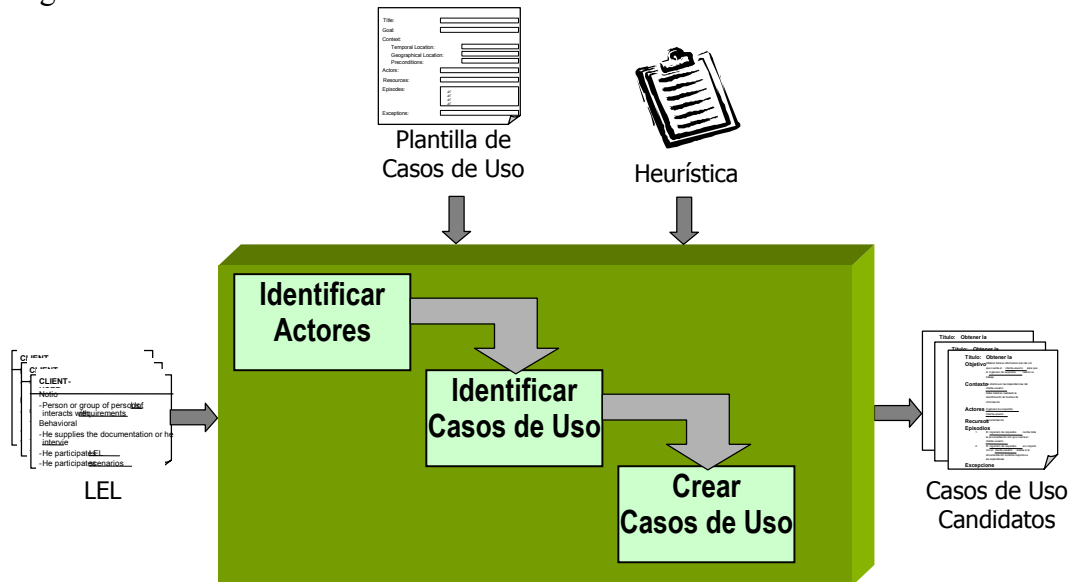


Figura 6. Derivar CU del LEL

Para describir los CU, se aplican técnicas de elicitación para recolectar información, como por ejemplo: entrevistas estructuradas, observación, lectura de documentación, análisis de protocolo, etc., y se usa la plantilla de casos de uso para elicitar. Se completa el resto de los componentes, donde se confirma y mejora el curso normal de eventos (Escenario Principal de Éxito), se indaga sobre cursos alternativos y excepciones (Extensiones). Si surgen nuevas actividades que se realizan en el UdeD y que no fueron derivadas del LEL se agregan nuevos CU. Al describir los CU es conveniente utilizar los términos del LEL y considerar que el Actor Principal y los Involucrados serán preferentemente términos del LEL del tipo sujeto.

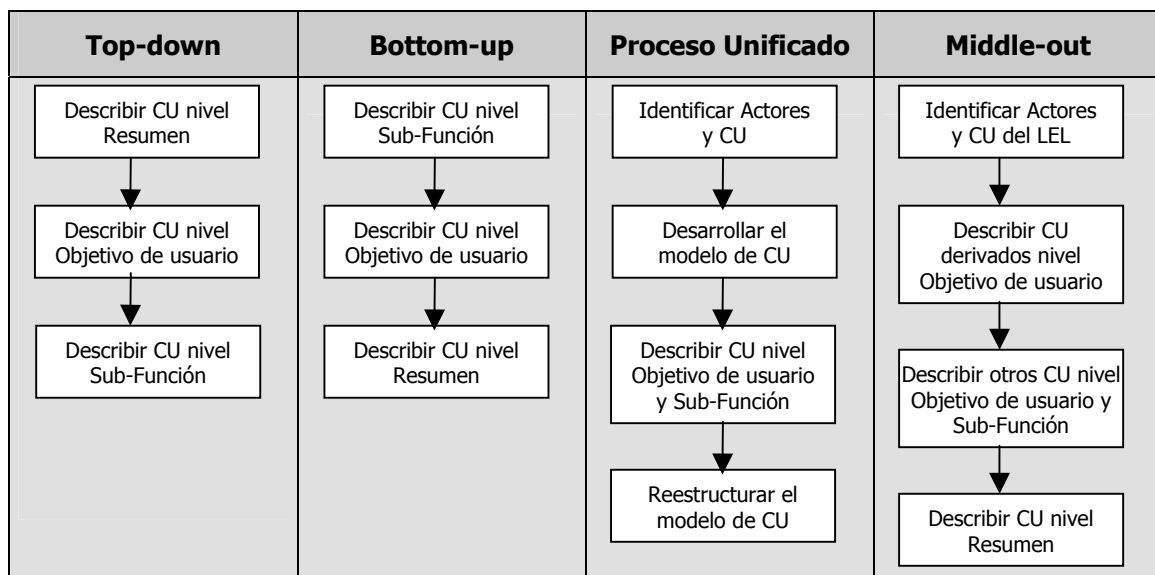
Posteriormente se estudian los CU en su conjunto y se los organiza: a) descomponiendo un CU en varios CU o componiendo dos o más CU en uno sólo, de manera de evitar superposición, redundancia, dispersión y lograr homogeneidad en el nivel de detalle; b) integrando el conjunto de CU en uno o más CU de nivel Resumen, de manera de lograr un panorama general del sistema bajo estudio. De esta forma, usando la misma plantilla de CU se describen las relaciones entre el conjunto de CU escribiendo un CU de nivel superior.

