

**Universidad de las Ciencias Informáticas**  
**“Facultad 4”**



**Título: “Procedimiento para desarrollar la  
Ingeniería de Requisitos en el proyecto Sistema  
de Gestión Penitenciaria”**

Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas

**Autoras:** Aracelis Reina Betancourt Cruz

Yanet Vega Miniet

**Tutor:** Ing. Madelín Haro Pérez

Junio de 2007

## DECLARACIÓN DE AUTORÍA

Declaramos que somos las únicas autoras de este trabajo y autorizamos al proyecto SIGEP de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Aracelis Reina Betancourt Cruz

---

Madelín Haro Pérez

---

Yanet Vega Miniet

## **DATOS DE CONTACTO**

Tutora: Ing. Madelín Haro Pérez

- Graduada en 2002 de la carrera Ingeniería Informática en el ISPJAE.
- Ha trabajado en la Empresa de Servicios Informáticos (ESI) de Cienfuegos y a partir de enero de 2003 en la UCI.
- Ha sido tutora de tesis durante los 5 cursos de la UCI y cotutora o consultora. Los temas trabajados han estado relacionados con el ciclo de vida del software (análisis, base de datos, arquitectura, prototipos, requisitos...), Gestión de Proyecto, Seguridad de sistemas informáticos y Linux.
- Ha participado en eventos como el Forum de Ciencia y Técnica, UCIENCIA y Universidad 2008 a nivel de base.
- Opta actualmente por la categoría docente de Asistente y cursa la maestría de Gestión de Proyecto.
- Se ha vinculado a los proyectos productivos en el rol de Jefe de Proyecto, Analista y Jefe de Capacitación.
- Ha impartido cursos de postgrado a funcionarios de empresas, profesores y trabajadores de la UCI y a los Directores de los IPI.
- Ha impartido cursos de capacitación en dos proyectos de producción en el extranjero.
- Ha recibido un curso de postgrado en el instituto TATA, en la India, recibiendo el certificado internacional de habilidades.
- Es asesora del Departamento Docente Central de Práctica Profesional.

## **AGRADECIMIENTOS**

A todos, gracias.

Especialmente a Heysell, Javier y Adolfo, quienes ayudaron enormemente a elaborar este trabajo,  
a Made, por su apoyo durante 5 años en la UCI,  
a Michel, por su ayuda y paciencia.

Las Autoras

## **RESUMEN**

Desarrollar una Especificación de Requisitos de Software en el ambiente de cambios existente en la Dirección General de Servicios Penitenciarios de Venezuela pone en riesgo el éxito del proyecto Sistema de Gestión Penitenciaria. Para minimizarlo fue necesario definir un procedimiento que permitiera obtener y mantener un acuerdo respecto a los requisitos del sistema. Dicho procedimiento se elaboró basándose en la propuesta de RUP para la Modelación de Negocio y los Requisitos, las buenas prácticas recomendadas por la bibliografía consultada y la experiencia de las autoras en la primera etapa de captura de requisitos. Se redefinieron flujos de trabajo, actividades, artefactos y roles para adecuarlos al contexto en el que se desarrolla el Sistema de Gestión Penitenciaria. Para validar el procedimiento propuesto se utilizaron técnicas del método de expertos.

## **PALABRAS CLAVE**

Ingeniería de requisitos, requisitos de software, seguimiento a los requisitos de software, artefactos, actividades, roles, flujos de trabajo, modelación de negocio.

# TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
RESUMEN	II
INTRODUCCIÓN	1
CAPÍTULO I FUNDAMENTACIÓN TEÓRICA PARA LA DISCIPLINA INGENIERÍA DE REQUISITOS	4
Introducción	4
1.1    Procesos de desarrollo de software. Estado del arte	4
1.2    Requisitos de software	5
1.2.1    Contexto para el desarrollo de la ERS	7
1.2.2    Características que debe poseer una ERS	8
1.2.3    Prácticas recomendadas para desarrollar la ERS	10
1.3    El Proceso Unificado de Desarrollo	12
1.3.1    Modelación de negocio	14
1.3.2    Ingeniería de requisitos en RUP	15
Conclusiones parciales	17
CAPÍTULO II PROCEDIMIENTO PARA DESARROLLAR LA INGENIERÍA DE REQUISITOS EN EL PROYECTO SIGEP	18
Introducción	18
2.1    Objetivos de la Ingeniería de Requisitos en SIGEP	18
2.2    Consideraciones para la elaboración del procedimiento	20
2.3    Modelación de negocio	21
2.3.1    Evaluación de la Organización	24
2.3.2    Comprensión del Negocio	26

2.4	Requisitos	29
2.4.1	Análisis del Problema	31
2.4.2	Definición del Sistema	33
2.4.3	Gestión de Cambios a los Requisitos	36
2.4.4	Seguimiento a los requisitos	38
2.5	Descripción de los artefactos	48
2.5.1	Artefactos elaborados por el Equipo de Desarrollo	48
2.5.2	Artefactos entregados al Equipo de Desarrollo	59
2.6	Roles	62
	Conclusiones parciales	64
CAPÍTULO III VALIDACIÓN DEL PROCEDIMIENTO		66
	Introducción	66
3.1	Selección del grupo de expertos	66
3.2	Validación de la propuesta	68
	Conclusiones parciales	71
CONCLUSIONES		72
RECOMENDACIONES		73
BIBLIOGRAFÍA		74
ANEXOS		77
	Anexo 1. Aspectos a considerar para la Evaluación de la Organización. (MEJÍAS '06)	77
	Anexo 2. Encuesta para determinar el coeficiente de competencia de los expertos	79
	Anexo 3. Evaluación del Procedimiento para desarrollar la Ingeniería de Requisitos en el proyecto Sistema de Gestión Penitenciaria (SIGEP)	80
	Anexo 4. Caracterización de los expertos	83

## ÍNDICE DE FIGURAS

Figura 1. Contexto de elaboración de la ERS (IEEE '98a). .....	7
Figura 2 Fases y flujos de trabajo en RUP (RATIONAL SOFTWARE '03b).....	13
Figura 3. Objetivos de la Ingeniería de Requisitos en SIGEP. ....	20
Figura 4. Modelación de Negocio. ....	23
Figura 5. Evaluación de la organización. ....	25
Figura 6. Comprensión del negocio. ....	28
Figura 7. Requisitos.....	30
Figura 8. Análisis del problema.....	32
Figura 9. Definición del Sistema. ....	35
Figura 10. Gestión de Cambios a los Requisitos.....	37
Figura 11. Traza de los requisitos a través de los artefactos que se deben elaborar en SIGEP.....	39
Figura 12. Elementos de seguimiento a los requisitos. ....	42

## ÍNDICE DE TABLAS

Tabla 1. Artefactos que se deben elaborar en SIGEP.....	38
Tabla 2. Artefactos propuestos para desarrollar la Modelación de Negocio y la Ingeniería de Requisitos en SIGEP.....	48

## INTRODUCCIÓN

El ascenso de Hugo Chávez al poder en la República Bolivariana de Venezuela significó el inicio de una revolución social, económica, política y cultural. Por primera vez millones de venezolanos pobres tuvieron participación en las decisiones políticas y económicas del país, acceso a la educación, la cultura, la salud y el deporte. Algunos de los programas con los que se ha beneficiado la población son la Misión Barrio Adentro, la Misión Ribas, que ha permitido graduar a más de 77 000 bachilleres; la Misión Robinson, a través de la cual se han alfabetizado alrededor de 1 300 000 venezolanos; la Misión Sucre, que otorgó en el 2003 cerca de 30 000 becas universitarias; la Misión Mercal, para la lucha contra el hambre y la Misión Vuelvan Caras, para enfrentar el desempleo y la pobreza.

Otro frente de transformación social ha sido la garantía de la seguridad ciudadana y el respeto a la Constitución, en este sentido el gobierno venezolano formuló el proyecto Modernización del Sistema Penitenciario para revertir la crítica situación existente en los penales. Como parte de este proyecto se contrató a la Universidad de las Ciencias Informáticas (UCI) el desarrollo del Sistema de Gestión Penitenciaria (SIGEP) para apoyar la actividad de los Establecimientos, Unidades Técnicas de Apoyo y Centros de Tratamiento Comunitario y resolver las necesidades de gestión, información, comunicación y apoyo a la toma de decisiones de la Dirección General de Servicios Penitenciarios (DGSP).

La DGSP se encuentra inmersa en un profundo proceso de transformación organizacional; ello implica que se establezcan un conjunto de definiciones para el funcionamiento eficiente de algunas áreas claves dentro de la institución, surjan o se modifiquen áreas de trabajo, cargos y procesos. El desarrollo de un producto de software en las condiciones DGSP, donde los procesos no están bien definidos y existen continuos cambios dificulta la identificación y estabilización de los requisitos del sistema, y por tanto, pone en riesgo el éxito del proyecto. (ARIAS y OTROS '06).

Para mitigar este riesgo, el desarrollo del software fue concebido por etapas, para incluir en cada una de ellas los procesos que tuvieran mayor nivel de definición, dejando para las etapas posteriores los menos definidos de modo que se pueda avanzar en su concepción previamente a la automatización (ARIAS y OTROS '06). Es necesario también, definir un procedimiento para capturar los requisitos del software, basándose en la disciplina de Requisitos propuesta por el Proceso Unificado de Desarrollo (RUP, por sus siglas en inglés), proceso definido para el desarrollo del proyecto; que permita, en el ambiente de

cambios existente en la DGSP, mantener un acuerdo entre los clientes, usuarios finales y el equipo de desarrollo en cuanto a lo que debe hacer el sistema. Por ello se ha definido como problema científico:

¿Cómo obtener una Especificación de Requisitos de Software (ERS) en el ambiente de cambios existente en la DGSP, que guíe al equipo de desarrollo en la construcción del SIGEP, de manera que este apoye los procesos de la organización y satisfaga las necesidades y expectativas de los interesados?

Partiendo del problema anteriormente planteado se ha determinado como objeto de estudio el proceso de desarrollo de software en el proyecto SIGEP, este proceso implica la aplicación de diferentes disciplinas técnicas. Para dar solución al problema enunciado es necesario profundizar en el estudio de la Ingeniería de Requisitos como disciplina en el proyecto SIGEP, la cual ha sido identificada como el campo de acción del presente trabajo. Se ha definido como objetivo:

Elaborar un procedimiento para obtener y mantener una ERS en el ambiente de cambios existente en la DGSP que guíe al equipo de desarrollo en la construcción del SIGEP.

Las siguientes preguntas orientan la investigación para el logro del objetivo propuesto:

- ¿Qué elementos ponen en riesgo la obtención y mantenimiento de un conjunto de requisitos en SIGEP?
- ¿Qué proponen RUP, la bibliografía disponible sobre el tema y otras metodologías para desarrollar la Ingeniería de Requisitos? ¿Qué elementos deben mantenerse en el proyecto SIGEP y cuáles deben ser enriquecidos, modificados o eliminados para mitigar los riesgos identificados?

Para responder las interrogantes anteriormente planteadas se definen las siguientes tareas:

- Identificar las características que debe tener la ERS para que guíe al equipo de desarrollo en la construcción de un sistema que apoye los procesos de negocio y satisfaga a los interesados.
- Identificar los factores que influyen negativamente en el desarrollo de la ERS en SIGEP.
- Identificar buenas prácticas, estrategias y métodos que contribuyan a minimizar los problemas durante la Ingeniería de Requisitos en SIGEP.
- Partiendo de que se ha definido RUP como proceso de referencia para el proyecto (ARIAS y OTROS '06), valorar cualitativamente la aplicabilidad de cada uno de los elementos identificados como resultado

de la tarea anterior y de los propuestos por la disciplina Requisitos tomando en cuenta las particularidades del proyecto SIGEP.

- Elaborar y modelar un procedimiento para desarrollar la Ingeniería de Requisitos en el proyecto SIGEP que defina: objetivos, actividades, artefactos y roles.
- Validar el procedimiento propuesto mediante la aplicación de técnicas del método de expertos.

# **CAPÍTULO I FUNDAMENTACIÓN TEÓRICA PARA LA DISCIPLINA INGENIERÍA DE REQUISITOS**

## **Introducción**

En el capítulo se presenta un estudio sobre el estado actual de los procesos de desarrollo de software. Se definen los principales conceptos relacionados con la Ingeniería de Requisitos, los elementos que deben considerarse mientras se desarrolla la ERS, las características que debe poseer y buenas prácticas para elaborarla. Finalmente se describe RUP como proceso de referencia para el desarrollo de la Modelación de Negocio y la Ingeniería de Requisitos en SIGEP.

### **1.1 Procesos de desarrollo de software. Estado del arte**

El software está presente en la mayoría de las actividades que desarrollan las sociedades actuales; transacciones financieras, control de tráfico aéreo e intervenciones quirúrgicas son marcos habituales de aplicación de los sistemas de software. En la medida en que el software se ha insertado en mayor número de ramas del quehacer humano y su papel se ha vuelto más crítico la industria se ha visto obligada a desarrollar métodos, procesos y herramientas para garantizar productos de calidad, cumpliendo los cronogramas y dentro de los costos. Sin embargo, el número de proyectos entregados en tiempo, dentro del presupuesto y que satisfacen las expectativas de los clientes y usuarios finales es muy bajo, varios estudios lo estiman inferior al 20%. Otras investigaciones (LEISHMAN y COOK '02), (CASTILLO '06; YOUNG '02) señalaron a los problemas en los requisitos como la principal causa del gran número de proyectos fracasados, con errores o que exceden el costo y los cronogramas. A esto se suman otros, como planificaciones, procesos, metodologías y herramientas de gestión inadecuadas para la empresa y/o el proyecto según (CASTILLO '06; LEISHMAN y COOK '02).

En general la Industria de Software y en particular la Industria Latinoamericana de Software, tienen como objetivo fortalecer la competitividad de sus empresas mediante el mejoramiento de los procesos de desarrollo de software. En el caso latinoamericano se evidencia además un esfuerzo por lograr insertarse en el mercado internacional de productos y servicios informáticos así como alcanzar niveles internacionales de calidad. Sin embargo, en la región predominan la pequeña y mediana empresa y los modelos y estándares de calidad y procesos existentes abordan el problema fundamentalmente para empresas de gran tamaño, muchas de las cuales tienen procesos definidos. (VILLARROEL y otros. '06),

(MORDEZKI y *otros*. '06) Debido a lo anterior en Latinoamérica el esfuerzo está dirigido a la elaboración de modelos de procesos y evaluación apropiados para la región.

En el caso cubano, la visión de la industria de software y especialmente de la UCI, es que se convierta en una de las ramas más productivas y de mayor aporte de recursos al país. (RUIZ '07) Actualmente el 90% de los estudiantes están organizados en más de 200 proyectos productivos dirigidos tanto a la informatización de sectores claves del país como a la exportación, fundamentalmente en proyectos en el marco del ALBA. (RUIZ '07), (GIL '07)

En la UCI, es una necesidad la adecuación de los modelos de producción y los modelos de calidad internacionales a las características de la institución. (RUIZ '07) Actualmente los esfuerzos realizados en la adecuación de los procesos se han visto limitados a los proyectos productivos, que tomando RUP como proceso de desarrollo, lo han adaptado a sus características, pero estos esfuerzos no han sido documentados y socializados en la comunidad universitaria. Por ello en SIGEP se debe adecuar el proceso propuesto por RUP.

## **1.2 Requisitos de software**

La IEEE 1233-1998 define requisito como:

- “i) Una condición o capacidad que necesita el usuario para resolver algún problema o alcanzar un objetivo.
- ii) Condición o capacidad que debe cumplir o poseer un sistema o componente del sistema para satisfacer un contrato, estándar, especificación u otro documento formalmente impuesto.
- iii) Una representación documental de una condición o capacidad como en i o en ii.” (IEEE '98a)

“La Ingeniería de Requisitos es el uso sistemático de procedimientos, técnicas, lenguajes y herramientas para obtener con un coste reducido el análisis, documentación, evolución continua de las necesidades del usuario y la especificación del comportamiento externo que satisfaga las necesidades del usuario” (PRESSMAN '05) El resultado es una “colección estructurada de información, que contiene los requisitos del sistema” (IEEE '98a), esto es, una Especificación de Requisitos de Software (ERS).

