

Elicitación de Requisitos de Usuario Mediante Plantillas y Patrones de Requisitos*

A. Durán Toro, B. Bernárdez Jiménez, M. Toro Bonilla y A. Ruiz Cortés

Departamento de Lenguajes y Sistemas Informáticos, Facultad de Informática y Estadística,
Universidad de Sevilla, Avda. Reina Mercedes s/n, 41012 Sevilla, España
{amador,beat,mtoro,aruiz}@lsi.us.es

Resumen Dentro de la ingeniería de requisitos se han identificado diversos tipos de problemas, uno de los cuales es expresar los requisitos de forma que todos los participantes sean capaces de comprenderlos, especialmente los que no son profesionales informáticos. Para ello, lo habitual es usar lenguaje natural que, por lo general, es el único lenguaje común entre todas las comunidades involucradas en el proceso. Los problemas que presenta el uso del lenguaje natural son de sobra conocidos, pero usar notaciones más formales demasiado pronto en esta fase de desarrollo es una decisión de gran riesgo que puede provocar que los requisitos sean imposibles de entender para los clientes y usuarios. Por otro lado, incluso un uso semánticamente correcto del lenguaje natural no garantiza la comprensión entre todos los participantes si se utiliza un estilo excesivamente técnico o los responsables de la redacción de los requisitos no poseen las oportunas habilidades de redacción.

En este artículo se propone el uso de plantillas y patrones de requisitos para mejorar tanto el propio proceso de elicitación como la comprensión de los requisitos. Se han identificado dos tipos de patrones: *patrones lingüísticos*, que son frases en lenguaje natural fáciles de entender, que se usan con frecuencia en descripciones de requisitos, y que pueden parametrizarse y utilizarse para rellenar las plantillas de requisitos, y *patrones de requisitos*, que son plantillas de requisitos genéricos que suelen aparecer con bastante frecuencia durante el proceso de elicitación y que pueden reutilizarse adaptándolos a cada caso particular.

Palabras clave: ingeniería de requisitos, elicitación de requisitos, patrones de requisitos

1 Introducción

La elicitación de requisitos puede definirse como la actividad de la ingeniería de requisitos en la que los clientes y usuarios de un sistema software descubren, revelan, articulan y comprenden sus requisitos [11]. Este proceso no está exento de problemas. Según [4], los problemas de la elicitación de requisitos pueden clasificarse en tres grandes grupos: problemas de *ámbito*, es decir, decidir los límites del sistema, problemas de *volatilidad*, ya que los requisitos evolucionan en el tiempo, y problemas de *comprensión* entre los distintos participantes involucrados en el proceso.

* Este trabajo está financiado por el proyecto "MENHIR" de la CICYT. TIC 97-0593-C05-01, Ministerio de Educación y Ciencia.

Uno de los principales problemas de comprensión es expresar los requisitos de forma que puedan entenderse no sólo por los ingenieros de requisitos, sino también por clientes y usuarios no profesionales en informática, algo que no suele abordarse por las técnicas de elicitación habituales como las entrevistas, el JAD o el *brainstorming* [11].

La elección habitual para expresar los requisitos elicitados es el lenguaje natural, ya que frecuentemente es el único lenguaje común entre clientes, usuarios e ingenieros de requisitos. Los problemas del lenguaje natural son de sobra conocidos, pero utilizar notaciones más formales demasiado pronto tiene el riesgo de que los requisitos sean imposibles de entender por parte de los clientes y usuarios. A veces, incluso usando lenguaje natural correcto, si los responsables de redactarlos no se esmeran en su redacción, los requisitos no se entienden por un estilo excesivamente técnico o farragoso.