La modalidad de construcción presentada es middle-out pues se parte del LEL que tiene información de nivel medio de detalle, creando un conjunto de CU que describen actividades del UdeD, no siendo éstas ni los procesos del negocio ni procedimientos operativos detallados. A partir de estos CU se describen luego CU más detallados y finalmente se describen CU nivel resumen que agrupan a varios CU (integración).

En un enfoque top-down, se comenzaría describiendo CU nivel Resumen, donde luego cada paso del Escenario Principal de Éxito se describiría con CU de nivel Objetivo de usuario, y los pasos de algunos de estos CU se describirían a su vez como sub-casos de uso. Este enfoque es viable si se tiene a priori un conocimiento importante del negocio como para poder determinar cuáles son los procesos del negocio y las actividades que los componen. En un enfoque bottom-up se empiezan describiendo pasos que se agrupan en escenarios, los cuales luego conforman los CU nivel Objetivo de usuario y nivel Sub-función, para luego agrupar éstos en CU nivel Resumen. Este enfoque puede realizarse en un proyecto relativamente pequeño, sino los ingenieros de software se pueden enmarañar en los detalles, dificultándose la tarea de síntesis y abstracción.

En la Tabla 1 se describen las actividades usando distintas modalidades, siendo el enfoque middle-out el que se presenta en este artículo.

Tabla 1. Enfoques en la construcción de casos de uso



4.2 Derivar Casos de Uso

La actividad de derivación tiene por objetivo identificar actores y contar con algunos CU a partir de los cuales se facilita la elicitación de más información para completar esos CU y encontrar nuevos.

Es una actividad semi-automática para la cual se ha desarrollado una herramienta pero que requiere cierta intervención humana en la interpretación semántica de información del LEL que puede o no trasladarse a los CU.

Se detalla a continuación la heurística de derivación, cuyos tres pasos se grafican en la Figura 6, donde se indican las entradas al proceso y el producto a obtener.

H1. Identificar Actores:

Se extraen del LEL todos los términos que fueron clasificados como sujetos. Cada término tipo sujeto es considerado un actor candidato para los CU. Se puede seleccionar de esta lista algunos o todos los sujetos.

H2. Identificar Casos de Uso:

Por cada actor, se toman sus impactos y se considera que cada impacto es un posible CU. Esto es debido a que al definir un término en el LEL clasificado como sujeto, se coloca en el impacto las responsabilidades y actividades que realiza dicho sujeto. El nombre del CU corresponde a la acción del impacto más el predicado del impacto.

H3. Crear Casos de Uso a partir del LEL:

Para cada CU identificado, se empiezan a completar algunos de sus componentes de la siguiente manera.

H3.1. Del término tipo sujeto que dio origen al CU:

H3.1.1. El *Actor Principal* será el término sujeto.

H3.1.2. La *Precondición* se puede obtener de las interrelaciones que pueden observarse de los impactos del sujeto, en referencia con el impacto que dio origen al CU.

H3.1.3. El *Disparador* puede surgir del propio impacto que dio origen al CU.

H3.2. Si el impacto que dio origen al CU no contiene ningún término tipo verbo:

H3.2.1. Identificar otros términos del LEL en el impacto del sujeto.

H3.2.2. Si alguno de estos términos es del tipo sujeto, incluirlos en el componente *Involucrados*.

H3.3. Si el impacto contiene un término tipo verbo, acceder a la información de dicho término:

H3.3.1. Incluir en *Involucrados* todos los términos tipo sujeto encontrados en la descripción del término verbo.