La ERS tradicionalmente ha sido vista como un documento que comunica los requisitos tanto a los clientes y usuarios como al equipo de desarrollo, en muchos casos sin comprender que esta actúa como un puente entre ambos grupos y que por tanto, debe estar representada en lenguajes diferentes y con diferentes niveles de formalidad.

Existen varias formas de representar la ERS que incluyen desde elaborar solamente una lista de características y funciones hasta interpretar el Modelo de Casos de Uso como la ERS. Al respecto (LARMAN '02) indica:

“El enfoque tradicional de la lista de características y funciones para capturar los requisitos, puede contribuir a que los proyectos no proporcionen lo que las personas realmente quieren, puesto que no fomenta que el personal involucrado considere los requisitos en un contexto amplio de uso del sistema, en un escenario para alcanzar algún resultado observable de valor, o algún objetivo. Por el contrario, los casos de uso sitúan las características y funciones en un contexto orientado al objetivo.” En el libro *Writing Effective Use Case*, (COCKBURN '00) plantea que los casos de uso son realmente requisitos, que no se deben convertir en otro tipo de requisito funcional y que escritos correctamente detallan lo que debe hacer el sistema. Pero se indica además:

“Ellos no son todos los requisitos - los casos de uso-. (...) Ellos constituyen sólo una parte (tal vez la tercera) de todos los requisitos que usted debe capturar – una parte muy importante, pero sólo una parte.” (COCKBURN '00)

A lo señalado por *Cockburn* pudiera añadirse que a pesar de que los casos de uso están descritos en lenguaje natural no son fácilmente comprensibles por los clientes y usuarios finales y una vez identificados, es difícil avanzar en su descripción si los interesados no logran imaginarse el sistema con mayor nivel de detalle. Según la experiencia de las autoras el uso de prototipos es útil para identificar, detallar y validar los requisitos funcionales con los clientes y usuarios finales. Sin embargo, un prototipo, no tiene toda la información que necesita la comunidad técnica para construir el sistema, tales como las pre y poscondiciones que debe cumplir cada requisito funcional, que en este caso son de suma importancia, las reglas del negocio que deben ser satisfechas o respetadas por el sistema y las entidades conceptuales involucradas en cada funcionalidad.

A consideración de las autoras lo ideal es emplear una combinación de enfoques que permita que los requisitos sean comprensibles para los clientes y usuarios finales y que estén lo suficientemente claros para el equipo de desarrollo de manera que lo guíen en la construcción del sistema correcto.

### 1.2.1 Contexto para el desarrollo de la ERS

Como se ha señalado la ERS está dirigida a dos audiencias, que a su vez, brindan una retroalimentación al proceso de elaboración de esta. Pero hay un tercer factor que influye en el proceso: el ambiente; el cual puede imponer restricciones de mercado, culturales, físicas, técnicas, entre otras. (IEEE '98a)

La imagen a continuación muestra cómo interactúan los interesados, la comunidad técnica y el ambiente con el proceso de elaboración de la ERS. Esta representación fue tomada de la IEEE, la cual se modificó según las siguientes consideraciones: el término cliente fue sustituido por interesados. Los términos cliente y usuario final no son sinónimos y ambos forman parte de la misma audiencia, un término empleado para englobarlos a ambos es interesado. Se añadieron las reglas de negocio como un elemento importante que aporta el ambiente.

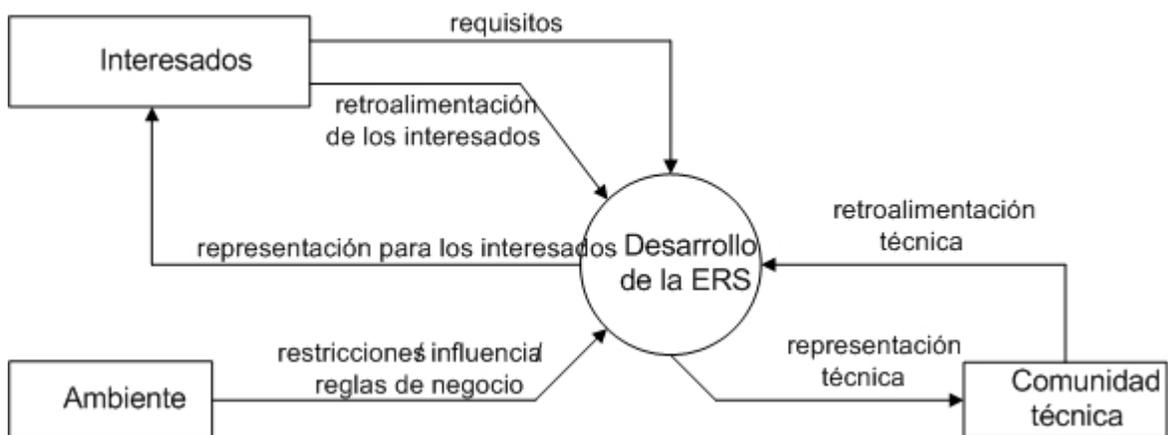


Figura 1. Contexto de elaboración de la ERS (IEEE '98a).

#### Interesados

Los interesados son el elemento más importante en el contexto de la ERS. La meta del proceso de software es desarrollar un sistema que satisfaga sus necesidades y expectativas. (IEEE '98a; JACOBSON y otros. '04).

Los analistas, quienes forman parte de la Comunidad Técnica, obtienen de los interesados los requisitos del sistema y con ellos construyen de manera iterativa la ERS. La ERS es validada por los interesados a través de una representación de esta en un lenguaje que sea fácilmente comprensible por ellos. Los elementos resultantes de estas validaciones constituyen una retroalimentación para el desarrollo de la ERS.

## **Ambiente**

El ambiente en el cual se desarrolla la ERS impone restricciones y reglas a los requisitos e influye sobre ellos. Según el Estándar 1223-1998 (IEEE '98a) las influencias del ambiente pueden clasificarse en los siguientes grupos que se solapan: políticas, de mercado, estándares y regulaciones técnicas, culturales, organizacionales y físicas.

Estas restricciones, reglas de negocio e influencias deben ser registradas en la ERS.

## **Comunidad Técnica**

La Comunidad Técnica es la encargada de construir el software que necesitan los interesados, por tanto, deberá desarrollar el diseño, implementación, prueba, integración y mantenimiento del sistema. Mientras se desarrollan estas actividades la Comunidad Técnica propondrá cambios, eliminaciones en la ERS y nuevos requisitos, retroalimentando así el desarrollo de esta. Por ello, es recomendable involucrar tempranamente a la Comunidad Técnica en la elaboración de la ERS.

### **1.2.2 Características que debe poseer una ERS**

Una ERS debe ser, según la norma IEEE 830 – 1998 (IEEE '98b):

- **Correcta.** La ERS es correcta sí, y sólo sí, contiene todos los requisitos que el software debe satisfacer. No hay una herramienta o procedimiento que garantice esta característica, sin embargo, limitarse a obtener de una vez lo que “quiere” el cliente pone en riesgo el logro de esta característica. Realizar un proceso iterativo, donde se involucren los interesados, seguir la traza de los requisitos y validarlos con los interesados contribuyen a lograr una ERS correcta. (IEEE '98a; LEISHMAN y COOK '02)
- **No ambigua.** La ERS es no ambigua sí, y sólo sí, cada requisito tiene una única interpretación. Para elaborar una ERS no ambigua es necesario mantener y documentar un acuerdo entre el equipo de desarrollo y los interesados respecto a los requisitos. Cuando se usa el lenguaje natural para documentar los requisitos pueden introducirse ambigüedades, sin embargo, este es fácilmente comprendido por los interesados, a diferencia de los lenguajes formales, que requieren conocimientos específicos pero permiten reducir la ambigüedad. Anteriormente se sugirió usar una combinación de representaciones a lo cual habría que añadir que si en dicha combinación se hace uso del lenguaje natural debe mantenerse un glosario de términos para disminuir la ambigüedad. (IEEE '98a)
- **Completa.** La ERS es completa sí, y sólo sí, incluye los siguientes elementos:

- Todos los requisitos funcionales y no funcionales son conocidos y documentados en la ERS.
- Están definidas todas las responsabilidades del sistema respecto a los datos de entrada, tanto válidos como no válidos y respecto a los datos de salida.
- En los documentos de la ERS todas las figuras, tablas, diagramas y definiciones de términos están nombrados y referenciados.

Desarrollar de forma iterativa la ERS de software, validarla y usar múltiples representaciones contribuyen a lograr la completitud. (IEEE '98a; LEISHMAN y COOK '02; PRESSMAN '05)

- Consistente. La ERS es consistente sí, y sólo sí, ningún subconjunto de la misma entra en contradicción con otro subconjunto (IEEE '98a). Las revisiones técnicas y el uso de diferentes representaciones para los requisitos contribuyen a reconocer la inconsistencia.
- Clasificada por importancia. La ERS cumple con esta característica si cada requisito puede ser identificado de manera única y tiene un atributo que indica su importancia desde el punto de vista de los interesados. Todos los requisitos no son igualmente importantes, algunos son críticos para el sistema y otros son deseables, conocer estos atributos es útil para planificar las iteraciones. (IEEE '98a)
- Comprobable. La ERS es comprobable sí, y sólo sí, se puede comprobar por una persona o máquina mediante un procedimiento finito y cuya relación costo-beneficio sea aceptable que cada requisito está en el sistema desarrollado (IEEE '98a).
- Modificable. La ERS es modificable sí, y sólo sí, puede ser modificada fácilmente manteniendo la completitud y consistencia. Mantener la traza de los requisitos, organizarlos adecuadamente y usar referencias cruzadas contribuye a hacerla modificable. (IEEE '98a)
- Posible de rastrear. La ERS es posible de rastrear si está claro el origen de cada requisito y es posible determinar los elementos relacionados en etapas posteriores del desarrollo. (IEEE '98a)

Los requisitos cambian inevitablemente durante el ciclo de vida del software, para evaluar el impacto de los cambios y gestionarlos es necesario seguir la traza de los requisitos hacia otros artefactos.

Mantener la traza de los requisitos es la habilidad de seguir un elemento del proyecto hacia otro elemento relacionado. (*Concept: Traceability* '03) En otras palabras, es la capacidad de seguir la evolución de un elemento durante el desarrollo de un proyecto de software. Tiene como objetivos:

- “Entender la fuente de los requisitos.
- Gestionar el alcance del proyecto.
- Gestionar los cambios en los requisitos.
- Evaluar el impacto de los cambios en los requisitos.
- Evaluar el impacto de los fallos detectados durante las pruebas en los requisitos.
- Verificar que todos los requisitos del sistema fueron cumplidos.” (*Concept: Traceability '03*)

En el documento *Traceability Strategies for Managing Requirements with Use Cases* (SPENCE y PROBASCOS '98) se definen dos tipos de traza: la implícita y la explícita.

En todo proceso de desarrollo donde se utilice *Unified Model Language* (UML) existe cierta traza implícita, debido fundamentalmente a las relaciones entre los artefactos que establece este lenguaje. (SPENCE y PROBASCOS '98) Sin embargo, cuando se utilizan múltiples representaciones de los requisitos es necesario establecer explícitamente la traza de los requisitos.

La traza explícita tiene el inconveniente de que requiere un esfuerzo adicional, pero en proyectos de gran tamaño, donde se auguran numerosos cambios a los procesos de negocio y por tanto a los requisitos, las autoras consideran que es necesario emplearla. Ahora bien, cada proyecto debe definir en concordancia con sus necesidades los elementos a través de los cuales seguirá la traza de los requisitos. Se debe lograr que no falten elementos, impidiendo de esta manera hacer el seguimiento, y que no existan elementos superfluos que lo dificulten.

### **1.2.3 Prácticas recomendadas para desarrollar la ERS**

De manera general la bibliografía consultada de los siguientes autores: (JACOBSON y otros. '04), (PRESSMAN '05), (FERNÁNDEZ '02), (IEEE '98a), (IEEE '98b; PERALTA '03), (GONZÁLEZ y FERNÁNDEZ '06), (RATIONAL SOFTWARE '03a; YOUNG '02) (COCKBURN '00) sugiere que para obtener una ERS que satisfaga las características anteriores es recomendable:

- Comprender el contexto del sistema. El software debe añadir valor al negocio que está automatizando, para lograrlo es necesario que el equipo de desarrollo comprenda el negocio y que los requisitos del sistema se correspondan con los procesos, metas y reglas del negocio.

- Comprender el problema. Antes de buscar una solución es necesario llegar a un acuerdo en el problema que será resuelto, identificar los interesados en la solución, definir los límites e identificar las restricciones impuestas al sistema.
- Evaluación y síntesis de la solución. Una vez obtenido un acuerdo en cuanto al problema a resolver los analistas e interesados evalúan y proponen una solución, centrándose en el qué y no en el cómo.
- Verificación y validación. El artículo *Recommended Requirements Gathering Practices* (YOUNG '02) expresa que se ha estimado que el 85% de los defectos en el desarrollo de software se originan en los requisitos. Además, el costo de encontrar y corregir los errores se incrementa en la medida en que el proyecto se adentra en nuevas fases del desarrollo. Por ello es importante verificar y validar los requisitos tempranamente. Esto puede hacerse mediante revisiones formales, sesiones de validación con los clientes, prototipos y entregas a corto plazo de pequeñas partes funcionales del software como plantean las metodologías ágiles.
- Documentar los requisitos. ¿Qué y cuánta documentación generar durante la Ingeniería de Requisitos? La respuesta a la interrogante anterior recibirá variadas respuestas de acuerdo a la metodología de desarrollo que haya adoptado cada proyecto u organización. Sin embargo, la mayoría coinciden en que es necesario documentar los requisitos del software: RUP propone hacerlo mediante la Especificación de Requisitos de Software, las Especificaciones Suplementarias y el Modelo de Casos de Uso; XP sugiere el empleo de Historias de Usuario; Crystal indica utilizar un Archivo de requisitos y Desarrollo Manejado Por Rasgos (*Feature Driven Development-FDD*) plantea documentar los requisitos como Rasgos, que son similares a las Historias de Usuario. El Modelado Ágil, es un complemento a las metodologías ágiles para facilitar el modelado y documentación efectiva de sistemas de software, uno de sus principios es sólo desarrollar los artefactos que sean necesarios para el proyecto, con lo cual coinciden otros autores que indican que cada proyecto adecue los procesos de desarrollo y artefactos a sus necesidades. (HURTADO y BASTARRICA '05; JACOBS '04; ORANTES '07)
- Priorizar los requisitos. Es necesario definir qué requisitos serán incluidos en cada iteración del desarrollo, para ello se deben priorizar los requisitos a partir de criterios que cada equipo de proyecto tendrá que definir, la importancia para los interesados, como se señaló anteriormente, no debe omitirse.
- Exigir que los interesados se involucren. Para que los interesados queden satisfechos con el producto final es necesario que sean parte activa del equipo, fundamentalmente durante la modelación del negocio

y la captura de requisitos; las metodologías ágiles sugieren que haya un cliente o representante de este junto al equipo de desarrollo durante todo el ciclo de vida del proyecto.(HURTADO y BASTARRICA '05) (ORANTES '07)

- Obtener los requisitos de forma iterativa. Cuando se inicia la captura de requisitos muchos aspectos del sistema en construcción no están claros, según avanza la definición de la solución el conocimiento sobre los requisitos se enriquece, aparecen nuevos requisitos y se clarifican detalles sobre los ya existentes. Obtenerlos en iteraciones sucesivas permite registrar esta nueva información. (ADOLPH y BRAMBLE '01) (COCKBURN '00)
- Controlar los cambios a los requisitos. El mundo está en constante evolución, los negocios, la economía, la política y la sociedad están cambiando continuamente y suponer que los requisitos no van a cambiar durante el desarrollo de software no es realista. Lo importante es estar preparados para asumir los cambios mediante un proceso de gestión de cambios que garantice la completitud y consistencia de la ERS.
- Mantener la traza de cada requisito. Para evitar el riesgo de implementar “requisitos inútiles” debe conocerse el origen de cada requisito y con el objetivo de evaluar el impacto de los cambios cada requisito debe seguirse hacia los elementos relacionados.
- Mantener un glosario de términos. Debe mantenerse entre el equipo de desarrollo de software y los interesados en el proyecto un entendimiento común sobre los requisitos, cuando se utiliza el lenguaje natural para documentar los requisitos pueden introducirse ambigüedades; mantener un glosario de términos contribuye a mejorar el entendimiento.