En este artículo se presentan plantillas y patrones de requisitos que pueden mejorar tanto el propio proceso de elicitación como la comprensión de los requisitos, ayudando a los ingenieros de requisitos y a los clientes y usuarios a expresar requisitos para sistemas de información usando lenguaje natural. Se han desarrollado plantillas para tres grandes grupos de requisitos: de almacenamiento de información, funcionales y no funcionales. También se han identificado dos tipos de patrones: *patrones lingüísticos (patrones-L)*, que son frases fáciles de entender muy utilizadas en las descripciones de requisitos en lenguaje natural, y *patrones de requisitos (patrones-R)*, que son plantillas de requisitos genéricos que aparecen muy frecuentemente durante el proceso de elicitación de requisitos y que pueden reutilizarse adaptándolas a cada caso concreto.

La organización del artículo es la siguiente. El modelo de procesos de ingeniería de requisitos que se asumirá durante el resto del artículo se presenta en la sección 2. En la sección 3 se presentan las plantillas y patrones para los tres grupos de requisitos de sistemas de información identificados. En la sección 4 se comparan otros trabajos relacionados y, por último, en la sección 5 se exponen las conclusiones y se indican los posibles trabajos futuros.

2 Modelo de procesos de ingeniería de requisitos

Durante el resto del artículo se asumirá el modelo de procesos de ingeniería de requisitos que se muestra en la figura 1, cuya principal característica es la iteratividad, y en el que se definen tres actividades principales: *elicitación*, *análisis* y *validación*.

En el modelo se definen dos posibles ciclos de iteración. El ciclo principal, *elicitación-análisis-validación*, indica la posibilidad de que durante el proceso de validación de los requisitos por parte de los clientes y usuarios se descubran errores o se eliciten nuevos requisitos que hasta entonces estaban ocultos. En esas circunstancias es necesario corregir dichos defectos mediante nuevas reuniones de elicitación, y a continuación repetir las actividades de análisis y validación.

El segundo ciclo, *elicitación-análisis*, se trata de un ciclo interno que indica la posibilidad de que durante la realización del análisis de los requisitos elicitados se descubran errores, lo que puede provocar la necesidad de nuevas reuniones de elicitación y el posterior análisis de sus resultados. Las características principales de las tres actividades (ver figura 2) son las siguientes:

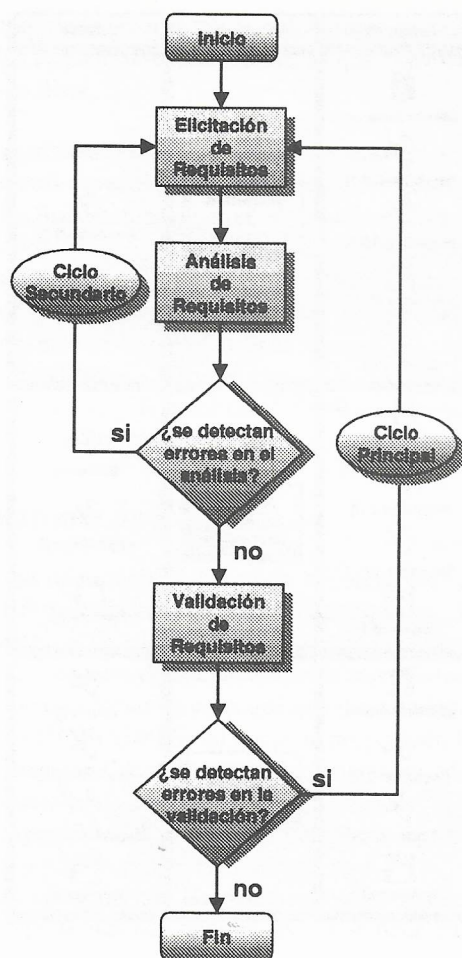


Figura1. Modelo de procesos de ingeniería de requisitos: iteración de actividades

- **Elicitación de requisitos:** en esta actividad se deben obtener los requisitos a partir de la información proporcionada por clientes y usuarios mediante técnicas de elicitación ya mencionadas anteriormente. Además de dicha información, las entradas de esta actividad son los requisitos expresados en el lenguaje de los clientes y usuarios (requisitos-C), los requisitos expresados en el lenguaje de los desarrolladores (requisitos-D), el prototipo y los errores resultado de la iteración anterior del proceso¹. La salida de esta actividad es una nueva versión (o la primera si se trata de la primera iteración) de los requisitos-C, que serán analizados en la siguiente actividad.