H3.3.2. Incluir en el *Escenario Principal de Éxito* como pasos cada uno de los impactos del término verbo.

H3.3.3. En caso que algún impacto del término verbo contenga alguna condición bajo la cual se realiza dicha acción, este impacto irá como un paso en las *Extensiones*.

H3.3.4. Si no se completó *Precondición* en H3.1.2 ni *Disparador* en H3.1.3, entonces esta información puede estar presente en la noción del símbolo verbo.

H3.3.5. *Garantías de Éxito* y *Garantías Mínimas* se pueden obtener de la lectura de la noción del símbolo verbo.

5 HERRAMIENTA PARA DERIVAR CASOS DE USO⁴

⁴ La herramienta se ha construido en el marco del trabajo final de A. Migliaro y N. Grieco en la carrera de postgrado “Especialización en Ingeniería del Software” de la UCA.

Se ha construido una aplicación en Visual Basic.Net con Microsoft SQL que permite mantener LEL asociados a distintos universos de discurso y múltiples conjuntos de CU por cada LEL. Esto significa que se pueden derivar distintos sub-grupos de CU de un mismo LEL seleccionando todos o algunos de los términos de tipo sujeto del LEL.

Respecto al LEL, la herramienta permite ingresar cada término con sinónimos, modificar o eliminar símbolos, alta rápida de símbolos; al escribir la noción y el impacto sugiere la mención a otros términos; al ver la definición de un término se puede consultar la definición de los términos a los que hace referencia; realiza verificaciones de consistencia y asiste para la corrección de defectos. Verifica la existencia de símbolos duplicados, símbolos incompletos, símbolos sin referencias a otros términos, y símbolos no referenciados.