### **1.3 El Proceso Unificado de Desarrollo**

Un proceso de desarrollo de software es el:

“Conjunto de actividades para desarrollar y mantener el software y los productos asociados (documentos de diseño, casos de prueba, manuales de usuario...) y gestionar su producción” (MONTILVA '06)

En el Proyecto Técnico Económico de SIGEP se define como proceso de referencia RUP, el cual define un “marco de trabajo genérico que puede especializarse para gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y

diferentes tamaños de proyecto.” RUP es un proceso que se caracteriza por ser según (JACOBSON y otros. '04):

- Dirigido por casos de uso, los casos de uso no sólo son una herramienta para especificar los requisitos; ellos guían el diseño, la implementación, pruebas y son utilizados para la estimación de tiempo y esfuerzo de desarrollo.
- Centrado en la arquitectura. Los arquitectos moldean el sistema desde varios puntos de vista –vistas arquitectónicas- para darle una forma que permita que el sistema evolucione.
- Iterativo e incremental. Es útil dividir el proyecto en iteraciones que resultan en un incremento del producto.

RUP define los flujos de trabajo Modelación del Negocio, Requisitos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Gestión de Proyecto, Gestión de Configuración y Ambiente. La figura a continuación muestra el esfuerzo dedicado a cada uno de estos subprocesos en cada etapa del ciclo de vida.

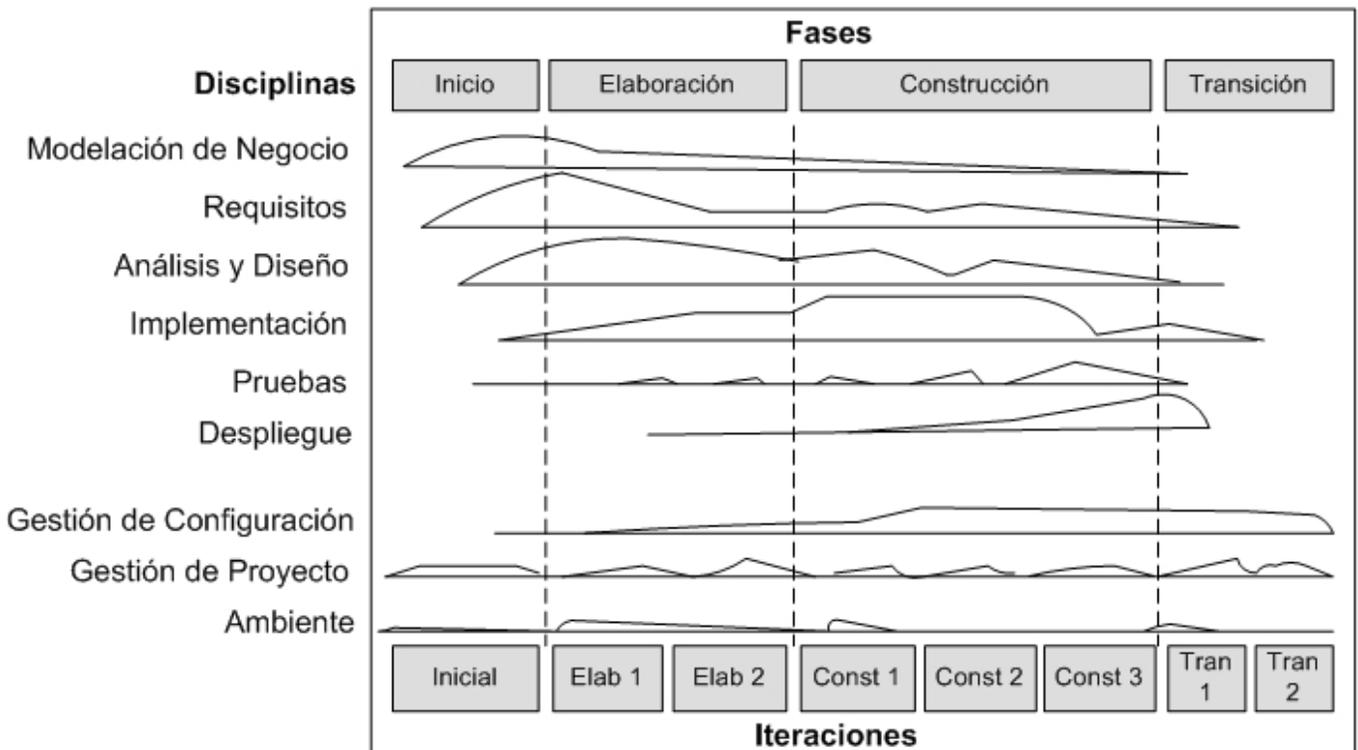


Figura 2 Fases y flujos de trabajo en RUP (RATIONAL SOFTWARE '03b).

Los flujos de trabajo mostrados anteriormente proponen la realización de un gran número de actividades y la elaboración de un amplio conjunto de artefactos que usualmente los proyectos de desarrollo de software se comprometen a desarrollar pero que no realizan en su totalidad debido a falta de tiempo o a que descubren que no eran necesarios. Sin embargo, RUP indica que al inicio del proyecto se realice una adecuación de cada flujo de trabajo. En este punto es útil recurrir a un principio del Modelado Ágil: producir sólo los artefactos y realizar las actividades que tienen un propósito dentro del proyecto.

### **1.3.1 Modelación de negocio**

Un software debe añadir valor al negocio que automatiza. Al respecto en el libro Ingeniería de software: Un enfoque práctico, se propone desarrollar un modelo de negocio para derivar los requisitos del sistema antes de iniciar la Ingeniería de Requisitos. (PRESSMAN '05)

RUP propone seis escenarios para la Modelación del Negocio. En dependencia del escenario en que se desarrolla, cada proyecto de software debe ajustar las actividades que se realizan. Cuando se modela el negocio con el propósito de desarrollar un sistema sólo para representar información se puede desarrollar un Modelo de Dominio, sin considerar los flujos de actividades en los procesos de negocio. En el resto de los casos es necesario desarrollar un Modelo de Negocio, y cada proyecto, luego de evaluar la organización debe definir con qué profundidad desarrollar este modelo.

Un Modelo de Negocio es una abstracción de cómo funciona el negocio. (ERIKSSON y PENKER '00). El Modelo de Negocio, a diferencia del Modelo de Dominio describe los procesos de negocio.

Un proceso de negocio es un grupo coherente de actividades que son necesarias para entregar un producto o servicio con un valor tangible para el usuario. (TOLIS y NILSSON).

RUP representa los procesos de negocio como casos de uso de negocio, centrados en los actores y en el valor que estos esperan obtener del proceso, y realizaciones de casos de uso de negocio, que detallan cómo ocurre el proceso de negocio. En la actualidad la mayoría de las organizaciones han adoptado sistemas de gestión basados en procesos de manera tal que le permitan garantizar el logro de sus objetivos. Modelar los procesos de negocio, según el criterio de las autoras, representándolos como comúnmente se hace en las organizaciones contribuye a mejorar la comunicación entre el equipo de desarrollo y los interesados. Al emplear casos de uso y realizaciones de casos de uso del negocio para representar los procesos los interesados se verán obligados a comprender una nueva forma de representación, lo cual requerirá emplear mayor tiempo y esfuerzo. Además, un caso uso de negocio

describe la interacción entre el actor y el negocio como si este fuera una caja negra, la descripción de cómo ocurre el proceso internamente se encuentra en la realización del caso de uso de negocio; de esta manera la descripción de un proceso de negocio está dividida y dificulta el entendimiento del mismo por parte de los interesados.

El empleo de lenguajes de modelado visual permite representar con mayor claridad la estructura, dinámica y funcionalidad del proceso y mejora la comunicación entre el equipo de desarrollo y los interesados. El lenguaje más utilizado en la modelación de procesos dentro de proyectos informáticos es UML. UML *Business* –es una extensión de UML- y BPMN –notación propuesta por el consorcio *Business Process Management Initiative*- son otros dos lenguajes gráficos para la representación de procesos. Estos lenguajes permiten modelar los objetivos, actividades, actores, recursos, productos, herramientas y reglas de negocio.

Las autoras consideran que el uso de UML para modelar la vista estructural de la organización, UML *Business* para representar los procesos, sus objetivos, recursos, productos, herramientas y reglas y BPMN para modelar el flujo de actividades de los procesos facilita la representación de todos los aspectos del negocio mencionados anteriormente. Estos lenguajes serán empleados también en la modelación del procedimiento propuesto en este trabajo.

### **1.3.2 Ingeniería de requisitos en RUP**

El flujo de trabajo Requisitos, el cual define el proceso para desarrollar la Ingeniería de Requisitos en RUP, propone las siguientes actividades:

1. Análisis del problema.

El propósito del análisis del problema es obtener un acuerdo en cuanto al problema que será resuelto, la información obtenida sobre el problema se captura en el documento Visión. Los términos del proyecto se definen en el Glosario de Términos. Para determinar las fronteras del sistema se inicia la identificación de los actores con los que interactúa.

2. Comprensión de las necesidades de los interesados.

El propósito es entender las necesidades de los interesados en el proyecto derivadas de talleres, entrevistas, reuniones y de las reglas del negocio. Esta información resulta en un refinamiento del

artefacto Visión. Además, durante la realización de este flujo de trabajo se puede comenzar un debate de los requisitos funcionales del sistema en términos de sus casos de uso y actores.

### 3. Definición del sistema.

El objetivo es comenzar a definir el alcance de alto nivel del sistema esbozando los requisitos.

Las actividades que se enfocan en el análisis del problema y el entendimiento de las necesidades de los interesados crean iteraciones tempranas de definiciones del sistema, incluyendo los rasgos definidos en la Visión y un primer borrador de los requisitos detallados. Definiendo el sistema, el equipo debe enfocarse en identificar actores y casos de uso más completos y extiende los requisitos no-funcionales globales definidos en las Especificaciones Suplementarias.

### 4. Administración del alcance del sistema.

El objetivo es hacer el alcance del sistema que comienza a desarrollarse tan explícito como sea posible y enfocarse en un grupo manejable de requisitos para trabajar en la iteración. Gestionar el alcance del sistema es una actividad continua que divide el proyecto en partes más manejables. Basándose en los atributos de los requisitos se decide cuáles incluir en cada iteración.

### 5. Refinamiento de la definición del sistema.

El propósito del refinamiento de la definición del sistema es detallar los requisitos.

El resultado es un entendimiento completo de la funcionalidad del sistema expresado en los requisitos, detallado en los artefactos de la especificación y los prototipos esbozados. Este trabajo comienza típicamente por reexaminar la definición de actores existente y si es necesario describirlos brevemente, entonces se continúa detallando los casos de uso que han sido previamente esbozados para cada actor.

### 6. Gestión de los cambios a los requisitos.

El propósito de este detalle de flujo de trabajo es evaluar el impacto de los cambios necesarios en los requisitos y gestionar el impacto de los cambios aprobados.

Los cambios en los requisitos tienen impacto en otros artefactos, por ejemplo, los modelos producidos en el trabajo de análisis, diseño y pruebas. Para determinar el impacto de los cambios es necesario identificar las relaciones de los requisitos con el resto de los artefactos producidos durante el desarrollo.

Cada vez que los requisitos sean modificados deben actualizarse los atributos de los requisitos y sus relaciones.

El flujo de trabajo propuesto por RUP, cumple con la mayoría de las prácticas recomendadas para desarrollar una ERS, sin embargo, a consideración de las autoras deben añadirse actividades de validación con los interesados para detectar tempranamente inconformidades y necesidades de cambios a los requisitos. Este flujo de trabajo tampoco contempla la elaboración de prototipos como un método para la obtención, refinamiento y validación de los requisitos, lo cual resulta de gran utilidad ya que los interesados entienden mejor el prototipo que otro tipo de especificaciones. No obstante, se debe ser cuidadoso y no sustituir la ERS por prototipos, porque estos no representan todos los requisitos del sistema.

### **Conclusiones parciales**

Los problemas en los requisitos son una de las principales causas de que los proyectos de software no cumplan los cronogramas, sobrepasen los costos y de que se obtengan sistemas que no satisfacen las expectativas de los clientes y usuarios finales. En la UCI no existe un modelo de proceso para desarrollar la Ingeniería de Requisitos adecuado a la institución y a sus proyectos productivos que permita minimizar estos problemas.

Considerando los elementos señalados anteriormente, es necesario definir un procedimiento para desarrollar la Ingeniería de Requisitos en el proyecto SIGEP que se adecue específicamente a las características de este, usando como referencia RUP y basándose en las prácticas recomendadas según la bibliografía consultada sobre el tema.

# CAPÍTULO II PROCEDIMIENTO PARA DESARROLLAR LA INGENIERÍA DE REQUISITOS EN EL PROYECTO SIGEP

## Introducción

La obtención de una ERS adecuada es una condición necesaria, aunque no suficiente, para el éxito de los proyectos de software. En el capítulo se detalla el procedimiento propuesto para desarrollar la Ingeniería de Requisitos en el proyecto SIGEP mediante la descripción gráfica y textual de sus objetivos, subprocesos, artefactos y roles.

### 2.1 Objetivos de la Ingeniería de Requisitos en SIGEP

Según se definió en el Capítulo I la Ingeniería de Requisitos tiene como fin **elaborar y mantener la ERS**, la cual, a su vez, debe guiar al equipo de desarrollo en la construcción del sistema correcto. Construir el sistema correcto significa proveer todas las funcionalidades útiles para apoyar los procesos estratégicos de la organización, satisfacer las necesidades y expectativas de los interesados y garantizar el cumplimiento por parte del software de las reglas del negocio.

La DGSP, para quien se desarrolla el SIGEP, es la institución responsable de la organización y el funcionamiento de los centros de cumplimiento de pena y de medidas cautelares privativas de libertad en Venezuela. A grandes rasgos puede decirse que las deficiencias en el sistema penitenciario son múltiples y abarcan aspectos de infraestructura, legalidad, recursos humanos, gestión, comunicación y organización. (ARIAS y OTROS '06) De estas las que mayor influencia tienen en el desarrollo de la ERS son las asociadas con los recursos humanos y las organizativas.