¹ Los términos requisitos-C y requisitos-D se han tomado de [2]. Por otro lado, en la figura 2 se asume que la iteración actual es la i -ésima, de forma que los productos de la iteración anterior tienen el subíndice $i-1$ y los de la iteración actual i .

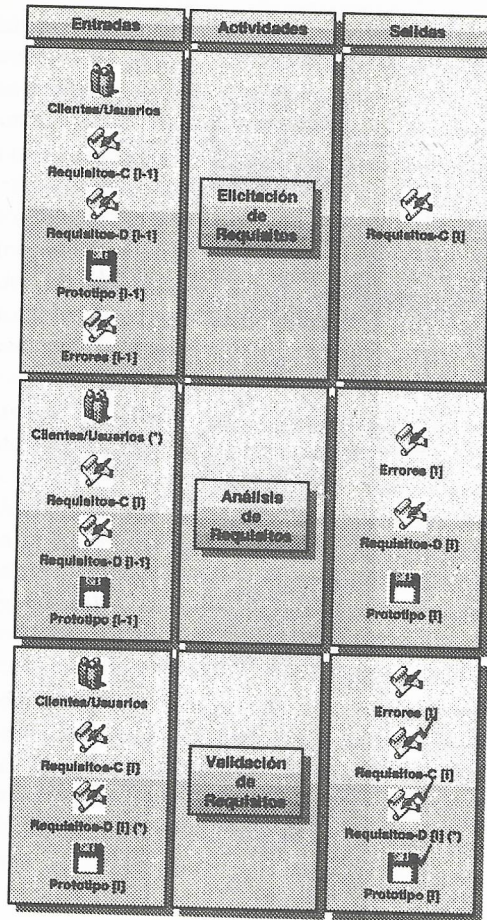


Figura2. Modelo de procesos de ingeniería de requisitos: entradas–actividades–salidas

- **Análisis de requisitos:** en esta actividad se debe razonar sobre los requisitos elicitados para detectar errores e identificar nuevos requisitos, normalmente mediante la construcción de modelos, en la que podrían participar aquellos clientes y usuarios con los conocimientos apropiados². Otros autores [11], añaden una actividad diferenciada del análisis a la que denominan *especificación*, en la que se registran los requisitos en uno o más documentos. En el modelo propuesto se ha optado por asumir que el proceso de registrar los requisitos es implícito a la elicitación en el caso de los requisitos–C, y al análisis en el caso de los requisitos–D. Las entradas de esta actividad son los requisitos–C elicitados en la actividad anterior, los requisitos–D y el prototipo de la iteración previa e información proveniente de clientes y usuarios en el caso de que tengan conocimientos de la técnicas apropiadas. La salida de esta actividad es una nueva versión (o la primera si se trata de

² Indicado en la figura 2 por un asterisco en el icono de clientes y usuarios.

la primera iteración) de los requisitos-D y del prototipo, así como posibles errores encontrados en los requisitos-C analizados, lo que provocaría que se repitiese el ciclo elicitación-análisis.

- **Validación de requisitos:** en esta actividad los clientes y usuarios deben confirmar que los requisitos especificados son válidos, correctos y completos, mediante inspecciones de los documentos generados y mediante la evaluación del prototipo, que por lo general conlleva la elicitación de nuevos requisitos. Las entradas de esta actividad son los requisitos-C, los requisitos-D (en el caso de que los clientes y usuarios tengan conocimientos suficientes para comprender las notaciones utilizadas³), el prototipo y la información de validación proveniente de clientes y usuarios. La salida de esta actividad son las versiones validadas, total o parcialmente, de los requisitos-C, requisitos-D y del prototipo⁴. En el caso de que se detecten errores o se eliciten nuevos requisitos se repite el ciclo completo de actividades.

