

Una propuesta para mejorar la completitud de requisitos utilizando un enfoque lingüístico*

Carlos M. Zapata J.** , Aldrin F. Jaramillo F.*** y
Fernando Arango I.****

Resumen

La calidad de los productos de software está estrechamente relacionada con la calidad de los requisitos especificados desde las primeras etapas del proceso de desarrollo; las propuestas encaminadas a la especificación de requisitos realizan incipientes esfuerzos para lograr que los requisitos del software sean lo suficientemente completos como para lograr la traducción de las necesidades y expectativas de los usuarios al producto final. En este artículo se presenta una propuesta para mejorar la calidad, en cuanto a completitud, de especificaciones de requisitos escritas en un subconjunto del español denominado español restringido, utilizando para ello un enfoque lingüístico basado en la gramática de casos.

Palabras claves: Ingeniería de requisitos, calidad de requerimientos, Procesamiento de Lenguaje Natural (PLN), gramática de casos, análisis automático de completitud de requisitos.

Abstract

Software product quality is closely linked with requirements quality from development process initial stages; proposals directed to requirements specification make incipient efforts to reach software requirements complete enough to reach the translation of user needs and expectations into the final product. In this paper we present a proposal for quality enhancement, especially in completeness, of requirements specifications

Fecha de recepción: 14 de febrero de 2005
Fecha de aceptación: 14 de mayo de 2006

* Este artículo se elaboró como resultado parcial de la tesis de maestría "Método para el reconocimiento semiautomático de operaciones a partir de Grafos Conceptuales de Sowa".

** Grupo de Investigación UN-INFO. Escuela de Sistemas. Facultad de Minas, Universidad Nacional de Colombia. cmzapata@unal.edu.co

Dirección: Universidad Nacional de Colombia, Carrera 80 N° 65-223, Bloque M8, Oficina 112, Bogotá (Colombia).

*** Departamento de Ingeniería de Sistemas, Universidad de Antioquia. afjara@udea.edu.co

**** Grupo de Investigación UN-INFO. Escuela de Sistemas. Facultad de Minas, Universidad Nacional de Colombia. farango@unal.edu.co

written in restricted Spanish, a subset from Spanish, using a linguistic Case-Grammar-based approach to reach this goal.

Key words: Requirements Engineering, Requirements quality, Natural Language Processing (NLP), Case Grammar, Automated Analysis of requirements completeness.

1. INTRODUCCIÓN

En el proceso de desarrollo de software, la etapa de elicitación de requisitos es reconocida como una etapa crítica para el éxito de los proyectos, puesto que la calidad del producto final, en gran medida, está determinada por la calidad de la especificación de requisitos [1], [2], [3]. Así mismo, las especificaciones de requisitos cumplen una tarea fundamental en el proceso de Análisis y Diseño Orientado a Objetos (ADOO), ya que a partir de ellas se lleva a cabo la identificación de los elementos que conforman el sistema [4], [5], [6]. Debido a esto, son diversas las propuestas dirigidas a mejorar la calidad de las especificaciones de requisitos, muchas de las cuales se basan en la interpretación de un lenguaje cercano al lenguaje natural¹, escritas especialmente en inglés, alemán, japonés y francés [7]. Sin embargo, se desconocen propuestas definidas para especificaciones escritas en español.

En este artículo se presenta una propuesta para mejorar la calidad, a nivel de completitud, de especificaciones de requisitos escritas en *español restringido*. La propuesta está basada en la *gramática de casos* planteada por Fillmore [8] y en un diálogo con el usuario que permite adquirir la información ausente detectada en la especificación.

Este artículo está organizado así: la sección 2 se dedica a la presentación de algunos trabajos relacionados; los planteamientos lingüísticos que sustentan la propuesta en relación con la gramática de casos se describen en la sección 3; posteriormente, en la sección 4 se presenta la propuesta, y en la 5 se plantea un caso de estudio mediante el cual se examinan los principales resultados. Las conclusiones y los trabajos futuros que se generan a partir de esta propuesta se presentan en las secciones 6 y 7 respectivamente.