En el aspecto organizativo se evidencia que no existe definición de algunos de los procesos claves de la organización. Los procesos existentes varían de un establecimiento a otro y en algunos casos violan las leyes vigentes y los derechos de los individuos; la documentación gestionada tampoco es estándar. Para resolver estos problemas se decidió iniciar la Modernización del Sistema Penitenciario. En el marco de este proyecto se están definiendo y reelaborando procesos, estructuras y formatos de documentos. Algunos puntos de los procesos definidos por el Equipo de Transformación Organizacional contradicen las opiniones y puntos de vista de los especialistas funcionales, quienes a su vez, en ocasiones se resisten a aceptar los cambios o tienen criterios condicionados por prácticas que se pretende revertir. Alcanzar la meta propuesta para la Ingeniería de Requisitos, impone en SIGEP, en primer lugar, **asegurar un**

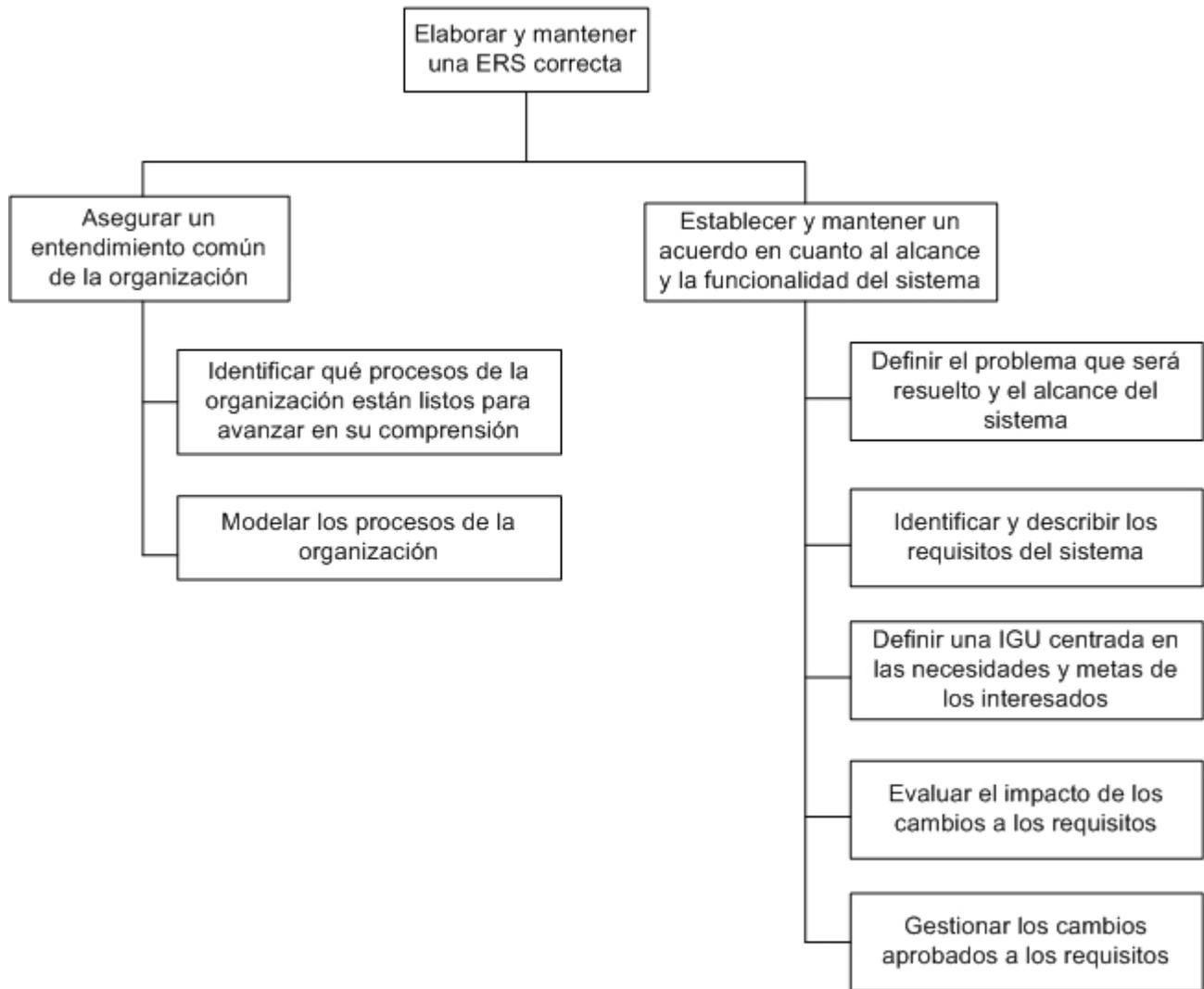
**entendimiento común de la organización** por parte de todos los grupos de interesados y del equipo de desarrollo que sirva como base para obtener un conjunto de requisitos que apoye los procesos de la organización y respete las reglas de negocio.

Alcanzar este objetivo, esta a su vez condicionado por **identificar los procesos de la organización que están listos para avanzar en su comprensión** y **modelar los procesos de la organización** con la participación de ambos grupos de interesados hasta alcanzar un entendimiento común respecto a ellos.

Luego de elaborar un Modelo de Negocio respecto al cual exista un entendimiento entre los interesados y el equipo de desarrollo, se debe establecer y mantener un **acuerdo en cuanto al alcance y la funcionalidad del sistema**. Para conseguir este objetivo es necesario:

- **Definir el problema que será resuelto y el alcance del sistema.**
- **Identificar y describir los requisitos del sistema.**
- **Definir una IGU centrada en las necesidades y metas de los interesados.**
- **Evaluar el impacto de los cambios a los requisitos y**
- **Gestionar los cambios aprobados a los requisitos.**

Partiendo de los elementos anteriormente expuestos en el proyecto SIGEP se definieron y organizaron los objetivos de la Ingeniería de Requisitos según se muestra en la Figura 3.



**Figura 3. Objetivos de la Ingeniería de Requisitos en SIGEP.**

## **2.2 Consideraciones para la elaboración del procedimiento**

Para conseguir los objetivos propuestos se redefinieron los flujos de trabajo Modelación de Negocio y Requisitos propuestos por RUP. En la elaboración de la propuesta se tuvieron como premisas RUP, las recomendaciones para desarrollar la ERS extraídas de la bibliografía consultada: (JACOBSON y *otros*. '04), (PRESSMAN '05), (FERNÁNDEZ '02), (IEEE '98a), (IEEE '98b), (PERALTA '03), (GONZÁLEZ y FERNÁNDEZ '06), (RATIONAL SOFTWARE '03a; YOUNG '02) (COCKBURN '00), (ADOLPH y BRAMBLE '01) (COCKBURN '00), el contexto para el desarrollo del SIGEP y la experiencia personal de las autoras en el desarrollo de la captura de requisitos en la primera etapa del proyecto.

La situación existente con los procesos de la DGSP impuso que se iniciara la identificación de los problemas a resolver por el software y la determinación del alcance del mismo antes de proceder a desarrollar un Modelo de Casos de Uso preliminar, como propone RUP. Este ambiente de cambios también impone desarrollar nuevas actividades de seguimiento explícito a la traza de los requisitos para mantener una ERS que cumpla las características señaladas por la (IEEE '98a). Además, el conjunto de actividades que se realizan debe permitir gestionar los cambios a la ERS de una manera ágil, por lo que debe limitarse a las que son realmente necesarias para el proyecto.

Con el objetivo de garantizar la construcción del producto correcto se introdujeron en el flujo de trabajo un conjunto de puntos de control: validaciones con el cliente y revisiones técnicas que no aparecían en el proceso propuesto por RUP. Adicionalmente se definieron validaciones de los requisitos a través de pequeñas partes funcionales del software entregadas al concluir cada iteración de la fase de Construcción.

En RUP la Modelación de Negocio y los Requisitos son dos flujos de trabajo diferentes. En el proyecto SIGEP, donde los procesos de negocio de DGSP no están bien definidos y el equipo de desarrollo no tiene un amplio dominio del funcionamiento de este, la Modelación del Negocio es una precondition para la obtención de los requisitos de software. Por ello se ha decidido definir cómo se realizará la Modelación del Negocio como parte del procedimiento para obtener los requisitos de software.

### **2.3 Modelación de negocio**

La Modelación del Negocio se realiza para alcanzar un entendimiento común de la organización por parte del equipo de desarrollo y los interesados en el proyecto.

Según los seis escenarios propuestos por RUP para la Modelación de Negocio, cuando se están redefiniendo los procesos de la organización es necesario modelar tanto los procesos actuales como los nuevos. En el caso del proyecto Modernización del Sistema Penitenciario el Equipo de Transformación Organizacional y el de desarrollo no son los mismos y dentro del alcance del proyecto SIGEP no está realizar la reingeniería del negocio. Esto quiere decir que los responsables de identificar y describir los nuevos procesos son los miembros del Equipo de Transformación Organizacional, por lo que para alcanzar la meta propuesta para la Modelación del Negocio en SIGEP es suficiente con modelar los procesos que estén definidos por la Transformación Organizacional, pero para ello es necesario primero

determinar qué procesos han sido definidos. Entonces, las actividades a desarrollar durante la Modelación de negocio son la Evaluación de la Organización y la Compresión del Negocio.

Considerando que el Equipo de Transformación Organizacional está en la obligación de describir los procesos de negocio, ¿por qué el equipo de desarrollo los modela? En primer lugar proponerse la modelación de los procesos de negocio como meta obliga a los Analistas a examinar y comprender los detalles de cada proceso, el modelo obtenido permite identificar puntos débiles o ausentes en la definición de procesos entregada a los Analistas y mejorarla, así como trabajar conjuntamente con el Equipo de Transformación Organizacional en lograr un entendimiento común de la organización.

Las entradas a la Modelación del Negocio son la Legislación Penitenciaria y los Documentos Resultantes de la Transformación Organizacional y se obtienen el Modelo de Negocio, el Glosario de términos y la Matriz de Seguimiento a los Requisitos.

La figura a continuación muestra el proceso Modelación de Negocio.

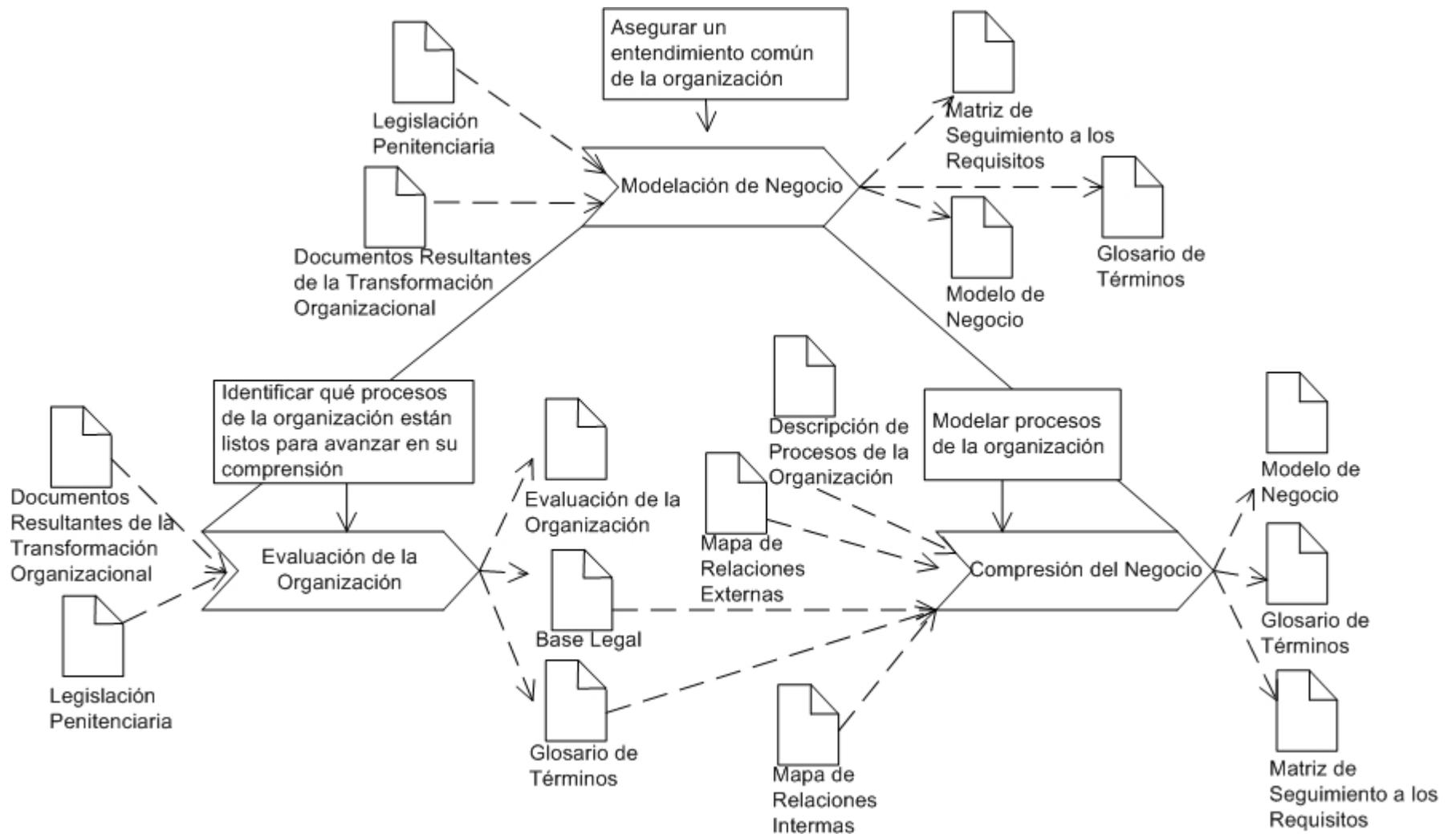


Figura 4. Modelación de Negocio.

### 2.3.1 Evaluación de la Organización

La Evaluación de la Organización está definida en la propuesta de RUP, su meta es establecer los objetivos de la Modelación del Negocio, en SIGEP se realiza en la fase de Inicio de cada etapa de desarrollo con el objetivo de identificar los procesos de la organización que están listos para avanzar en su comprensión.

La evaluación la inicia el Asesor en Transformación Organizacional con la **elaboración del instrumento para la evaluación de la organización** –el instrumento definido en SIGEP se muestra en el Anexo 1- , el cual define los aspectos que serán evaluados. Para **aplicar el instrumento para la evaluación de la organización** el Asesor busca elementos mediante la revisión del Informe Resultante de la Transformación Organizacional, la Descripción de Procesos de la Organización elaborada por el Equipo de Transformación y la Legislación Penitenciaria vigente. Realiza entrevistas tanto al Equipo de Transformación Organizacional como a especialistas venezolanos y cubanos y conduce sesiones de trabajo en grupo. Con los elementos obtenidos de la aplicación de estas técnicas se **elabora el informe** donde se registran los resultados de esta actividad, la Evaluación de la Organización. En RUP las tres actividades propuestas en SIGEP para evaluar la organización - resaltadas en negrita- se definen como una sola actividad.

Concluida la Evaluación de la organización el Líder de Proyecto, conjuntamente con los interesados **determina qué procesos de la organización se van a desarrollar en la etapa** y cuáles se postergan para las etapas siguientes. El principal criterio a considerar es el nivel de definición de cada proceso. Esta actividad es semejante a la actividad Establecer y Ajustar los Objetivos, propuesta por RUP.

Paralelamente a las actividades desarrolladas por el Asesor de Transformación Organizacional y el Líder de Proyecto, el Asesor Jurídico estudia la Descripción de Procesos de la Organización y la Legislación Penitenciaria para determinar qué aspectos de la legalidad debe cumplir cada proceso de la organización para **elaborar la Base Legal**, que no se limita sólo a hacer una correspondencia entre procesos y artículos de la ley, si no que contempla una explicación de cómo se aplica la ley en el proceso. Mientras desarrolla estas actividades los términos importantes, que puedan ser ambiguos o desconocidos son documentados en el Glosario de Términos. Esta actividad es una especialización de la propuesta por RUP para mantener las reglas de negocio en un ambiente donde la mayoría de las reglas de negocio y las más críticas son impuestas por la legislación. A continuación se muestra el flujo de actividades de este proceso.

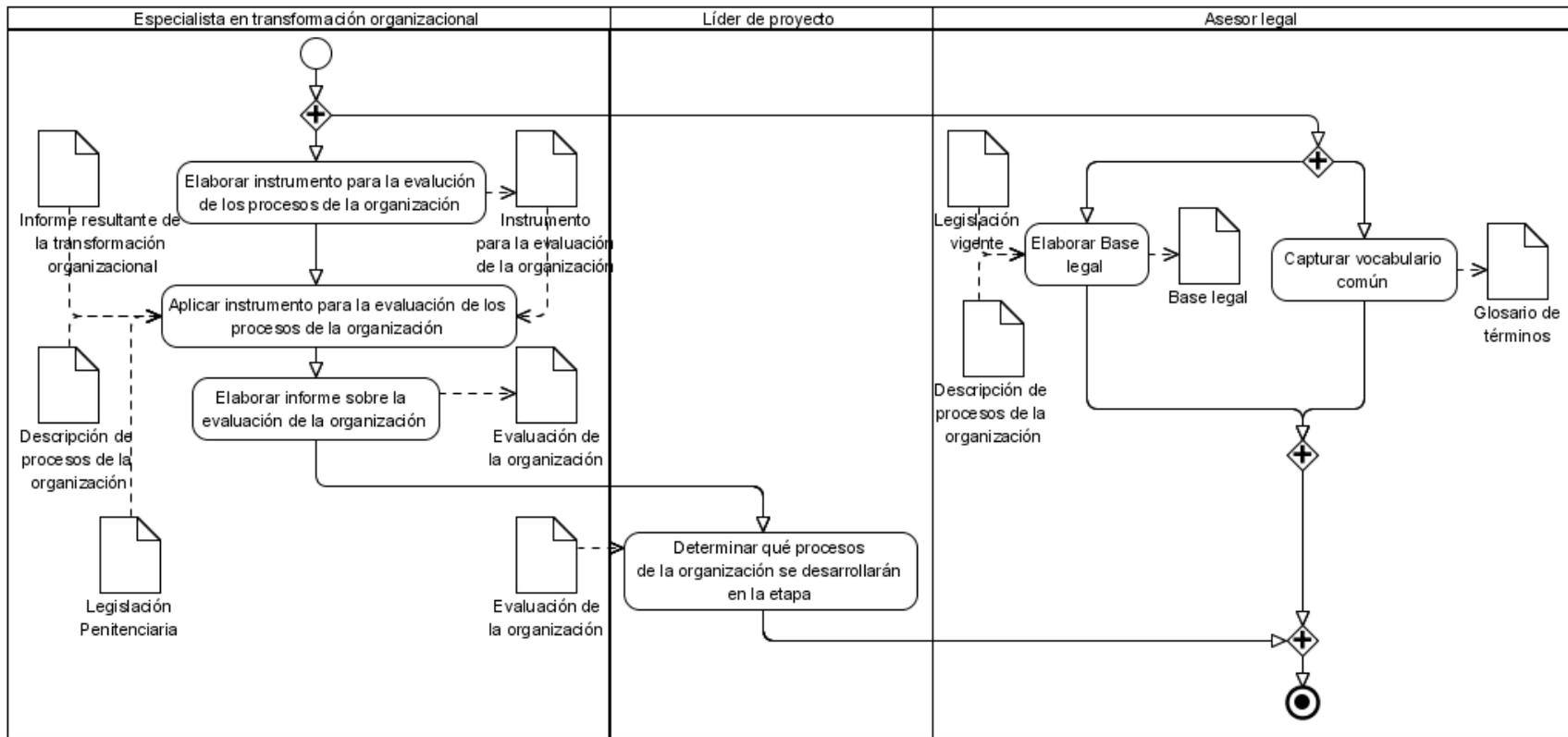


Figura 5. Evaluación de la organización.