3 Plantillas y patrones de requisitos

Las plantillas y patrones de requisitos que se presentan en los siguientes apartados están pensadas para utilizarse durante la elicitación de requisitos, bien reutilizando plantillas ya rellenas (patrones-R) o rellenándolas completando huecos de frases ya escritas (patrones-L) durante las sesiones de elicitación, bien como forma de registrar y gestionar los requisitos-C. De esta forma se consigue que durante las sesiones de elicitación se trabaje con una filosofía WYSIWYG, tal como se propone en la técnica JAD [11], ya que los participantes manejan directamente la documentación final, lo que favorece su implicación en el proceso.

A continuación se describen las plantillas y los patrones para requisitos de sistemas de información. En la notación usada, las palabras entre < y > deben ser convenientemente reemplazadas, mientras que las palabras que se encuentran entre { y } y separadas por comas representan opciones de las que se debe escoger una.

3.1 Requisitos de almacenamiento de información

Plantilla y patrones-L de requisitos de almacenamiento de información Lo más importante en los sistemas de información es precisamente la información. La plantilla para requisitos de almacenamiento de información (figura 3) ayuda al usuario a responder a la pregunta "*¿qué información, relevante para los objetivos de su negocio, debe ser almacenada por el sistema?*". En la figura 4 puede verse un ejemplo de su uso, suponiendo como sistema un vídeo-club. El significado de los campos de la plantilla es el siguiente:

- **Identificador y nombre descriptivo:** siguiendo las recomendaciones del IEEE [1], cada requisito se debe identificar por un código único y un nombre descriptivo. Con objeto de conseguir una rápida identificación, los identificadores de los requisitos de almacenamiento de información comienzan con *RI*.

³ Expresado en la figura 2 por asteriscos en los iconos de requisitos-D.

⁴ Expresadas en la figura 2 por una marca de validación en los iconos correspondientes.



RI-<id>	<nombre descriptivo>
Versión	<nº de la versión actual> (<fecha de la versión actual>)
Autores	<autor de la versión actual> (<organización del autor>) ...
Fuentes	<fuente de la versión actual> (<organización de la fuente>) ...
Propósito	<propósito del requisito>
Descripción	El sistema deberá almacenar la información correspondiente a <concepto relevante>. En concreto:
Datos concretos	• <datos específicos sobre el concepto relevante> • ...
Intervalo temporal	{ pasado y presente, sólo presente }
Importancia	<importancia del requisito>
Urgencia	<urgencia del requisito>
Comentarios	<comentarios adicionales sobre el requisito>

Figura3. Plantilla y patrones-L para requisitos de almacenamiento de información

- **Versión:** para poder gestionar distintas versiones, este campo contiene el número y la fecha de la versión actual del requisito.
- **Autores, Fuentes:** estos campos contienen el nombre y la organización de los autores (ingenieros de requisitos) y de las fuentes (usuarios o clientes), de la versión actual del requisito.
- **Propósito:** este campo indica *porqué* el requisito es necesario para alcanzar los objetivos del negocio.
- **Descripción:** para los requisitos de almacenamiento de información este campo usa un patrón-L que se debe completar con el concepto relevante sobre el que se debe almacenar información.
- **Datos específicos:** este campo contiene una lista de los datos específicos asociados al concepto relevante.
- **Intervalo de tiempo:** este campo indica durante cuánto tiempo es relevante la información para el sistema. Puede tomar los valores *pasado y presente*, si la información es siempre relevante, o *sólo presente* si la información tiene un periodo de validez concreto. Por ejemplo, si el concepto es *empleados*, y el intervalo de tiempo es *presente y pasado*, quiere decir que los ex-empleados son relevantes para el sistema, mientras que un periodo de tiempo de *sólo presente* indicaría que los ex-empleados no se deben considerar.
- **Importancia:** este campo indica la importancia del requisito para los clientes y usuarios. Se puede asignar un valor numérico o alguna expresión enumerada como *vital*, *importante* o *quedaría bien*, tal como se propone en [10].
- **Urgencia:** este campo indica la urgencia del requisito para los clientes y usuarios. Como en el caso anterior, se puede asignar un valor numérico o una expresión enumerada como *inmediatamente*, *hay presión* o *puede esperar* [10].
- **Comentarios:** cualquier otra información sobre el requisito que no encaje en los apartados anteriores puede recogerse en este apartado.