¹ En el resto del artículo, cuando se hable de especificaciones de requisitos se refiere concretamente a aquellas especificaciones que se obtienen de la interpretación de un lenguaje cercano al lenguaje natural.

2. TRABAJOS RELACIONADOS

En esta sección se aborda una descripción general de propuestas tendientes a mejorar la calidad de especificaciones de requisitos, agrupadas por las limitaciones que exhiben. (Información adicional de ellas puede encontrarse en [7]).

2.1. Manejo de clasificaciones de verbos y estructuras propias carentes de generalidad

- Ohnishi (*Customizable Software Requirements Languages*) utiliza un lenguaje para la definición de requerimientos en idioma japonés restringido, al cual denomina X-JRDL, que se basa en casos semánticos y procura proveer una estructura que guíe al ingeniero de requisitos en la especificación. Sin embargo, la clasificación que realiza de los verbos es una clasificación propia y no tiene en cuenta que los verbos pueden tener diferentes sentidos dependiendo de los sustantivos que los acompañen.
- Rolland y Proix (*A Natural Language Approach for Requirements Engineering*) definen el modelo conceptual del sistema propuesto bajo un enfoque lingüístico que busca la formalización de mecanismos lingüísticos, los cuales son usados por el ingeniero de requisitos para abstraer fenómenos del mundo real. Su limitación es que utiliza una estructura definida para la determinación de los casos semánticos presentes en las especificaciones de requisitos, pero la frase podría no cumplir alguno de los patrones estructurales que define.

2.2. No se verifica la ausencia de información

- Wilson, Rosenberg y Hyatt (*Automated Quality Analysis of Natural Language Requirements Specifications*) abordan aspectos de calidad de los requerimientos mediante una herramienta llamada ARM (*Automated Requirements Measurement*), que evalúa la estructura del documento de requerimientos, así como la estructura de las oraciones individuales y el vocabulario utilizado. Esta propuesta no está dirigida a la detección de información ausente en la especificación.
- Rolland y Proix, cuya propuesta se describe en el numeral anterior, no emplean los patrones estructurales para evaluar la completitud de los requisitos.

- Fabbrini, Fusani, Gnesi y Lami (*Quality Evolution in Software Requirements Specifications*) establecen aspectos de calidad para las especificaciones planteadas en lenguaje natural, diferenciando entre aspectos de calidad de las oraciones y de todo el documento. Esta propuesta asume que las frases no presentan ausencia de información.
- Barr (*Identifikation von Spezifikationsmustern im Echtzeitentwurf Anhand der Fallstudie Antiblockiersystem*) plantea la detección de patrones de especificación, los cuales soportan la transformación de requerimientos en lenguaje natural no estructurado a un lenguaje de especificación formal, mediando para ello un lenguaje semiformal. En su enfoque se asume que la especificación no presenta ausencia de información.
- Juristo, Moreno y López (*How to Use Linguistic Instruments for ooa*) proponen un proceso que acepta cualquier especificación de requerimientos escrita en lenguaje natural, y arroja como resultado un modelo orientado a objetos; inicialmente se prepara la especificación detectando sinonimia y homonimia, además de clasificar el discurso en estático y dinámico, para luego reformatearlo de acuerdo con un lenguaje llamado *lenguaje de utilidad*, que se utiliza para la conversión final al modelo orientado a objetos. (Información adicional puede encontrarse en [9]). Sin embargo, aunque se presentan guías para dar formato a los requerimientos, no se realiza un examen de la calidad de las especificaciones a nivel de completitud.
- Dallner, Günther y Rupp (*Die Erstellung Eines Objektmodells Aus Natürlichsprachlichen Beschreibungen*) plantean un método para dar soporte a un analista de sistemas durante el proceso de identificación de requerimientos de un modelo de análisis orientado a objetos (Aoo) con descripciones en lenguaje natural, especificando algunos patrones lingüísticos que pueden ser trasladados directamente a un modelo de Aoo. Como las demás, esta propuesta no evalúa la completitud de las especificaciones.
- Bryant (*Object Oriented Natural Language Requirements Specification and Linguistically Based Requirements Engineering*) plantea una metodología para convertir una notación en lenguaje natural a un diseño orientado a objetos que se basa en la teoría de gramáticas de dos niveles (*Two-Level Grammars*, TLG), en la cual el proceso de generación del diseño oo puede ser automatizado porque la especificación TLG es lo suficientemente formal para permitirlo. Esta propuesta asume que la especificación de requisitos no presenta ausencia de información.