### 2.3.2 Comprensión del Negocio

La Comprensión del Negocio se desarrolla durante la fase de Elaboración con el propósito de modelar los procesos de negocio para alcanzar un entendimiento común respecto a ellos entre el equipo de desarrollo y los interesados.

Esta actividad no está en RUP, pero las tareas que propone son similares a las de la Descripción del Negocio Actual, Refinamiento de la Definición de Procesos de Negocio, Diseño de las Realizaciones de Procesos de Negocio y Refinamiento de Roles y Responsabilidades. En RUP, algunas de estas actividades están dirigidas a identificar y describir casos de uso de negocio y otras a identificar y describir las realizaciones de estos casos de uso, en SIGEP, donde los procesos no se describirán en estos dos artefactos, sino como procesos de negocio, basta con describir y modelar de forma iterativa dichos procesos.

La transformación organizacional incluye la identificación y descripción de los procesos de negocio, durante la Comprensión del Negocio sólo se trabaja con procesos correctamente definidos, por tanto, el documento Descripción de Procesos de la Organización entregado a los Analistas ya tiene identificados y organizados los procesos y puede iniciarse su modelación y descripción. La **descripción de los procesos de negocio** se inicia con la identificación de los puntos débiles o ausentes, apoyándose en las siguientes preguntas (MEJÍAS y VEGA '06):

- ¿Están descritas todas las alternativas, correctas e incorrectas del flujo de actividades del proceso de negocio?
- ¿Se han identificado y descrito todas las entradas y salidas de los procesos de negocio?
- ¿Para obtener las salidas de los procesos identificadas es necesario describir aspectos específicos que no están en la Descripción de Procesos de la Organización?
- ¿Se contradice el contenido de la Descripción de Procesos de la Organización con lo indicado en el Mapa de Relaciones Internas y el Mapa de Relaciones Externas?
- ¿Contradice el proceso algún aspecto de la legislación vigente?

En la medida en que se van identificando puntos débiles o ausentes y surgen dudas los Analistas las deben aclarar con los interesados en sesiones de trabajo en grupo y continuar su modelación y

descripción hasta que de cada uno de los procesos definidos para la etapa hayan sido descritos todos los caminos básicos y alternativos, todas las entradas, salidas y actividades, y se hayan identificado los responsables de dichas actividades. Mientras se describen los procesos de negocio se **captura el vocabulario común** en el Glosario de Términos.

Seguidamente ocurren en paralelo ciclos de **validaciones de la Descripción de Procesos de Negocio** con los interesados y ciclos de **revisiones técnicas** formales hasta que la descripción refleje correctamente los procesos de negocio y se haya construido correctamente desde el punto de vista técnico. Las observaciones y señalamientos realizados tanto por los interesados como por el Revisor Técnico son registradas en el Listado de no Conformidades. Desde que comienza la descripción de los procesos se inicia el **registro** de los datos requeridos para cada Proceso de Negocio en **la Matriz de Seguimiento a los Requisitos**.

El flujo de actividades de este proceso se muestra a continuación.



## 2.4 Requisitos

El flujo de Requisitos se realiza para establecer y mantener un acuerdo entre el equipo de desarrollo y los interesados respecto al alcance y la funcionalidad del sistema.

La propuesta de RUP para este subproceso incluye la realización de las seis actividades descritas en el Capítulo I, en SIGEP sólo se plantean tres actividades, englobando una de ellas las principales tareas propuestas por RUP durante la Comprensión de las Necesidades de los Interesados, la Definición del Sistema, la Gestión del Alcance del Sistema y el Refinamiento de la Definición del Sistema.

Para alcanzar la meta propuesta en SIGEP es necesario definir el problema que será resuelto y el alcance del sistema, para lo cual se realiza el Análisis del problema. Además hay que identificar y describir qué debe hacer el sistema y las características que debe tener, definir una interfaz gráfica de usuario (IGU) centrada en las necesidades y metas de los interesados y proveer al equipo de desarrollo un mejor entendimiento del sistema, para alcanzar estos objetivos se desarrolla la Definición del Sistema. Finalmente, para mantener el acuerdo alcanzado, es necesario evaluar el impacto de los cambios a los requisitos y gestionar los cambios aprobados, para ello se desarrolla la Gestión de Cambios a los Requisitos.

El Modelo de Negocio, la Matriz de Seguimiento a los Requisitos, el Glosario de Términos y el Diagnóstico de la DGSP constituyen entradas a este proceso y las salidas son la Visión, la Especificación de Requisitos de Software y la Matriz de Seguimiento a los Requisitos.

La figura a continuación ilustra este proceso.

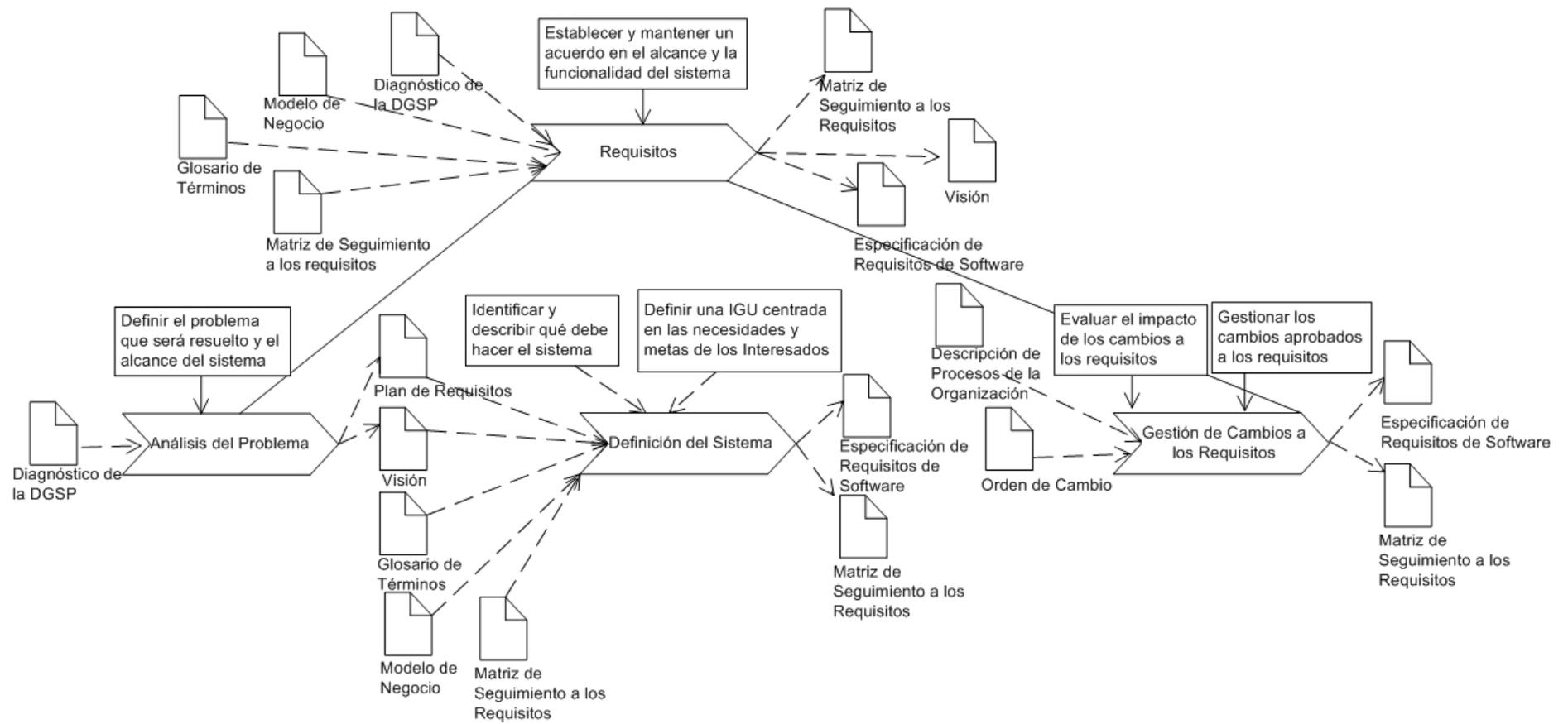


Figura 7. Requisitos

### 2.4.1 Análisis del Problema

El Análisis del Problema se realiza en la fase de Inicio del primer ciclo de desarrollo con el objetivo de definir el problema que será resuelto y el alcance del sistema.

En RUP esta actividad se inicia con la Identificación de Actores y Casos de Uso del Sistema, en SIGEP no era posible realizar esta tarea al inicio del proyecto porque no estaban definidos y modelados los procesos de negocio. El resto de las tareas propuestas por RUP para esta actividad se han mantenido en la presente propuesta.

Este subproceso se inicia con la **elaboración de la Visión** del sistema. Para ello el Líder de Proyecto estudia el Diagnóstico de la DGSP, documento que recoge el resultado del diagnóstico realizado a la institución en el 2005 e inicia la identificación de los problemas que pueden ser resueltos total o parcialmente por un software, en reuniones de trabajo conjuntas con los líderes por la parte venezolana del proyecto definen los problemas que definitivamente serán resueltos por el sistema y perfilan las características y funcionalidades que debe tener el software.

Mientras el Líder de Proyecto negocia para la elaboración de la Visión, las características del software y de la organización para la cual se desarrolla que van siendo identificadas sirven de entrada para que los Analistas **elaboren el Plan de Requisitos** para el proyecto. Este documento se somete a un ciclo de **revisiones** hasta que queda aprobado por el Revisor Técnico si cumple las normas y estándares definidos para el proyecto. Las observaciones y señalamientos son registrados en el Listado de no Conformidades.

A continuación se muestra el flujo de actividades de este subproceso.

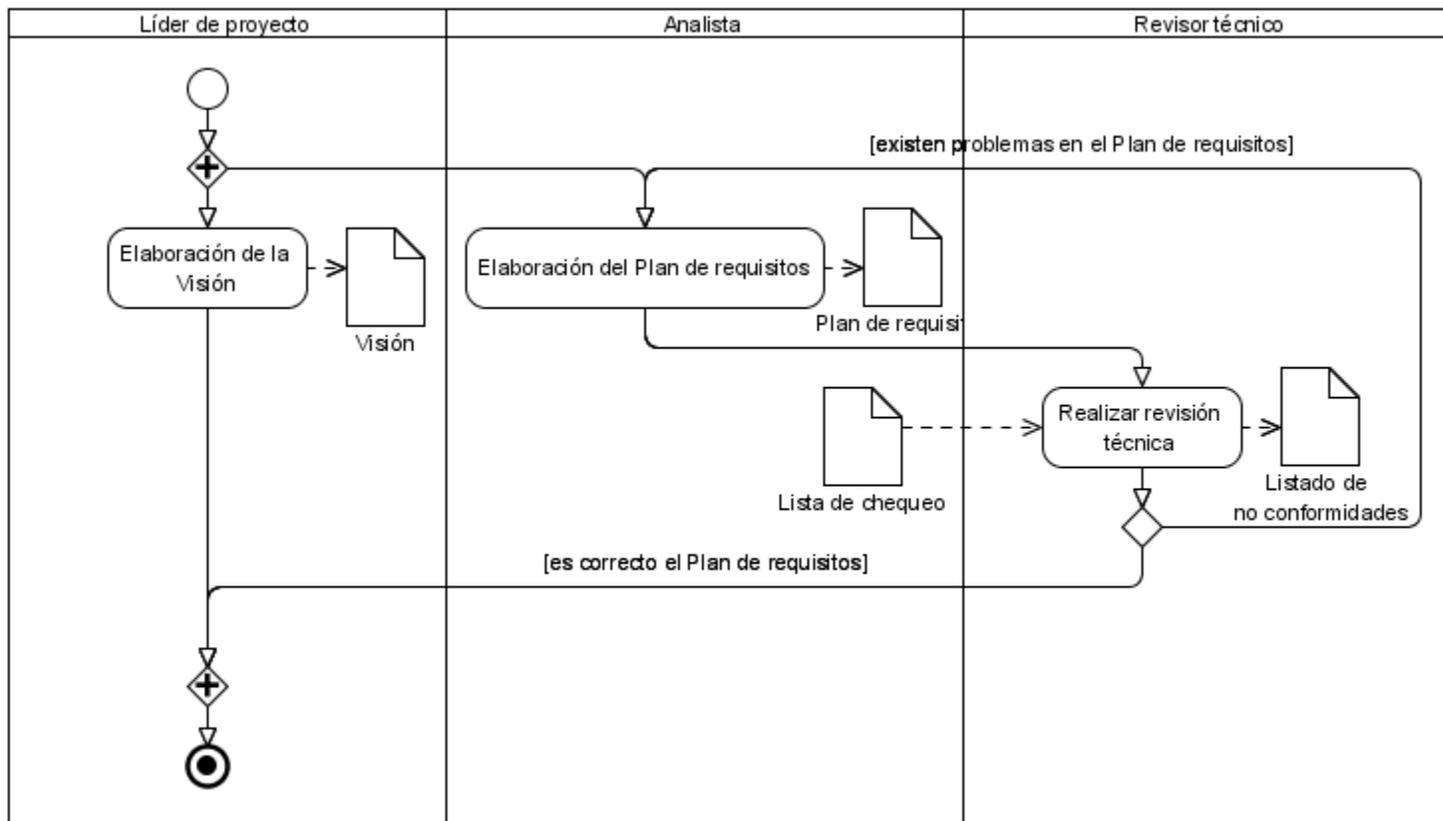


Figura 8. Análisis del problema.

## 2.4.2 Definición del Sistema

La Definición del Sistema se desarrolla en la fase de Elaboración con los siguientes objetivos:

- Identificar y describir qué debe hacer el sistema y qué características debe tener, esto es, los requisitos funcionales y no funcionales respectivamente.
- Definir una IGU centrada en las necesidades de los interesados.
- Proveer al equipo de desarrollo un mejor entendimiento del sistema.

Esta actividad define las principales tareas propuestas por RUP para identificar, describir y seguir la traza de los requisitos durante la Comprensión de las Necesidades de los Interesados, la Definición del Sistema, la Gestión del Alcance del Sistema y el Refinamiento de la Definición del Sistema.

Este subproceso se inicia con la **identificación de actores, casos de uso y salidas del sistema** a partir de la Visión y el Modelo de Negocio siguiendo las indicaciones del Plan de Requisitos. Mediante la revisión de los documentos mencionados anteriormente, entrevistas y reuniones grupales con los interesados, los Analistas identifican los trabajadores del negocio que pueden ser asistidos por el sistema en pasos o procesos que desarrollan, estos trabajadores se capturan como actores y los pasos o procesos como Casos de Uso y Salidas del Sistema. Todos son representados en el Modelo de Casos de Uso.