RI-01	Información sobre películas
Versión	1.0 (12/Mayo/1999)
Autores	A. Durán (Universidad de Sevilla)
Fuentes	R. Corchuelo (Super Video Shop)
Propósito	conocer las películas que han estado o están disponibles en el vídeo-club
Descripción	El sistema deberá almacenar la información correspondiente a las películas del vídeo-club. En concreto:
Datos concretos	<ul style="list-style-type: none"> • Título de la película • Número de cintas de la película alquiladas en cada momento • Número de cintas de la película disponibles para ser alquiladas en cada momento • Tipo de la película: infantil, acción, ciencia-ficción o adultos • Duración de la película, en horas y minutos • Actores principales de la película • Director de la película • Productora de la película • Año de producción de la película
Intervalo temporal	pasado y presente
Importancia	vital
Urgencia	inmediatamente
Comentarios	ninguno

Figura4. Ejemplo de requisito de almacenamiento de información

Patrones-R de requisitos de almacenamiento de información Después de usar las plantillas y patrones en más de 40 prácticas académicas, se han identificado requisitos muy parecidos que están presentes en la mayoría de los proyectos. Para los requisitos de almacenamiento de la información se han identificado algunos patrones-R como la información sobre clientes (ver figura 5), productos, pedidos, facturas, etc. Estos patrones-R se pueden clasificar de acuerdo con diferentes criterios y almacenarse en repositorios para su posterior reutilización.

RI-x	Información sobre <i>clientes</i>
...	...
Descripción	El sistema deberá almacenar la información correspondiente a <i>los clientes del negocio</i> . En concreto:
Datos concretos	<ul style="list-style-type: none"> • Número de identificación fiscal del <i>cliente</i> • Nombre y apellidos del <i>cliente</i> • Domicilio del <i>cliente</i> • Números de teléfono del <i>cliente cliente</i> • Dirección de E-mail del <i>cliente</i>
...	...

Figura5. Ejemplo de patrón-R (*cliente*) de almacenamiento de información

RF-<id>	<nombre descriptivo>	
<i>Versión, Autores, Fuentes y Propósito igual que en la plantilla anterior</i>		
Propósito	<propósito del requisito>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando <evento de activación>	
Precondición	<precondición del caso de uso>	
Secuencia normal	Paso	Acción

	n	{El {<actor>, sistema} <acción realizada por actor/sistema>, Se realiza el caso de uso <caso de uso (RF-x)>}
	n.1	Si <condición>, {el {<actor>, sistema} <acción realizada por actor/sistema>, se realiza el caso de uso <caso de uso (RF-x)>}

Postcondición	<postcondición del caso de uso>	
Excepciones	Paso	Acción
	p	Si <condición de excepción>, {el {<actor>, sistema} <acción realizada por actor/sistema>, se realiza el caso de uso <caso de uso (RF-x)>}, a continuación este caso de uso {continúa , aborta}

Rendimiento	Paso	Cota de tiempo
	q	m segundos

Frecuencia	Este caso de uso se espera que se realice una media de <nº de veces> veces / <unidad de tiempo>	
<i>Importancia, Urgencia y Comentarios igual que en la plantilla anterior</i>		

Figura6. Plantilla y patrones-L para requisitos funcionales (casos de uso)

3.2 Requisitos funcionales

Plantillas y patrones-L de requisitos funcionales Los sistemas de información no sólo almacenan información, también deben proporcionar servicios usando la información que almacenan. La plantilla de requisitos funcionales, ver figura 6, describe casos de uso [9], y ayuda a los clientes y usuarios a responder a la pregunta "¿qué debe hacer el sistema con la información almacenada para alcanzar los objetivos de su negocio?". En la figura 7 se muestra un ejemplo en el que, por motivos de espacio, sólo se muestran los campos específicos de la plantilla, suponiendo el mismo sistema que en el ejemplo anterior. El significado de los campos específicos de esta plantilla es el siguiente:

- **Identificador y nombre descriptivo:** igual que en la plantilla anterior, excepto que los identificadores de los requisitos funcionales empiezan con *RF*.
- **Descripción:** para los requisitos funcionales, este campo contiene un patrón-L que debe completarse con el *evento de activación* que provoca la realización del caso de uso.
- **Precondición:** en este campo se expresan las condiciones necesarias para que se pueda realizar el caso de uso.