- Fliedl, Kop, Mayerthaler, Mayr y Winkler (*Linguistic Aspects of Dynamics in Requirements Specifications and Linguistically Based Requirements Engineering*) plantean cómo, a partir de especificaciones en lenguaje natural, en el campo de los sistemas de información se llega a una especificación intermedia denominada KCPM (*Klagenfurt Conceptual Predesign Model*), de la cual puede derivarse el diseño OO (UML, OMT o ER) mediante reglas [10]. Pese a lo completo de su método, los autores no realizan una evaluación de las especificaciones a nivel de completitud, y la ausencia de esta información sólo se percibe de manera indirecta a través de la ausencia de los elementos que conforman los diagramas.

2.3. No se concibe que el proceso pueda ser automatizable

- Götz y Rupp (*Regelwerk–Natürlichsprachliche Methode*) se enfocan en la detección de defectos en las especificaciones elaboradas en lenguaje natural y en la definición de reglas que permitan establecer si una sentencia es correcta o no. Sin embargo, aunque se presentan reglas que sirven como guías para la evaluación de la especificación, éstas aparecen sólo como derroteros que debe seguir el analista a modo de lista de chequeo, sin considerar la posible automatización del proceso.
- Dallner Günther y Rupp, cuya propuesta se describe en el numeral anterior, tienen un método concebido para dar soporte al analista en el proceso de modelamiento orientado a objetos, pero no para que sea soportado por una herramienta.

Como se puede apreciar, los métodos anteriormente expuestos presentan limitaciones a nivel de análisis de completitud de las especificaciones. Así mismo, se desconocen propuestas en este sentido basadas en el idioma español. En la siguiente sección se presentan los conceptos que sustentan la investigación que motiva este artículo.

3. GRAMÁTICA DE CASOS

En la gramática de casos, propuesta por Fillmore [8], el verbo es el foco de una oración, y los sustantivos que lo acompañan se encuentran en un tipo de relación particular con el verbo que se denomina *caso*. Cada verbo es clasificado de acuerdo con su marco de casos, es decir, con el conjunto particular de casos que deben ocurrir cada vez que el verbo es usado en una oración. No existe consenso en cuanto al conjunto de casos que debe ser utilizado para definir un marco de casos; sin embargo, es aceptado que un número pequeño de casos

es suficiente para modelar todos los verbos en el lenguaje [8], [11]. Además, está estipulado que, con pocas excepciones, el mismo caso no puede ocurrir más de una vez en una misma oración [12].

En la tabla 1 se describen los principales casos utilizados en la propuesta.

Tabla 1
Casos utilizados en la propuesta

Caso	Definición
Agnt	Agente: Persona o cosa que realiza un evento.
Instr	Instrumento: Objeto inanimado que un agente utiliza para llevar a cabo un evento. Puede ser parafraseado mediante la palabra <i>usando</i> .
Exp	Experimentador: Entidad que recibe, acepta, experimenta o sufre el efecto de una acción.
Thm	Tema: Entidad que es movida por la acción o evento denotado por el predicado.
Perc	Percibido: Hace referencia a la entidad percibida; a menudo es apareado con <i>exp</i> .
Go	Objetivo: Lugar al cual se mueve algo, o cosa hacia la cual se dirige la acción.
Loc	Ubicación: Identifica el lugar o la orientación espacial de un estado o acción.