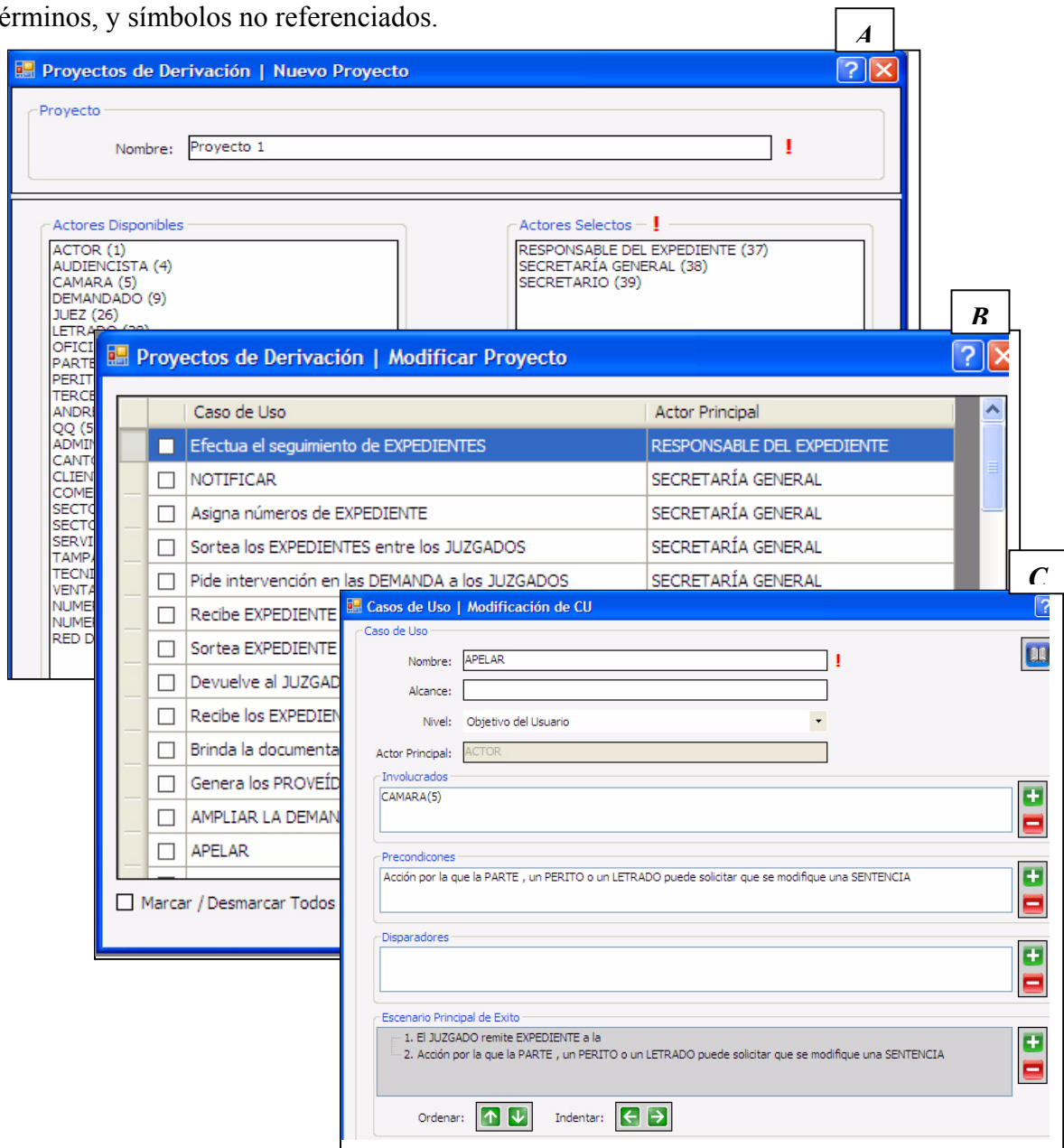


Figura 7. Pasos de la herramienta al derivar CU

La herramienta genera automáticamente un conjunto de CU a partir de un LEL, aplicando la heurística de derivación descrita en la sección anterior. Se han incorporado a la herramienta los

pasos: H1, H2, H3.1.1, H3.2.1, H3.2.2, H3.3.1, H3.3.2 y H3.3.4. Para ello, se genera un proyecto seleccionando un LEL y seleccionando uno, varios o todos los actores que la herramienta sugiere (ver Figura 7-A). A partir de esto, la herramienta genera una lista de CU candidatos (ver Figura 7-B) y luego presenta una descripción incompleta de cada CU candidato (ver Figura 7-C). La herramienta permite completar los CU derivados, agregar nuevos CU o eliminar existentes. Otras facilidades que brinda la herramienta son consultar los CU, generar reportes y estadísticas. Adicionalmente, la aplicación administra usuarios y brinda ayuda contextual.

6 CONCLUSIONES

Construir los CU del negocio permite identificar problemas y oportunidades de mejoras en los procesos del negocio actual. Por lo tanto, es una etapa esencial cuando se supone que los procesos del negocio son ineficientes, por ejemplo, cuando se está admitiendo una reingeniería de los procesos.

Dada su rica expresividad, los CU son un medio que facilitan la elicitación de requisitos, la negociación de los mismos y su validación. Estimulan la imaginación, facilitando la generación de propuestas por parte de todos los involucrados. Permiten presentar sin grandes costos (por su facilidad de construcción) distintas alternativas de solución a los problemas y necesidades manifestadas en el UdeD.

Se observa que el Proceso Unificado se centra más en el diseño de los requisitos para una solución dada que en estudiar el contexto donde el software operará y en establecer los requisitos del software acorde al negocio e independientes de implementación; pues en los CU que modelan los requisitos ya se incluye la arquitectura del sistema, aún cuando inicialmente se describen CU del negocio cuya utilidad queda diluida en el proceso. A diferencia de la mayoría de las propuestas basadas en CU que usan un diagrama de CU para mostrar las relaciones entre los CU, acá se propone usar la misma plantilla de CU para mostrar estas relaciones, facilitando la lectura por parte de los usuarios, recordando que además se describen usando su propio vocabulario.

La propuesta expuesta presenta un punto de arranque para construir CU, que es partir de información contenida en el LEL. Esto es una ventaja muy importante frente a propuestas que sugieren identificar actores y casos de uso sin dar pautas de cómo arribar a dicha información. Por otro lado, se ha construido una herramienta que automatiza en gran parte el proceso de derivación de los CU desde el LEL. Se espera ampliar la herramienta para que abarque otras actividades en la construcción de los CU, como ser incorporar las operaciones para reorganizar los CU y sistematizar la integración y la verificación de los CU.

REFERENCIAS

- [1] Alspaugh T.A., Antón A.I., Barnes T. y Mott B.W. An Integrated Scenario Management Strategy. IEEE International Symposium On Requirements Engineering, Limerick, Irlanda, IEEE Computer Society Press, pp.142-149, 1999.
- [2] Arango G. Domain engineering for software reuse. Ph.D. thesis, Department of Computer Science, University of California, Irvine, 1988.
- [3] Beck K. Extreme Programming Explained: Embrace Change. Addison-Wesley, 2000.
- [4] Cockburn A. Writing Effective Use Cases, Humans and Technology. Addison-Wesley, 2000.
- [5] Dano B., Briand H. y Barbier F. An Approach Based on the Concept of Use Case to Produce Dynamic Object-Oriented Specifications. Third IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, Los Alamitos, CA, pp.54-64, 1997.