- **Secuencia normal:** este campo contiene la secuencia normal de interacciones del caso de uso. En cada paso, un actor o el sistema realiza una acción, o se realiza o se *usa* otro caso de uso, siguiendo la semántica de las relaciones *uses* y *extends* propuesta en [14]. Un paso puede tener subpasos condicionales, dentro de los cuales se puede realizar otro caso de uso, que *extendería* al caso de uso que se está describiendo [14]. Se asume que sólo se realiza un subpaso.
- **Postcondición:** aquí se expresan las condiciones que se deben cumplir después de la terminación normal del caso de uso.
- **Excepciones:** después de realizar un paso, pueden ocurrir situaciones excepcionales. Este campo especifica el comportamiento del sistema en cada una de esas circunstancias. Después de que se realice la acción o el caso de uso asociado a la excepción (otra *extensión* según [14]), el sistema puede continuar la secuencia normal o abortar el caso de uso.
- **Rendimiento:** en este campo puede especificarse el tiempo máximo para cada paso o subpaso en el que el sistema realice un acción.
- **Frecuencia:** en este campo se expresa la frecuencia esperada de realización del caso de uso, que aunque no es realmente un requisito, es una información interesante para los desarrolladores.

RF-07	Devolución de cintas de vídeo	
Propósito	Controlar la devolución de las cintas y los pagos de los clientes	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un cliente quiera devolver una o más cintas de vídeo	
Precondición	Todas las cintas son de Super Video Shop	
Secuencia normal	Paso	Acción
	1	El empleado solicita al sistema comenzar el proceso de devolución de cintas de vídeo
	2	El sistema solicita la identificación de las cintas de vídeo
	3	El empleado proporciona todas las identificaciones que se requieran
	4	El sistema calcula la cantidad e imprime la factura
	4.1	Si alguna cinta se devuelve tarde, el sistema carga un extra de un 10% por cada devolución fuera de plazo
	5	El cliente paga la factura
6	El empleado coloca las cintas en las estanterías	
Postcondición	La información almacenada se actualiza, las cintas vuelven a estar disponibles para un nuevo alquiler	
Excepciones	Paso	Acción
	3	Si algunas cintas no están registradas como alquiladas, el sistema informa de esta situación al empleado y no se incluyen dichas cintas en la factura, a continuación este caso de uso continúa
Rendimiento	Paso	Cota de tiempo
	4	5 segundos
Frecuencia	Este caso de uso se espera que se realice una media de 50 veces/día	

Figura 7. Ejemplo de requisito funcional (caso de uso)

Patrones-R de requisitos funcionales Para los requisitos funcionales se han identificado cuatro patrones-R genéricos a los que se ha denominado *CRUD* (*Create, Read, Update, Delete*) y que corresponden a los habituales procesos de altas, consultas, modificaciones y bajas. Estos patrones-R deben estar siempre presentes en cualquier sistema de información correcto, ya que la información almacenada por el sistema debe ser *creada* (ver figura 8) y *actualizada* para que esté sincronizada con su entorno. La información obsoleta debe ser *borrada* si no queremos que se desborde el espacio de almacenamiento disponible, y evidentemente, la información almacenada debe poder *consultarse*. Estos patrones-R describen los pasos habituales de estos tipos de operaciones.

RF-x	{ <i>Crear, Dar de alta, Registrar</i> } <nueva información>	
...	...	
Precondición	<nueva información> no está almacenada todavía	
Secuencia normal	Paso	Acción
	1	El <algún actor> solicita al sistema comenzar el proceso de { <i>crear, dar de alta, registrar</i> } <nueva información>
	2	El sistema solicita <nueva información>
	3	El <algún actor> proporciona <nueva información>
Postcondición	<nueva información> se almacena	
...	...	