El marco de casos de cada verbo presenta casos *obligatorios*, aquellos sin los cuales la oración carecería de sentido, y *opcionales*, los cuales modifican la oración haciéndola más específica en tiempo, lugar, modo, etc. En el ejemplo siguiente, tomado de [13], para la oración “John va a Boston en bus” se tienen los siguientes casos para el verbo *ir*:

Casos obligatorios

Tema (Thm): John

Objetivo (Go): Boston

Caso opcional

Instrumento (Instr): Bus

El marco de casos correspondiente a un verbo deberá estar disponible en un lexicón² junto con información de las frases nominales (sustantivos) que

² Diccionario computacional en el que aparecen las palabras con sus respectivos aspectos sintácticos y semánticos asociados de acuerdo con los criterios definidos para su composición. En este caso interesan los casos semánticos. En esta propuesta se tomará como base el lexicón desarrollado por la profesora Bonnie Dorr en la Universidad de Maryland [14].

aparecen en las oraciones, estableciendo para cada una de ellas los posibles casos que podrían desempeñar en un momento determinado. Información de las preposiciones que preceden a las frases nominales también hace parte del lexicón, puesto que éstas presentan indicios sobre la presencia de ciertos casos semánticos. Por ejemplo, la preposición *con* generalmente precede al caso *instr.* Esta información servirá de insumo para el análisis semiautomático de los casos que hacen parte de las oraciones, permitiendo determinar los casos ausentes en ellas.

4. DESCRIPCIÓN DE LA PROPUESTA

Dada una especificación de requisitos, para cada oración se realizan las siguientes etapas:

4.1. Análisis sintáctico

Mediante este proceso se valida que la oración pertenezca a la gramática que define un subconjunto del español (al cual se ha denominado *español restringido*). Además, se identifican las categorías (artículo, frase nominal, verbo, preposición, etc.) de cada uno de los constituyentes de la oración.

4.2. Determinación del sentido del verbo³

El sentido de un verbo no es único. Por ejemplo, el verbo *adquirir* puede tener, entre otros, el sentido de aprender (si lo que se adquiere es información) o comprar (si lo que se adquiere es un bien). En un lexicón se encuentran los posibles sentidos que puede tener un verbo y los casos asociados a cada uno de estos sentidos. En la tabla 2 se presenta un ejemplo con información del verbo *lograr* extraída de [14].

Mediante los casos semánticos presentes en la oración es posible establecer de manera semiautomática el sentido del verbo [15, 16 y 17]. Sin embargo, éste no es un proceso trivial, por lo tanto se pedirá al usuario que elija, de los posibles sentidos que ofrece el lexicón, aquel que corresponde al sentido de utilización del verbo en la oración.

³ Esto supone que cada oración sólo puede tener un verbo; es decir, debe omitirse el uso de verbos auxiliares.

Tabla 2
 Información del verbo *lograr* extraída de [14]

Verbo	Sentido	Casos obligatorios	Casos opcionales
Lograr	Get	Agnt, thm	Src(), ben(por)
	Attain	Agnt, thm	Src(), ben(por)
	Achieve	Agnt, thm	Src(), ben(por)
	Accomplish	Agnt, thm	Src(), ben(por)
	Obtain	Agnt, thm	Src(de), ben(por)
	Succeed in	Thm, loc	

4.3. Determinación de casos ausentes

Cada oración puede estar constituida por casos obligatorios y opcionales; inicialmente se da prioridad al apareamiento de casos obligatorios (ya que sin ellos la oración carece de sentido) y posteriormente se procede al apareamiento de los casos opcionales, los cuales deberán presentarse puesto que brindan información adicional del sistema. El proceso es análogo; sólo que en la determinación de los casos opcionales ausentes se tienen en cuenta las identificaciones realizadas para los casos obligatorios ausentes.