Seguidamente se procede a la **validación de los actores y requisitos funcionales identificados** hasta alcanzar un acuerdo con los interesados al respecto. Las no conformidades se recogen en el Listado de no Conformidades y se trabaja en ellas hasta que sean resueltas.

Teniendo como guía las plantillas de Casos de Uso y Salidas del Sistema se profundiza en el nivel de detalle de los requisitos describiendo primeramente los flujos básicos y el modelo conceptual, diccionario de datos y prototipo de IGU asociados a estos. Luego se procede a la identificación de los flujos alternativos y finalmente a su descripción con todos los elementos correspondientes. Los requisitos no funcionales identificados en esta actividad se incluyen en las Especificaciones de Casos de Uso, Salidas del Sistema y en las Especificaciones Suplementarias según corresponda.

Mientras se **describen los requisitos** se obtiene una visión más clara de las relaciones de inclusión, extensión y generalización/especialización existentes entre ellos, lo que permite **estructurar el Modelo de Casos de Uso**.

La descripción de los requisitos del sistema se desarrolla de forma iterativa y de conjunto con los interesados.

La Especificación de Requisitos de Software obtenida como resultado de las actividades anteriores, la cual contiene el Modelo de Casos de Uso, el Prototipo de Interfaz Gráfica de Usuario y las Especificaciones Suplementarias, es **validada con los interesados** y **evaluada por el Revisor Técnico** paralelamente. Este es un proceso cíclico hasta que se ha determinado que la Especificación de Requisitos de Software tiene las características descritas en el Capítulo I. Las no conformidades se registran en el Listado de no Conformidades y se trabaja en ellas hasta solucionarlas.

Durante este proceso la **Matriz de Seguimiento a los Requisitos es actualizada** continuamente con los requisitos identificados y la modificación, cuando corresponda, de sus atributos respectivos. El responsable de esta actividad es el Analista.

La figura siguiente muestra el flujo de actividades correspondiente a este subproceso. El Glosario de Términos y la Matriz de Seguimiento a los Requisitos pueden ser consultados durante todas las actividades de este subproceso, para evitar añadir complejidad a la figura no se representan en la misma.

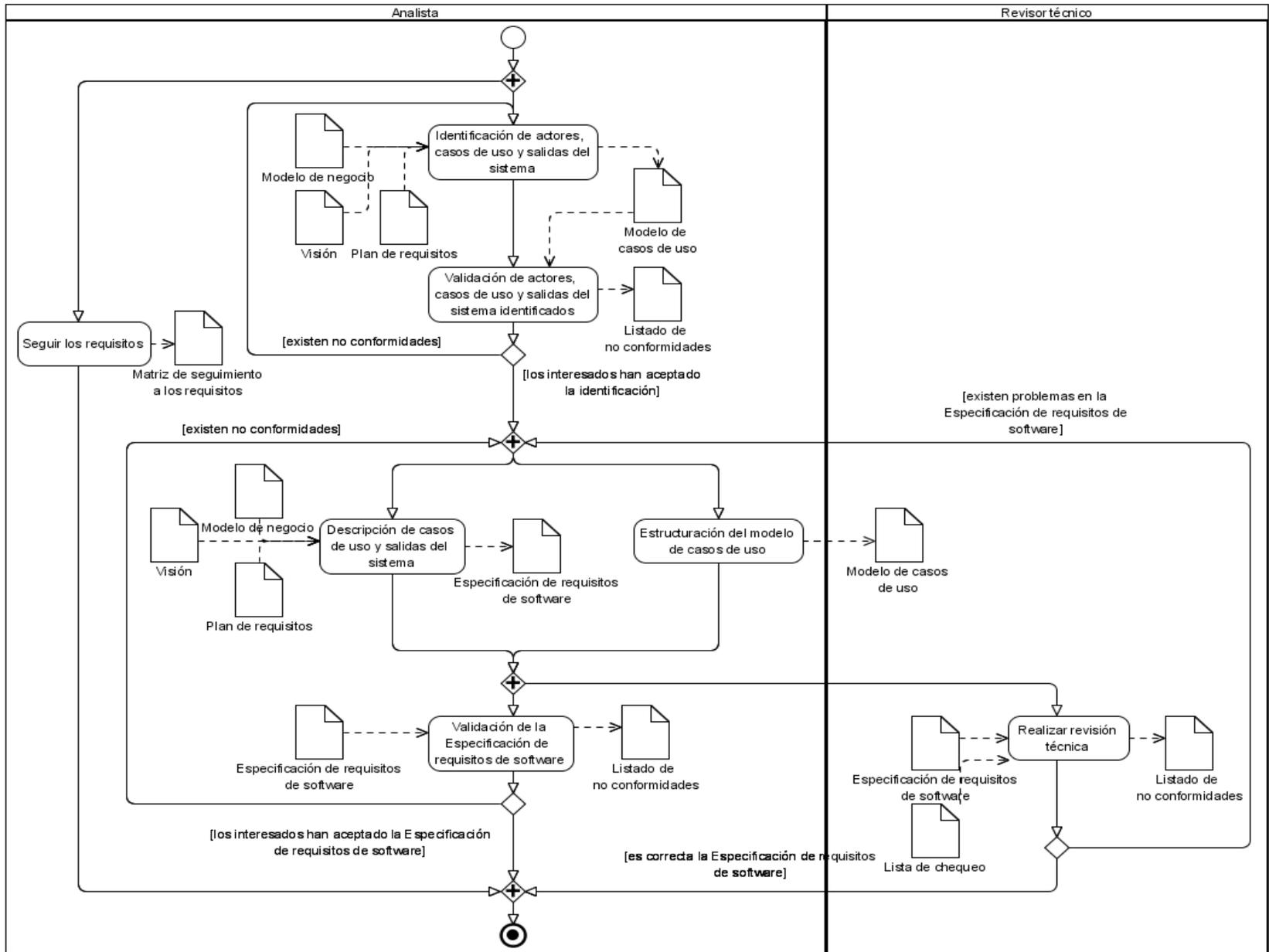


Figura 9. Definición del Sistema.

### 2.4.3 Gestión de Cambios a los Requisitos

La Gestión de Cambios a los Requisitos es un proceso que se desarrolla durante todo el ciclo de vida del software con el objetivo de evaluar el impacto de los cambios y de gestionar los cambios aprobados. Es útil recordar la gran probabilidad de que cambien los requisitos durante el proceso de desarrollo y que pueden aparecer nuevos requisitos, aunque deben representar un por ciento mínimo del grupo total de requisitos identificados si la Definición del Sistema se desarrolló correctamente.

Mediante el análisis de la Matriz de Seguimiento a los Requisitos se puede determinar qué elementos se verían afectados por los cambios solicitados y **evaluar su impacto** en función de la complejidad de los requisitos o artefactos que se ven afectados.

Una vez que se ha aprobado un cambio y emitido la Orden de Cambio correspondiente para llevarlo a cabo se realizan las **modificaciones pertinentes en la Especificación de Requisitos de Software**, la cual es sometida nuevamente a la **validación con los interesados** y a la **evaluación por parte del Revisor Técnico** como se explicó en el punto anterior. Durante esta etapa también es necesario **mantener la Matriz de Seguimiento a los Requisitos**.

Esta actividad es similar a la propuesta por RUP, la propuesta de SIGEP organiza las tareas en un flujo de actividades. Este se muestra a continuación.

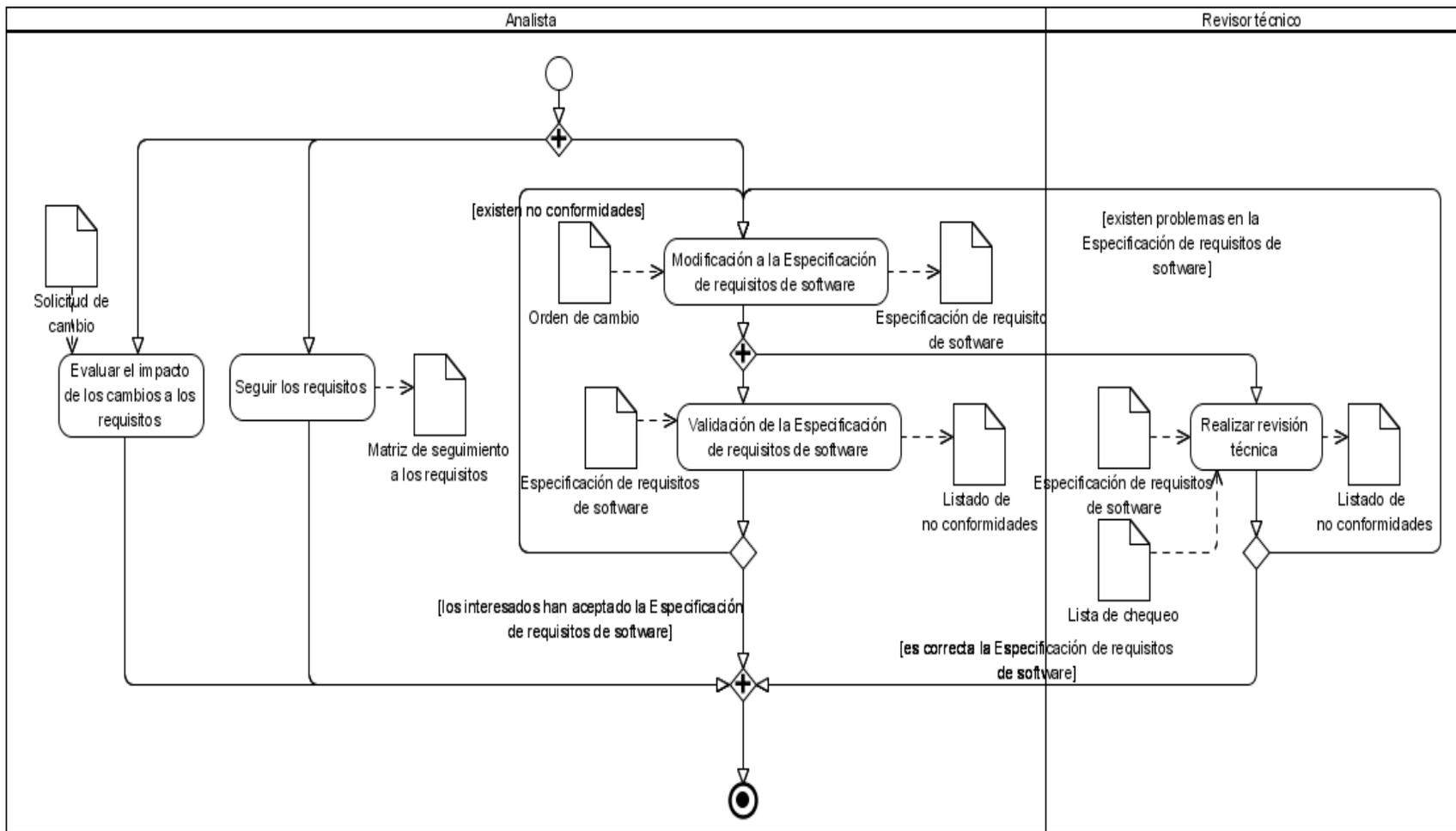


Figura 10. Gestión de Cambios a los Requisitos.

#### 2.4.4 Seguimiento a los requisitos

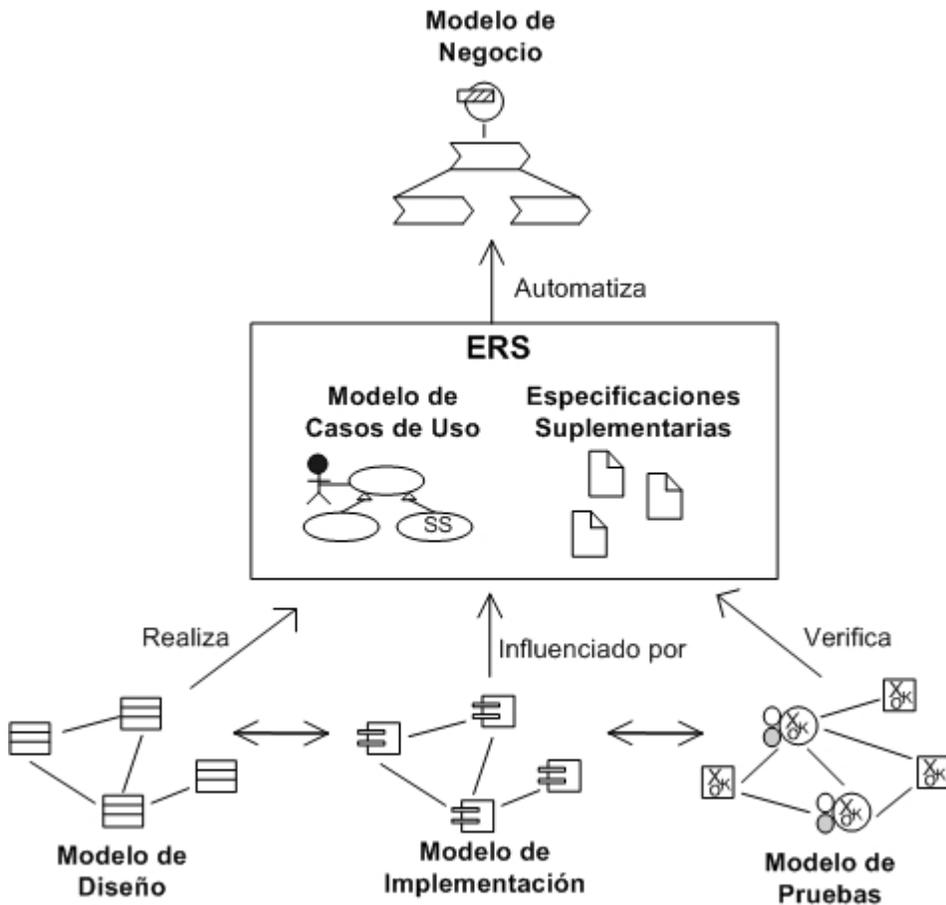
En el Capítulo I se definió que mantener la traza de los requisitos es la habilidad de seguir la evolución de los requisitos durante el ciclo de vida del software y que en proyectos donde se prevén frecuentes cambios a los requisitos es útil mantenerla explícitamente, sin embargo siempre existe cierta traza implícita.

En el caso de SIGEP, el Proyecto Técnico (ARIAS y OTROS '06) define los siguientes artefactos por disciplinas.

Disciplina	Artefactos
<b>Disciplinas del Proceso</b>	
Modelación del Negocio	Evaluación de áreas de la organización
	Base Legal
	Descripción de Procesos de Negocio
Requisitos	Visión
	Plan de Captura de Requisitos
	Modelo de Casos de Uso
	Especificaciones suplementarias
	Prototipo de Interfaz de Usuario
Análisis y Diseño	Modelo Lógico de Datos
	Modelo de Diseño
	Descripción de la Arquitectura
	Modelo Físico de Datos
Implementación	Modelo de Implementación
	Modelo de Despliegue
Pruebas	Plan de pruebas
	Listado de datos de prueba
	Casos de prueba
	Resultado de pruebas
Despliegue	Plan de aceptación del producto
	Plan de pruebas
	Listado de datos de prueba
	Casos de prueba
	Resultado de pruebas
	Material de entrenamiento
	Material de soporte al usuario final
Artefactos de Instalación	

**Tabla 1. Artefactos que se deben elaborar en SIGEP.**

Partiendo de los artefactos definidos anteriormente, se ha definido seguir la traza de los requisitos según se muestra en la siguiente figura.



**Figura 11. Traza de los requisitos a través de los artefactos que se deben elaborar en SIGEP.**

Para evitar añadir mayor complejidad a la figura se omitió de la misma el Glosario de Términos, pero todos los artefactos representados pueden contener términos incluidos en el Glosario y por tanto están relacionados con él. En este caso la traza se mantiene a través de una referencia desde los artefactos que utilizan un término hacia el Glosario de Términos.