Figura8. Ejemplo de patrón-R (*creación*) de requisito funcional

3.3 Requisitos no funcionales

Plantillas y patrones-L de requisitos no funcionales Otras características del sistema como son la privacidad, la fiabilidad, etc. se pueden expresar usando la plantilla para requisitos no funcionales de la que puede verse un ejemplo en la figura 9. El único patrón-L identificado hasta el momento para este tipo de requisitos no es más que una generalización de los usados para las descripciones de las plantillas anteriores: *El sistema deberá <capacidad del sistema>*.

RO-3	Operating System
Versión	1.0 (15/Enero/1999)
Autores	A. Durán (Universidad de Sevilla)
Fuentes	M. Toro (Super Video Shop)
Descripción	El sistema deberá operar bajo el sistema operativo Linux
Importancia	vital
Urgencia	inmediatamente
Comentarios	Comprobar compatibilidades con las diferentes versiones de Linux

Figura9. Ejemplo de requisito no funcional

Patrones-R de requisitos no funcionales El haber integrado los requisitos de rendimiento en la plantilla de requisitos funcionales ha provocado que, hasta el momento, no se hayan identificado patrones-R significativos de requisitos no funcionales. Como ejemplo, el requisito de la figura 9 puede usarse como un patrón-R para especificar el sistema operativo sobre el que tiene que funcionar el sistema si se cambiara su descripción a: *El sistema deberá funcionar sobre el sistema operativo < sistema operativo >*.

4 Trabajos relacionados

La idea de usar plantillas para expresar requisitos se basó inicialmente en las propuestas de Rumbaugh [14] y, principalmente, de Cockburn [5]. Posteriormente se introdujeron cambios teniendo en cuenta la plantilla para la especificación de requisitos *Volere* [12], que incorpora gran cantidad de requisitos no funcionales, y la plantilla para el documento de requisitos de usuario de la norma PSS-05 de la European Space Agency (ESA) desarrollada en el CERN [3]. Los patrones-L, fruto de nuestra experiencia con prácticas académicas, guardan similitud con los patrones lingüísticos propuestos en [13], basados en *case grammars*.

5 Conclusiones y trabajo futuro

En este artículo se han presentado plantillas y patrones de requisitos para sistemas de información que pueden ayudar a elicitar y expresar requisitos conservando las ventajas de usar lenguaje natural y evitando una formalización excesivamente temprana de los requisitos.

Las ideas aquí presentadas se han integrado dentro de una metodología de elicitación de requisitos presentada en [8] para la que se está desarrollando una herramienta CASE (ver figura 10). Recientemente, estas ideas han sido presentadas en [7].

Las plantillas y patrones se han aplicado satisfactoriamente en más de 40 prácticas académicas y se están utilizando actualmente en dos desarrollos reales de Sadiel S.A., una de las principales empresas andaluzas de producción de software, donde su utilización ha mejorado de forma significativa la comunicación con los clientes y usuarios. Sadiel utiliza la metodología estructurada MÉTRICA, en la que las plantillas y patrones presentados se han integrado satisfactoriamente siguiendo la propuesta presentada en [6].

Algunas líneas de trabajo futuro pueden incluir la adaptación de las plantillas con la realimentación de su aplicación en desarrollos reales, investigar nuevos patrones, especialmente para requisitos no funcionales, crear un repositorio para promover la reutilización y concluir la herramienta CASE.

Referencias

- [1] IEEE Recommended Practice for Software Requirements Specifications. IEEE/ANSI Standard 830-1993, Institute of Electrical and Electronics Engineers, 1993.
- [2] J. W. Brackett. Software Requirements. Curriculum Module SEI-CM-19-1.2, Software Engineering Institute, Carnegie Mellon University, 1990.