Para cada frase nominal de la oración se realizan los siguientes pasos:

- Se establece el sustantivo principal en la frase nominal.
- Con ayuda del lexicón se determinan los casos en los que puede participar el sustantivo.
- Se realiza la intersección o apareamiento de los casos del sustantivo y los casos obligatorios del verbo.
- Si la intersección da como resultado el conjunto vacío, el sistema deberá interrogar al usuario para que éste ingrese la información faltante. De lo contrario, siempre y cuando el caso no haya sido apareado con otra frase nominal, se almacena el caso que cumple la frase nominal en la oración (*agnt, loc, exp, etc.*). Considerando que un caso normalmente ocurre una sola vez en cada oración, esta información facilita el apareamiento de los casos correspondientes a las otras frases nominales de la oración.

- Si existen casos obligatorios ausentes. Luego de que el usuario ingrese la información requerida, se repite todo el proceso hasta que los casos obligatorios sean apareados.

Una vez apareados todos los casos obligatorios, puede llevarse a cabo el análisis de casos opcionales ausentes.

5. CASO DE ESTUDIO

Para ilustrar la propuesta, a continuación se presenta un caso de estudio adaptado de [18].

Especificación de requisitos en términos del escenario *Retiro de dinero*.

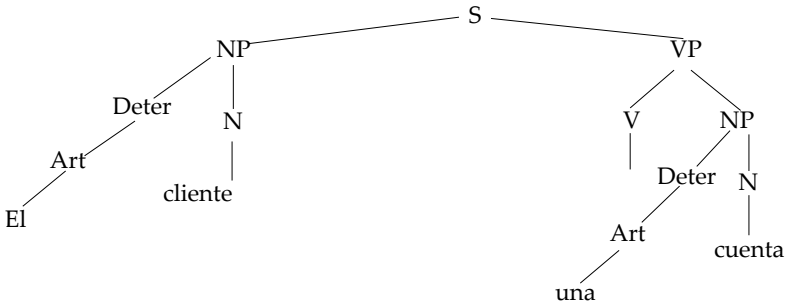
1. El cliente tiene una cuenta
2. El cliente conoce su clave
3. El sistema inicia el retiro
4. El cliente ingresa la cuenta
5. El sistema toma la cuenta y la clave
6. El cliente ingresa la cantidad desde el teclado
7. El sistema toma la cantidad
8. El sistema genera la información del retiro
9. El banco toma la información del retiro
10. El banco aprueba el retiro
11. El sistema dispensa el dinero
12. El retiro finaliza

Con el ánimo de mostrar el proceso, sólo se tomarán las cuatro primeras oraciones. Cada oración es sometida a los procesos especificados en la sección anterior. Se obtuvieron los siguientes resultados:

1. *Análisis sintáctico*

Se realiza cumpliendo con las reglas definidas por la gramática⁴, y arroja como resultado los árboles sintácticos que catalogan los constituyentes de cada una de las oraciones. En la figura que aparece a continuación se detalla el árbol sintáctico correspondiente a la frase N° 1. De forma análoga se realizan los árboles correspondientes a las demás frases.

⁴ Con el ánimo de hacer énfasis en el análisis de casos, se omiten los detalles relacionados con la gramática que restringe el español y el análisis sintáctico.



Las convenciones son: S = Oración, NP = Frase Nominal, Deter = Determinante, Art = Artículo, N = Sustantivo, VP = Frase Verbal, V = Verbo.

Figura 1. Arbol sintáctico correspondiente a la frase
El cliente tiene una cuenta.

2. Determinación del sentido del verbo

En la tabla 3 se ejemplifica el proceso para el verbo *tener*; de manera similar se procede con los verbos de las otras oraciones, extrayendo del lexicón la información requerida para llevar a cabo el proceso.

Tabla 3
Información del verbo *tener* extraída de [14]

Verbo	Sentido	Casos obligatorios	Casos opcionales
Tener	Get	Agnt, thm	src, ben(por)
	Get	Agnt, ben, thm	Src
	Hold	Agnt, thm	instr(por)
	Hold	Agnt, thm	Loc
	Hold	Exp, perc, mod-prop(a)	
	Hold	Prop(que)	
	Possess	Thm, loc	
	Have	Thm, loc	
	Hold	Thm, poss	
	Hold	Agnt, thm, loc	

Luego de esto, se presenta al usuario un menú con los sentidos del verbo *tener* (*get, hold, possess, have*) y se le solicita que elija el sentido que mejor se ajuste a la sentencia. Con esto, en lo que se refiere a este ejemplo, se obtiene

el sentido *possess*, que no presenta casos opcionales y cuyos casos obligatorios son *thm* y *loc*. De manera análoga se procede con los verbos de las otras oraciones. La información de los verbos luego de establecer sus sentidos se presenta en la tabla 4.