El Modelo de Negocio contiene procesos de negocio y reglas que los gobiernan, la ESR automatiza los procesos de negocio garantizando el cumplimiento de estas reglas. Los procesos de negocio se detallan en la Descripción de Procesos de Negocio y las reglas en la Base Legal, entre ellos existen una relación de traza que debe mantenerse mediante referencias. Los casos de uso y salidas del sistema, contenidos en el Modelo de Casos de Uso, y los requisitos no funcionales, contenidos en las Especificaciones

Suplementarias, automatizan procesos de negocio; para conocer la fuente de estos requisitos y evaluar el impacto de los cambios en ellos es útil conocer de cada caso de uso y salida del sistema qué procesos de negocio apoya. Ni el Modelo de Casos de Uso ni el Modelo de Negocio permiten el mantenimiento implícito de esta relación por lo que es necesario seguir explícitamente el elemento proceso de negocio hacia los requisitos del sistema. Tanto los procesos de negocio como los casos de uso, salidas del sistema y requisitos no funcionales son elementos de seguimiento a la traza de los requisitos.

Un elemento de seguimiento a la traza de los requisitos, en lo adelante, elemento de seguimiento, es cualquier elemento textual o modelo que es necesario seguir explícitamente desde cualquier otro elemento textual o modelo para mantener las dependencias entre ellos. (SPENCE y PROBASCO '98)

Los requisitos deben garantizar el cumplimiento de las reglas de negocio, para ello de cada proceso se identifican las reglas que tienen influencia en los requisitos que los automatizan, esta relación debe mantenerse explícitamente. El Modelo de Pruebas debe verificar que la ERS respete las reglas del negocio de la organización.

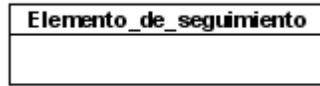
Los casos de uso y salidas del sistema pueden contener requisitos no funcionales, esta relación se mantiene mediante referencias desde las Descripciones de Casos de Uso y las Descripciones de Salidas del Sistema, hacia las Especificaciones Suplementarias.

El Modelo de Diseño describe las realizaciones de diseño del Modelo de Casos de Uso, esta relación es mantenida implícitamente asociando a cada caso de uso el sub-Modelo de diseño que lo realiza. El Modelo de Implementación se elabora a partir del Modelo de Diseño y recibe influencias de la ERS, estas relaciones pueden ser mantenidas organizando los subsistemas de implementación en correspondencia con los paquetes de diseño, por lo que no es necesario mantenerla explícitamente.

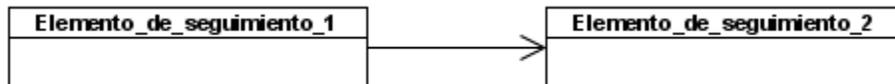
El Modelo de Pruebas verifica el cumplimiento de los requisitos en el Modelo de Implementación, para determinar el impacto de los fallos en las pruebas en cada requisito específico es necesario mantener una relación explícita entre estos y los casos de prueba.

Las relaciones entre los elementos de seguimiento a los requisitos descritos anteriormente se representan en la figura 12 utilizando el lenguaje UML de la siguiente manera (SPENCE y PROBASCO '98):

Las clases son usadas para representar los elementos de seguimiento a los requisitos.



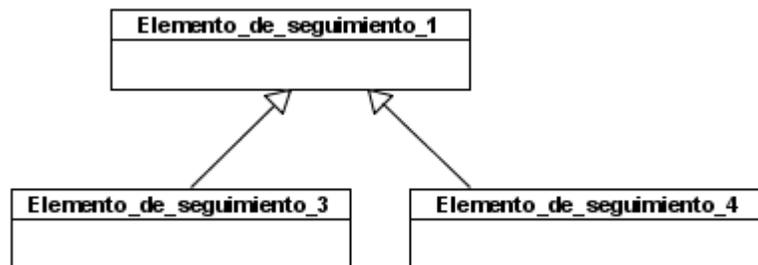
Las relaciones direccionales entre dos elementos de seguimiento a los requisitos son empleadas para representar las trazas. En el ejemplo a continuación Elemento\_de\_seguimiento\_1 puede seguirse hacia Elemento\_de\_seguimiento\_2.



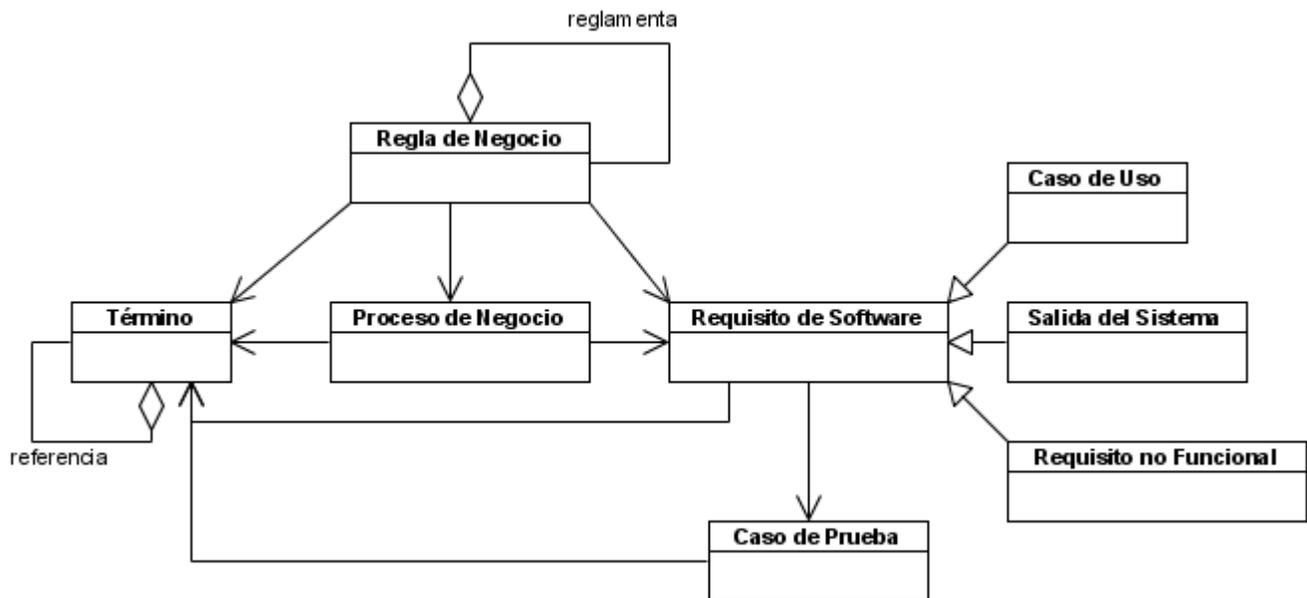
Las agregaciones recursivas son usadas para mostrar relaciones de jerarquía entre elementos de seguimiento a los requisitos. El nombre de la relación es utilizado para clarificar la relación.



La relación de generalización es empleada para representar subtipos de un elemento de seguimiento a los requisitos.



Atendiendo a lo anteriormente descrito se ha determinado seguir explícitamente la traza de los requisitos según se muestra en la figura siguiente.



**Figura 12. Elementos de seguimiento a los requisitos.**

De los elementos de seguimiento a los requisitos es útil mantener atributos, estos son usados para administrar el alcance del sistema, negociar la inclusión de los requisitos en una iteración u otra y determinar el avance del proyecto. Pueden ser empleados por los arquitectos para identificar los casos de uso arquitectónicamente significativos. (RATIONAL SOFTWARE '03a) Además, son útiles para crear vistas de los requisitos, por ejemplo, mostrar todos los casos de uso con una estabilidad alta.

Al igual que los elementos de seguimiento, los atributos, sus posibles valores y criterios para determinarlos deben ser definidos por cada proyecto según sus características. A continuación se describen los elementos de seguimiento en el proyecto SIGEP, los atributos de cada elemento, posibles valores de los atributos y criterios para determinarlos.

Todos los elementos de seguimiento a los requisitos deben tener un identificador que será definido por el Equipo de Gestión de Configuración según las políticas para nombrar todos los elementos producidos durante el ciclo de vida.

## **Regla de Negocio**

Una Regla de Negocio describe políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. (RATIONAL SOFTWARE '03a)

De cada proceso de negocio se identifican y describen las reglas que lo gobiernan, las cuales son evaluadas para determinar si son relevantes dentro del campo de acción del software y reflejadas mediante los requisitos funcionales y no funcionales.

Las reglas del negocio pueden hacer referencias a otras reglas del negocio o a términos en el Glosario, de ellas no se mantienen atributos excepto el identificador.

## **Proceso de Negocio**

Un Proceso de Negocio es un elemento de seguimiento a los requisitos que representa los procesos explicados en las Descripciones de Procesos de Negocio. Son útiles para evaluar el impacto de los cambios en la organización en los requisitos del sistema.

Los atributos que se mantienen de este elemento son:

### Estado

Este atributo permite mantener información sobre el avance de la Modelación de Negocio. Puede tomar los siguientes valores:

- **Identificado:** es un proceso que se ha identificado pero no ha sido descrito y aprobado conjuntamente por el equipo de desarrollo y los interesados.
- **Descrito:** es un proceso de negocio que ha sido descrito y aún no se ha aprobado en las validaciones y revisiones técnicas.
- **Aprobado:** es un proceso de negocio respecto al cual se ha alcanzado y expresado explícitamente un entendimiento común entre los interesados y el equipo de desarrollo.

### Estabilidad

Este atributo es usado para establecer prioridades en el desarrollo y para determinar qué Procesos de Negocio deben ser considerados en próximos intercambios con los interesados. Los valores de este atributo, que pueden ser: Alta, Media y Baja, se asignarán por el equipo de desarrollo y los interesados

atendiendo a los resultados de la evaluación realizada por el Especialista en Transformación Organizacional.

### **Requisito de Software**

Un Requisito de Software es una condición o capacidad con la cual el software debe cumplir. (RATIONAL SOFTWARE '03a) Como elemento de seguimiento a los requisitos es especializado por los Casos de Uso, Salidas del Sistema y Requisitos no Funcionales.

Sus atributos son:

#### Estado

Este atributo permite mantener información sobre el progreso del proyecto. Puede tomar los valores:

- Propuesto: es un requisito que se ha identificado pero no se ha acordado por parte de los interesados y el equipo de desarrollo su inclusión en el sistema.
- Aprobado: es un requisito que se estima útil y factible, cuya inclusión en el sistema ha sido aprobada por los interesados y el equipo de desarrollo.
- Incorporado: es un requisito que se ha implementado.
- Probado: es un requisito cuya inclusión en el sistema ha sido validada y verificada. Es decir, se ha comprobado que el sistema cumple este requisito y que lo hace correctamente.

#### Beneficio

Conocer el beneficio que aporta la implementación de un requisito a los interesados es útil para la planificación. Los posibles valores de este atributo son:

- Crítico: es un requisito esencial, si no está presente el sistema no contribuirá efectivamente a apoyar los procesos de negocio y no satisfará las expectativas de los interesados. Los fallos en la implementación de este tipo de requisitos provocan retrasos en el cronograma.
- Importante: es un requisito importante para la efectividad y eficiencia del producto. La funcionalidad no puede ser fácilmente provista de otra forma. No incluir un requisito importante puede afectar la satisfacción de los interesados, pero no debe afectar el cronograma.

- Deseado: es un requisito que será usado escasamente o cuya funcionalidad puede ser provista de alguna otra manera. No debe tener impactos serios en la satisfacción de los interesados.

#### Versión objetivo

El propósito de este atributo es indicar la versión del producto en la que el requisito debe aparecer por primera vez. Se le asignará un valor según la nomenclatura que se defina para las versiones por el equipo de Gestión de Configuración.

#### Asignado a

El objetivo es conocer qué persona o equipo tiene la responsabilidad de implementar el requisito. Los valores válidos son cadenas de caracteres con el nombre de la persona o equipo responsable.

#### Solicitado por

Conocer el origen de los requisitos es favorable para identificar aquellos que son realmente útiles y para las negociaciones en proyectos donde existen cambios frecuentes. El objetivo de este atributo es conocer quién solicitó y aprobó la inclusión de cada requisito.

De cada requisito debe conocerse quién solicitó y aprobó su incorporación al sistema (nombre y apellidos), cargo que ocupa en la organización y área funcional a la que pertenece.

### **Casos de Uso**

Un Caso de Uso es una descripción de la interacción de un actor con el sistema que le permite obtener un resultado de valor, cada Especificación de Caso de Uso describe un Caso de Uso.

Los Casos de Uso son un elemento de seguimiento a los requisitos imprescindible porque ellos contienen alrededor de la tercera parte de los requisitos del sistema (COCKBURN '00) y constituyen una guía para el desarrollo del software, las estimaciones y la planificación.

Los atributos de los Casos de Uso son:

#### Antecedentes

Permite mantener la relación de derivación, indicando a partir de que Procesos de Negocio se derivó el Caso de Uso. El valor de este atributo es una lista de los identificadores de elementos de seguimiento a los requisitos a partir de los cuales se derivó el Caso de Uso.

### Estabilidad

Este atributo es usado para establecer prioridades en el desarrollo y para determinar qué Casos de Uso deben ser considerados en próximas iteraciones e intercambios con los interesados. Los valores de este atributo, que pueden ser: Alta, Media y Baja, se asignarán por el equipo de desarrollo y los clientes considerando la posibilidad de que el requisito cambie. Los siguientes factores deben ser tomados en cuenta:

- No se ha iterado sobre la definición del Caso de Uso.
- El proceso de negocio que es automatizado por el Caso de Uso no está completamente definido en la organización o puede sufrir cambios.
- No existe un entendimiento por parte de los interesados y el equipo de desarrollo respecto al Caso de Uso.

### **Requisito no Funcional**

Los Requisitos no Funcionales describen atributos del sistema o del entorno del sistema, se detallan en las Especificaciones Suplementarias. Ellos son especializaciones de los Requisitos del Sistema y no se mantienen atributos adicionales.

### **Salida del Sistema**

Las Salidas del Sistema representan los elementos descritos en el artefacto Descripción de las Salidas del Sistema.

Sus atributos son:

### Antecedentes

Permite mantener la relación de derivación, indicando a partir de que Procesos de Negocio se derivó la Salida del Sistema. El valor de este atributo es una lista de los identificadores de elementos de seguimiento a los requisitos a partir de los cuales se derivó la Salida del Sistema.

### Estabilidad

Este atributo es usado para establecer prioridades en el desarrollo y para determinar qué Salidas del Sistema deben ser consideradas en próximas iteraciones e intercambios con los interesados. Los valores de este atributo, que pueden ser: Alta, Media y Baja, se asignarán por el equipo de desarrollo y los

clientes considerando la posibilidad de que el requisito cambie. Los siguientes factores deben ser tomados en cuenta:

- No se ha iterado sobre la definición de la Salida del Sistema.
- El proceso de negocio que es automatizado por la Salida del Sistema no está completamente definido en la organización o puede sufrir cambios.
- No existe un entendimiento por parte de los interesados y el equipo de desarrollo respecto a la Salida del Sistema.

### **Caso de Prueba**

Un Caso de Prueba es una especificación del conjunto de entradas, condiciones de ejecución y resultados esperados que evalúan los distintos aspectos que componen el sistema, están contenidos en el artefacto Caso de Prueba, que se elabora como parte del flujo de trabajo de Pruebas.

Mantener este elemento de seguimiento es útil porque permite saber qué requisitos han sido probados y qué requisitos se ven afectados por determinado fallo detectado durante las pruebas.

Los atributos de los CP son:

#### Estado

El objetivo de este atributo es mantener información sobre el avance de las pruebas. Puede tomar los valores:

- Elaborado: es un Caso de Prueba que se ha diseñado completamente pero que aún no se ha aplicado.
- Ejecutado: es un Caso de Prueba que ya ha sido ejecutado y del cual se tienen los resultados.

#### Antecedentes

Permite conocer qué Salidas del Sistema, Casos de Uso y Requisitos no Funcionales son verificados por el Caso de Prueba. El valor de este atributo es una lista de los identificadores de elementos de seguimiento a los requisitos a partir de los cuales se derivó el Caso de Prueba.

## Asignado a

El objetivo es conocer qué persona o equipo tiene la responsabilidad de elaborar y ejecutar cada Caso de Prueba. Los valores válidos son cadenas de caracteres con el nombre de la persona o equipo responsable.