- [6] Gottesdiener E. Business Rules as Requirements. *Software Development*, 7(12), Dic. 1999.
- [7] Hadad G.D.S., Doorn J.H. y Kaplan G.N. Creating Software System Context Glossaries. *Encyclopedia of Information Science and Technology*, IGI Global, 2º edición, Octubre 2008.
- [8] Jacobson I., Christerson M., Jonsson P. y Overgaard G. *Object-Oriented Software Engineering - A Use Case Driven Approach*. Reading, MA: Addison Wesley, Nueva York: ACM Press, 1992.
- [9] Jacobson I., Booch G. y Rumbaugh J. *The Unified Software Development Process*. Addison-Wesley, Reading, MA, 1º edición, Febrero 1999.
- [10] Jones S. y Maiden N. RESCUE: An Integrated Method for Specifying Requirements for Complex Sociotechnical Systems. *Information Science Publishing*, Idea Group Inc., Maté & Silva editores, Londres, capítulo XV:245-265, 2005.
- [11] Kovitz B.L. *Practical Software Requirements: A manual of Content and Style*. Greenwich, CT: Manning Publications Co., Octubre 1998.
- [12] Kruchten P. *The Rational Unified Process: An Introduction*. Addison-Wesley, 3º ed., 2004.
- [13] Leffingwell D. y Widrig D. *Managing Software Requirements - A unified approach*. Addison-Wesley Longman, 2º edición, 2003.
- [14] Leite J.C.S.P. y Franco A.P.M. A Strategy for Conceptual Model Acquisition. *IEEE First International Symposium on Requirements Engineering*, IEEE Computer Society Press, Los Alamitos, CA, pp.243-246, 1993.
- [15] Leite J.C.S.P., Hadad G.D.S., Doorn J.H. y Kaplan G.N. A Scenario Construction Process. *Requirements Engineering Journal*, Springer-Verlag London Ltd., 5(1):38-61, 2000.
- [16] Leite J.C.S.P., Doorn J.H., Kaplan G.N., Hadad G.D.S. y Ridaio M.N. Defining System Context using Scenarios. *Perspectives on Software Requirements*, Kluwer Academic Publishers, EEUU, ISBN: 1-4020-7625-8, capítulo 8:169-199, 2004.
- [17] Loucopoulos P. y Karakostas V. *System Requirements Engineering*. McGraw-Hill, Londres, 1995.
- [18] Oberg R., Probasco L. y Ericsson M. *Applying Requirements Management with Use Cases*. Rational Software Corporation, 1998.
- [19] Paech B., Denger C., Kerkow D. y von Knethen A. *Requirements Engineering for Technical Products: Integrating Specification, Validation and Change Management*. Information Science Publishing, Idea Group Inc., Maté & Silva editores, Londres, capítulo X:153-169, 2005.
- [20] Potts C., Takahashi K. y Antón A.I. Inquiry-Based Requirements Analysis. *IEEE Software*, 11(2):21-32, 1994.
- [21] Regnell B., Anderson M. y Bergstrand J. A Hierarchical Use Case Model with Graphical Representation. *ECBS'96, IEEE International Symposium and Workshop on Engineering of Computer-Based Systems*, Alemania, pp.270-277, 1996.
- [22] Regnell B., Kimbler K. y Wesslén A. Improving the Use Case Driven Approach to Requirements Engineering. Ph.D. Thesis: *Requirements Engineering with Use Cases – a Basis for Software Development*, Department of Communication Systems, Lund University, Reporte Técnico 132, Paper I:43-63, 1999.
- [23] Rolland C. y Ben Achour C. Guiding the construction of textual use case specifications. *Data & Knowledge Engineering* 25:125-160, 1998.
- [24] Ross R. *The Business Rule Book: Classifying, Defining and Modeling Rules*. Business Rule Solutions, LLC, 2º edición, 1997.
- [25] Schneider G. y Winters J. *Applying Use Cases, A Practical Guide*. Addison-Wesley, 1998.
- [26] Whitenack B.G. Jr. RAPPeL: A Requirements Analysis Process Pattern Language for Object Oriented Development. *Knowledge Systems Corp.*, 1994.
- [27] Wirfs-Brock R. Designing Objects and Their Interactions: A Brief Look at Responsibility-Driven Design. *Scenario-Based Design: Envisioning Work and Technology in System Development*, editor J. Carroll, John Wiley, Nueva York, pp. 337-359, 1995.