Tabla 4

Información de los verbos luego de establecer sus sentidos

Verbo	Casos obligatorios	Casos opcionales
Tener	Thm, loc	
Conocer	Exp, perc	
Iniciar	Agnt	Thm, go
Ingresar	Agnt, exp	Instr

3. Determinación de casos ausentes

Para llevar a cabo esta tarea se requiere información sobre los casos asociados a cada uno de los verbos, la cual ha sido obtenida en la etapa anterior, e información sobre los casos que podría desempeñar cada frase nominal en una sentencia, la cual es suministrada por el lexicón, tal como se compendia en la tabla 5.

Tabla 5

Información de las frases nominales extraída del lexicón

Frase nominal	Casos posibles ⁵
Cliente	Agnt, exp, loc, perc
Cuenta	Thm, exp, perc
Clave	Thm, exp, perc
Retiro	Thm, exp, perc
Sistema	Agnt, exp, loc, perc

Luego se procede con el apareamiento de los casos:

⁵ Los casos que una frase nominal podría desempeñar en una frase están ordenados de acuerdo con la probabilidad de ocurrencia de ellos. Por ejemplo, en la frase nominal *cliente* es más probable que éste aparezca como *agnt* que como *exp*. De igual manera, es más probable que aparezca como *exp* que como *loc* y así sucesivamente. Esta disposición de los casos facilita su apareamiento.

3.1. Determinación de casos obligatorios ausentes

3.1.1. Sentencia 1: *El cliente tiene una cuenta*

Verbo: Tener		Frase nominal: Cliente
Casos obligatorios: Thm, loc	\cap	Posibles casos: Agnt, exp, loc, perc
Resultado del apareamiento: Cliente	=	Loc

El apareamiento de casos da como resultado que *cliente* desempeña el caso *loc* en la sentencia. Por lo tanto, el caso *loc* no puede presentarse nuevamente en la sentencia.

Verbo: Tener		Frase nominal: Cuenta
Casos obligatorios: Thm, loc	\cap	Posibles casos: Thm, exp, perc
Resultado del apareamiento: Cuenta	=	Thm

El verbo *tener* presenta los casos obligatorios *thm*, *loc*, y no tiene casos opcionales, por lo tanto puede omitirse la evaluación de casos opcionales y se concluye que la sentencia no presenta casos ausentes.

3.1.2. Sentencia 2: *El cliente conoce su clave*

Verbo: Conocer		Frase nominal: Cliente
Casos obligatorios: Exp, perc	\cap	Posibles casos: Agnt, exp, loc, perc
Resultado del apareamiento: Cliente	=	Exp

Verbo: Conocer		Frase nominal: Clave
Casos obligatorios: Exp, perc	\cap	Posibles casos: Thm, exp, perc
Resultado del apareamiento: Cliente	=	Perc

Se puede observar que en primera instancia el caso que aparee es *exp*. Sin embargo, éste debe ser descartado debido a que ya ha sido asignado. Los casos obligatorios han sido determinados y el verbo no presenta casos opcionales, por lo tanto puede concluirse que no existen casos ausentes en la sentencia.

3.1.3. Sentencia 3: *El sistema inicia el retiro*

Verbo: Iniciar		Frase nominal: Sistema
Casos obligatorios: Agnt	\cap	Posibles casos: Agnt, exp, loc, perc
Resultado del apareamiento: Sistema	=	Agnt

El único caso obligatorio *agnt* ha sido apareado. Así mismo, el verbo presenta casos opcionales que se deben aparear en la siguiente fase.