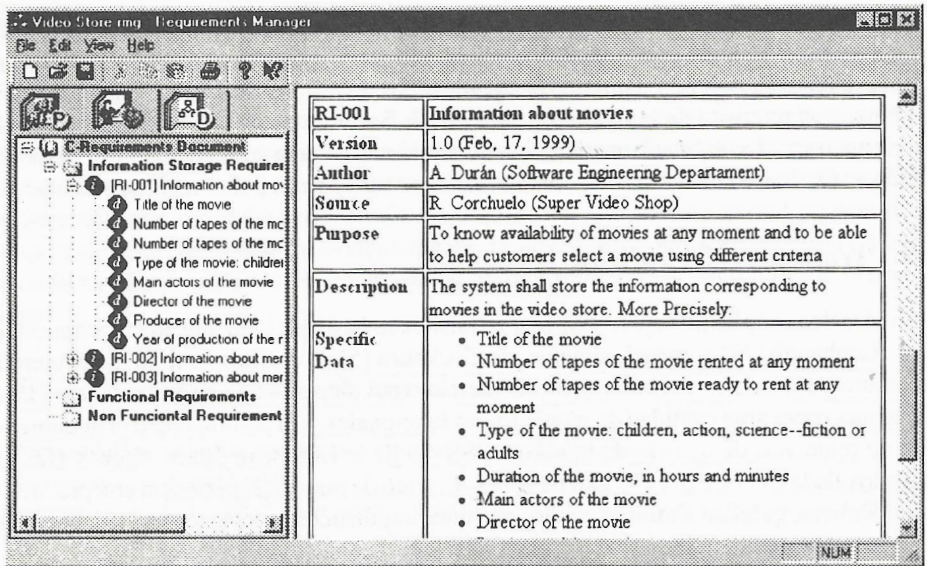


Figura10. Prototipo de herramienta CASE de gestión de requisitos

- [3] CERN. PSS-05 User Requirements Document Template. Technical report, CERN, 1998.
- [4] M. G. Christel y K. C. Kang. Issues in Requirements Elicitation. Technical Report CMU/SEI-92-TR-12, Software Engineering Institute, Carnegie Mellon University, 1996.
- [5] A. Cockburn. Goals and Use Cases. *JOOP*, Sep-Oct 1997.
- [6] A. Durán, B. Bernárdez, M. Toro, y R. Corchuelo. Una Propuesta para el Catálogo de Requisitos de MÉTRICA V2.1. *Novática*, Aceptado para su publicación. 1999.
- [7] A. Durán, B. Bernárdez, M. Toro, R. Corchuelo, A. Ruiz, y J. Pérez. Expressing Customer Requirements Using Natural Language Requirements Templates and Patterns. En *IMACS/IEEE CSCC'99 Proceedings*, Atenas, 1999.
- [8] A. Durán, B. Bernárdez, M. Toro, y A. Ruiz. Una Propuesta Metodológica para la Elicitación de Requisitos de un Sistema Software. En *Actas de las III Jornadas de Trabajo Menhir*, Murcia, 1998. Universidad de Murcia.
- [9] I. Jacobson, M. Christerson, P. Jonsson, y G. Övergaard. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Addison-Wesley, 4ª edición, 1993.
- [10] IBM OOTC. *Developing Object-Oriented Software: An Experience-Based Approach*. Prentice-Hall, 1996.
- [11] S. Raghavan, G. Zelesnik, y G. Ford. Lecture Notes on Requirements Elicitation. Educational Materials CMU/SEI-94-EM-10, Software Engineering Institute, Carnegie Mellon University, 1994.
- [12] J. Robertson y S. Robertson. *Volere Requirements Specification Template Edition 6.0*. Technical report, Atlantic Systems Guild, 1998.
- [13] C. Rolland y C. B. Achour. Guiding the Construction of Textual Use Case Specifications. *Data & Knowledge Engineering Journal*, 25(1-2):125-160, Marzo 1998.
- [14] J. Rumbaugh. Getting Started: Using Use Cases to Capture Requirements. *JOOP*, Sept. 1994.