3.1.4. Sentencia 4: *El cliente ingresa la cuenta*

Verbo: Ingresar		Frase nominal: Cliente
Casos obligatorios: Agnt, exp	\cap	Posibles casos: Agnt, exp, loc, perc
Resultado del apareamiento: Cliente	=	Agnt o exp?

Los casos *agnt* y *exp* han sido apareados y en esto se presenta ambigüedad. Por ello, se posterga la elección del caso con el fin de realizar el apareamiento de las otras frases nominales. Si una vez realizado dicho apareamiento la ambigüedad persiste, ésta podría ser resuelta teniendo en cuenta las posibilidades de ocurrencia de los casos (en este caso *agnt* tiene una mayor posibilidad de ocurrencia que *exp*).

Verbo: Ingresar		Frase nominal: Cuenta
Casos obligatorios: Agnt, exp	\cap	Posibles casos: Thm, exp, perc
Resultado del apareamiento: Cuenta	=	Exp

El apareamiento de *cuenta* corresponde a *exp*, lo cual resuelve la ambigüedad presentada anteriormente:

Verbo: Ingresar		Frase nominal: Cliente
Casos obligatorios: Agnt, exp	\cap	Posibles casos: Agnt, exp, loc, perc
Resultado del apareamiento: Cliente	=	Agnt

La sentencia no presenta omisión de casos obligatorios. Los casos opcionales que presenta el verbo serán apareados en la siguiente fase.

3.2. Determinación de casos opcionales ausentes

3.2.1. Sentencia 3: *El sistema inicia el retiro*

Considerando que Sistema = Agnt, se realizará la evaluación para las restantes frases nominales.

Verbo: Iniciar		Frase nominal: Retiro
Casos opcionales: Thm, go	\cap	Posibles casos: Thm, exp, perc
Resultado del apareamiento: Retiro	=	Thm

Todas las frases nominales de la sentencia (*sistema* y *retiro*) han sido apareadas. Sin embargo, la frase presenta la ausencia del caso opcional *go*.

3.2.2. **Sentencia 4:** *El cliente ingresa la cuenta*

Debido a que Cliente = Agnt y Cuenta = Exp, no existen otras frases nominales por evaluar. Por lo tanto, la frase presenta la ausencia del caso opcional *instr*.

En resumen, las oraciones 1 y 2 no presentan casos ausentes. La sentencia 3 presenta la ausencia del caso opcional *go*. Éste deberá ser establecido, para lo cual se interrogará al usuario de la siguiente manera:

- El sistema inicia el retiro: ¿De qué?

De manera similar, la sentencia 4 puede mejorar su completitud si se establece el caso ausente *instr*. De nuevo se interrogará al usuario:

- El cliente ingresa la cuenta: ¿A través de qué medio?

6. CONCLUSIONES

La calidad de los productos finales de software está directamente relacionada con la calidad de las especificaciones de requisitos establecidas desde las primeras etapas del proceso de desarrollo. De manera particular, la completitud de las especificaciones es uno de los aspectos que tienen mayor impacto en el Análisis y Diseño Orientado a Objetos (ADOO), en la medida en que facilita el reconocimiento de los elementos del sistema. En este artículo se ha presentado una propuesta para mejorar la completitud de especificaciones de requisitos escritas en español restringido. La propuesta está basada en un tratamiento lingüístico de las oraciones, haciendo uso de la gramática de casos propuesta por Fillmore [8], con el fin de detectar la información ausente en la especificación, y en un diálogo con el usuario para adquirir la información faltante.

La gramática de casos ha mostrado ser una poderosa herramienta en este proceso. Sin embargo, se reconocen retos importantes relacionados con el tratamiento de la ambigüedad en la determinación del sentido del verbo en cada sentencia y el apareamiento de casos.

Aunque existen trabajos relacionados, de los cuales se presenta una revisión general, se desconocen propuestas, en este sentido, para especificaciones

escritas en español. Así mismo, la definición de los recursos léxicos, a utilizar en combinación con la gramática de casos, y el método de apareamiento de casos son aspectos destacables de la investigación. Se presentó también un caso de estudio que arroja un análisis de los casos ausentes y puede servir de base para la elaboración de una herramienta computacional que contribuya a mejorar la completitud de los requisitos.

7. TRABAJOS FUTUROS

Como resultado de esta investigación, quedan planteados temas que pueden ser abordados en futuros proyectos, entre los que se cuentan:

- Automatizar el método de identificación del sentido de los verbos.
- Construir un lexicón especialmente diseñado para el apareamiento de casos del español bajo los postulados de la gramática de casos.
- Elaborar los casos de uso de un posible prototipo en el cual se implemente la propuesta, y desarrollar ese prototipo.
- Definir los mecanismos para generar el diálogo que busca la completitud de los requisitos.

REFERENCIAS

- Boehm, B.W., Clark, B., Horowitz et al. Cost models for future life cycle processes: Cocomo 2.0. *Annals of software engineering*. 1995.
- The Standish Group Report. Chaos. Standish Group Internal Report. 1995. Available: http://www.projectsmart.co.uk/docs/chaos_report.pdf [Citado Febrero 14 de 2005].
- Mich, L., Garigliano, R. NL-OOPS: A requirements analysis tool based on natural language processing. *Proceedings of the 3rd International Conference on Data Mining 2002*. Bologna. P321-330. 2002.
- Buchholz y Düsterhöft, Buchholz, E. Düsterhöft A. Using natural language for Database Design. *Proceedings Deutsche Jahrestagung für Künstliche Intelligenz*, Saarbrücken. 5p. 1994.
- Kristen, G. Object Orientation: The KISS Method. *From information architecture to information systems*. Addison- Wesley, Wokingham. 1994.
- Harmain y Gaizauskas. Harmain, H., Gaizauskas, R. CM-Builder: An automated NL-Based CASE Tool. *Proceedings of the fifteenth IEEE International Conference on Automated Software Engineering (ASE '00)*. Grenoble. 9 p. 2000.

- Fraunhofer IESE (Fiese), *A Survey on Approaches for Writing Precise Natural Language Requirements*. Fraunhofer Institut Experimentelles Software Engineering. 2001.
- Fillmore Ch. *The Case for Case*. In *Universals in Linguistics Theory*. E. Bach. R. Harms eds. Holt, Rinehart and Winston Publishing Company. P1-90. 1968.
- Juristo N., Morant José L. Moreno Ana M. *A formal approach for generating OO specifications from natural language*. Facultad de Informática. Universidad Politécnica de Madrid. 1997.
- Mayr, H., Kop, Ch. *A User Centered Approach to Requirements Modelling*. University of Klagenfurt. 2002.
- Cook W. *Case Grammar Theory*. Georgetown University Press. Washington D.C. 1989.
- Belkhouche, B. y Kozma, J. *Semantic Case Analysis of Informal Requirements*. S. Brinkkemper and F. Harmsen (eds.), *Memoranda Informatica 93-32*, Universiteit of Twente - The Netherlands. p163-182. 1993.
- Sowa J. F. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, Reading, MA. 1984.
- Dorr, B.J. *Computational Lexicon*. Copyright (c) University of Maryland. 2001.
- Green, R., Pearl, ., Dorr, B.J. Resnik, P. *Lexical resource integration across the syntax-semantics interface*. Institute for Advanced Computer Studies. Department of Computer Science. College of Information Studies. University of Maryland. 2001.
- Green, R., Pearl, L., Dorr, B.J. *Mapping WordNet senses to a lexical database of verbs*. Institute for Advanced Computer Studies. Department of Computer Science. University of Maryland. 2001.
- Green R, Pearl L, Dorr BJ. *Mapping lexical entries in a verbs database to WordNet senses*. Institute for Advanced Computer Studies. Department of Computer Science. University of Maryland. 2001.
- Dong L. Kalaivani Subramaniam. Far Behrouz H., Eberlein A. *Automatic transition from use-cases to class model*. Department of Electrical and Computer Engineering. University of Calgary. 2003.