## **Término**

Un Término permite definir las palabras relevantes o ambiguas tanto en el dominio del problema como técnicamente. Estos están contenidos en el Glosario de Términos y pueden ser referenciados desde cualquier artefacto o documento, de ellos no se mantienen atributos.

## **2.5 Descripción de los artefactos**

### **2.5.1 Artefactos elaborados por el Equipo de Desarrollo**

En el Capítulo I se explicó la necesidad de elaborar sólo los artefactos necesarios según las características de cada proyecto. En el caso de SIGEP se ha definido desarrollar los siguientes artefactos durante la Modelación de Negocio y la Ingeniería de Requisitos.

Disciplina	Artefactos	Fases			
		Inicio	Elaboración	Construcción	Transición
<b>Disciplinas del Proceso</b>					
Modelación de Negocio	Base Legal	c	r		
	Descripción de Procesos de Negocio		c	r	
	Evaluación de la Organización	c			
	Modelo de Negocio		c	r	
Requisitos	Descripción de las Salidas del Sistema		c	r	r
	Especificación de Casos de Uso		c	r	r
	Especificaciones Suplementarias		c	r	r
	Glosario de Términos	c	r	r	r
	Matriz de Seguimiento a los Requisitos		c	r	r
	Modelo Conceptual		c	r	r
	Diccionario de Datos		c	r	r
	Modelo de Casos de Uso		c	r	r
	Plan de Requisitos	c	r	r	
	Prototipo de IGU		c	r	
Visión	c	r	r		

**Tabla 2. Artefactos propuestos para desarrollar la Modelación de Negocio y la Ingeniería de Requisitos en SIGEP.**

c- comenzar; r- refinar

## Base Legal

La Base Legal persigue registrar los aspectos jurídicos que deben considerarse en cada proceso de negocio. Se elabora durante la Modelación de Negocio.

Este artefacto no está en la propuesta de RUP, fue definido en SIGEP debido a la necesidad de que el equipo de desarrollo conozca los elementos legales que rigen los procesos de la DGSP y cómo influyen en ellos. Recoge toda la base legal, normativas y reglamentos que sustenten formalmente los procesos que serán objeto de informatización.

Se listan alfabéticamente los procesos de negocio, a su vez, para cada proceso se listan los artículos que le corresponden ordenados alfabéticamente por el nombre de la ley, normativa o reglamento y se describe la influencia del artículo en el proceso. A continuación se muestra la estructura que debe tener el cuerpo del documento.

### **<Nombre del proceso de negocio X>**

<Nombre de la ley, reglamento o normativa>Artículo <Número del artículo Y>  
<Descripción de la influencia de Y en X>

<Nombre de la ley, reglamento o normativa>Artículo <Número del artículo Z>  
<Descripción de la influencia de Z en X>

### **<Nombre del proceso de negocio A>**

<Nombre de la ley, reglamento o normativa>Artículo <Número del artículo B>  
<Descripción de la influencia de B en A>

<Nombre de la ley, reglamento o normativa>Artículo <Número del artículo C>  
<Descripción de la influencia de C en A>

## Descripción de las Salidas del Sistema

Las Salidas del Sistema tienen como objetivo describir las funcionalidades del sistema que se limitan a mostrar información a los actores. Se identifican y describen durante el flujo de Requisitos.

Este artefacto no está en la propuesta de RUP pero fue definido en SIGEP debido a los requerimientos informativos que necesita periódicamente la institución y que deben mostrarse como reportes.

La Descripción de las Salidas del Sistema contiene: el formato de la Salida del Sistema, los datos que debe mostrar, la manera de representación de los mismos y los actores involucrados. Para los datos se definen los tipos de letra, tamaños, colores, espacios, encabezados, y en general, todo aquello relacionado con la presentación del informe; lo mismo ocurre con las tablas y gráficos. A continuación se muestra el formato de las Salidas del Sistema.

<b>Objetivo</b>		<i>Explicar el objetivo del reporte.</i>			
<b>Actores</b>		<i>En esta sección debe especificarse qué actores generan el reporte</i>			
<b>Precondición</b>		<i>La precondición declara de qué debe estar seguro el sistema antes de autorizar el inicio de la salida del sistema. Esto no debe ser chequeado de nuevo mientras se genera la salida del sistema. Usualmente tener una precondición indica que otro caso de uso o salida del sistema está o se ha ejecutado. Una precondición debe escribirse como una afirmación simple. Por ejemplo: El cliente ha sido validado.(COCKBURN '00)</i>			
<b>Garantías</b>	<b>Mínimas</b>	<i>Es lo mínimo a lo que se debe comprometer el sistema con los interesados. Tienen mayor importancia cuando la meta principal no puede ser cumplida. Debe escribirse como una afirmación simple, que debe ser cierta al terminar el caso de uso.(COCKBURN '00)</i>			
	<b>De éxito</b>	<i>Establece qué intereses deben ser satisfechos cuando se haya terminado exitosamente la salida del sistema. Debe escribirse como una afirmación simple.(COCKBURN '00)</i>			
<b>Acción que inicia la generación del reporte</b>		<i>Especifica qué evento debe ocurrir para que el sistema inicie la generación del reporte</i>			
<b>FORMATO</b>					
<b>Dimensiones</b>					
Ancho		Largo			
<b>Fuente</b>					
Tipo		Tamaño		Color	
<b>Párrafo</b>					
Alineación		Interlineado		Entre párrafos	
Tipo de documento		<i>Tipo de formato a exportar, PDF, o XLS (los XLS, sin imágenes)</i>			
Encabezado		<i>A veces suele ser el título (puede ser cualquier cosa, de cuantas páginas se necesite, una foto, un logo, una frase, etc)</i>			
Encabezado de página		<i>Qué se desea que sea el encabezado de cada página (una frase, un logo, etc.)</i>			
Pie de página		<i>Qué se desea que sea el pie de cada página (una frase, un logo, la hora y el día en que se hizo, etc.)</i>			
Resumen		<i>Lo último que sale del documento (una frase, un logo, la hora y el día</i>			

	<i>en que se hizo, etc.)</i>
Origen de los datos	<i>Modelo conceptual que representa la fuente de los datos</i>
Ejemplo de formato	<i>En esta sección se pondrá un ejemplo del reporte, si existe algún formato ya establecido en la institución.</i>
<b>ELEMENTOS A MOSTRAR</b>	

*Para las tablas*

Nombre					
Dimensiones					
Ancho		Largo			
Fuente					
Tipo		Tamaño		Color	
Párrafo					
Alineación		Interlineado		Entre párrafos	
Cálculos sobre campos			<i>Cálculos sobre campos (totales, promedios, máximos, mínimos).</i>		
Valores a destacar			<i>Valores a destacar, por ejemplo, de un listado de individuos se quieren resaltar los que llevan más de 1 año presos</i>		

*Para los gráficos*

Nombre					
Dimensiones					
Ancho		Largo			
Fuente					
Tipo		Tamaño		Color	
Formato					
Tipo			<i>Si es de barra, pastel, etc</i>		
Campos que debe mostrar			<i>Qué datos debe tener el gráfico</i>		

## Descripción de Procesos de Negocio

La Descripción de Procesos de Negocio persigue describir el comportamiento de cada proceso identificado en el Modelo de Negocio.

En RUP para describir los procesos del negocio se utilizan los Casos de uso del Negocio y la Especificación de Casos de Uso del Negocio. En SIGEP se definió el artefacto Descripción de Procesos de Negocio que contiene una descripción textual del mismo y un diagrama BPMN –notación propuesta por el consorcio *Business Process Management Initiative*- para representar el flujo de actividades de cada proceso. La tabla a continuación muestra el formato de los Procesos de Negocio.

<b>Descripción del proceso &lt;Nombre del proceso&gt;</b>	
<b>Objetivo</b>	<i>Indicar el objetivo del proceso</i>
<b>Clientes Internos</b>	<i>Listar los clientes del proceso que están dentro de la misma organización</i>
<b>Clientes Externos</b>	<i>Listar los clientes del proceso que pertenecen a otras organizaciones</i>
<b>Indicadores del proceso</b>	<i>Indicar las expresiones cualitativas o cuantitativas que auxiliadas de métricas indican el nivel de eficiencia, eficacia y efectividad con que se ejecuta el proceso.</i>
<b>Subprocesos</b>	<i>Listar los subprocesos incluidos en el proceso</i>
<b>Evento(s) que lo genera(n)</b>	<i>Indicar qué eventos inician el proceso</i>
<b>Sustento Jurídico</b>	<i>Referenciar los elementos correspondientes en la Base Legal</i>
<b>Precondiciones</b>	<i>La precondición declara las condiciones que deben existir para que se inicie el proceso.</i>
<b>Poscondiciones</b>	<i>La poscondición declara las condiciones en que debe quedar la organización al concluir el proceso.</i>
<b>Régimen en que se aplica</b>	<i>Existen dos regímenes penitenciarios en Venezuela: intramuros y extramuros, aquí se debe indicar en cuál o cuáles tiene lugar este proceso.</i>
<b>FLUJO BÁSICO</b>	
<b>Diagrama principal del proceso</b>	
<i>Incluir el diagrama del proceso</i>	
<b>Descripción del Flujo Básico</b>	
Paso <X> Descripción del paso	
<b>DESCRIPCIÓN DE LOS FLUJOS PARALELOS</b>	
<b>Nombre del flujo</b>	<i>Indicar el nombre del flujo</i>
<b>Descripción del Flujo</b>	
Paso <X> Descripción del paso	
<b>DESCRIPCIÓN DE EXTENSIONES</b>	
<b>Nombre del flujo</b>	<i>Indicar el nombre del flujo como la condición que lo genera</i>
<b>Descripción del Flujo</b>	
Paso <X> Descripción del paso	
<b>DESCRIPCIÓN DE SUBPROCESOS</b>	
<b>Nombre del flujo</b>	<i>Indicar el nombre del subproceso</i>
<b>Descripción del Flujo</b>	

## Diccionario de Datos

El Diccionario de Datos tiene como objetivo describir los atributos de cada entidad identificada en el Modelo Conceptual.

En la propuesta de RUP no aparece el artefacto Diccionario de Datos en el flujo de trabajo Requisitos; aunque durante el flujo de trabajo Modelación del Negocio se identifican las entidades de negocio y se describen brevemente sus atributos. En el Diccionario de Datos propuesto para el SIGEP por cada concepto del mundo real identificado en el Modelo Conceptual se genera una tabla que describe el nombre de cada uno de los atributos que lo identifican y de cada atributo, se define el tipo de dato, las restricciones y una breve descripción.

De cada entidad en el Diccionario de Datos se registran los siguientes elementos.

Nombre de la entidad		<i>El nombre de cada entidad debe ser único y descriptivo del concepto del mundo real que representa</i>			
Nombre del atributo	Descripción	Tipo	Puede ser nulo	Restricciones	
				Clase válidas	Clases no válidas
<i>Descriptivo del elemento del mundo real que representa y único para cada entidad</i>	<i>Descripción del dato</i>	<i>Tipo del dato en la realidad</i>	<i>Determina si el dato puede ser nulo</i>	<i>Valores que puede tomar el atributo</i>	<i>Valores que no puede tomar el atributo</i>

## Especificación de Casos de Uso

La Especificación de Casos de Uso es una narración del uso del sistema y se centra en la intención del actor. Tiene como objetivo describir los requisitos funcionales y producir un resultado de valor para un actor en particular.

A partir del Modelo de Negocio se identifican y describen los casos de uso del sistema. La Especificación de Casos de Uso es una descripción textual de la interacción entre el actor y el sistema.

Este artefacto aparece en la propuesta de RUP y constituye uno de los principios que guían todo el proceso de desarrollo. Sin embargo, a la plantilla propuesta por RUP se le proponen modificaciones para su utilización en SIGEP. Por ejemplo: se le adiciona el nombre del actor que inicia el caso de uso, las poscondiciones se dividen en mínimas y de éxito, y se especifica la acción que inicia el caso de uso. Además se asocia a la descripción del caso de uso el prototipo de interfaz gráfica de usuario, los formatos

de entrada y salida de los documentos que sean relevantes tenerlos en cuenta en el caso de uso. Y por último, se asocian el Modelo Conceptual, Diccionario de Datos y Asuntos Pendientes. (ADOLPH y BRAMBLE '01)

La plantilla para las Especificaciones de Casos de Uso propuesta se muestra a continuación.

<b>Objetivo</b>		<i>Es el objetivo que persigue el actor primario cuando interactúa con el caso de uso</i>
<b>Actores</b>		<i>Deben tener un nombre descriptivo, puede haber varios actores relacionados con cada caso de uso. El actor primario debe señalarse añadiendo después de él "(primario)"</i>
<b>Resumen</b>		<i>Breve descripción del caso de uso</i>
<b>Precondición</b>		<i>La precondición declara de qué debe estar seguro el sistema antes de autorizar el inicio del caso de uso. Esto no debe ser chequeado de nuevo mientras dura el caso de uso. Usualmente tener una precondición indica que otro caso de uso está o se ha ejecutado. Una precondición debe escribirse como una afirmación simple. Por ejemplo: El cliente ha sido validado.(COCKBURN '00)</i>
<b>Garantías</b>	<b>Mínimas</b>	<i>Es lo mínimo a lo que se debe comprometer el sistema con los interesados. Tienen mayor importancia cuando la meta principal no puede ser cumplida. Debe escribirse como una afirmación simple, que debe ser cierta al terminar el caso de uso.(COCKBURN '00)</i>
	<b>De éxito</b>	<i>Establece qué intereses deben ser satisfechos cuando se haya terminado exitosamente el caso de uso. Debe escribirse como una afirmación simple, que debe ser cierta al terminar el caso de uso.(COCKBURN '00)</i>
<b>Acción que inicia el caso de uso</b>		<i>Especifica qué evento debe ocurrir para que el sistema inicie el caso de uso</i>
<b>Flujo de eventos</b>		
<b>Flujo básico &lt;Nombre del flujo básico&gt;</b>		
Actor	Sistema	
1		
	2	
<b>Flujos alternativos &lt;Condición que dio lugar a la extensión&gt;</b>		
Actor	Sistema	
1		
	2	
<b>Relaciones</b>	<b>CU incluidos</b>	<i>Listado de los casos de uso y/o salidas del sistema incluidos por el caso de uso</i>
	<b>CU extensión</b>	<i>Listado de los casos de uso y/o salidas del sistema que extienden el caso de uso</i>
<b>Requisitos especiales</b>		<i>Son los requisitos no funcionales específicos para el caso de uso</i>
<b>Asuntos pendientes</b>		<i>Posibles mejoras al caso de uso. Elementos pendientes respecto al caso de uso</i>

**Prototipo elemental de interfaz gráfica de usuario:**

*Imagen que muestra el prototipo de interfaz de usuario.*

**Formatos de:**

*En esta sección se especificarán los formatos de entrada y salida de la información, según corresponda, si existe alguno establecido. Por ejemplo, si en la actualidad se utiliza alguna plantilla para introducir determinados datos, esta se inserta aquí.*

**Entrada**

*Incluir formatos de entrada.*

**Salida**

*Incluir formatos de salida.*

**Modelo conceptual**

*Según se explica en la descripción del artefacto Modelo Conceptual.*

**Diccionario de datos**

*Según se explica en la descripción del artefacto Diccionario de Datos.*

## **Especificación de Requisitos de Software**

La Especificación de Requisitos de Software es una colección de los requisitos del sistema representados de manera detallada, clara y concisa. Constituye el resultado de las actividades desarrolladas durante el flujo de trabajo de Requisitos.

Como se explicó anteriormente puede ser un documento, puede interpretarse como el Modelo de Casos de Uso o como un conjunto de artefactos. En el caso de SIGEP la Especificación de Requisitos de Software es el conjunto de los artefactos Modelo de Casos, Especificaciones Suplementarias, Prototipo de Interfaz Gráfica de Usuario y Glosario de Términos.

## **Especificaciones Suplementarias**

El documento Especificaciones Suplementarias contiene los requisitos no funcionales del sistema. Comprende los requisitos de calidad, organizados según su clasificación “FURPS+” (funcionalidad, facilidad de uso, fiabilidad, rendimiento y soporte) e incluye restricciones de software y hardware; restricciones de desarrollo, cuestiones de internacionalización, documentación y ayuda, licencia y otras cuestiones legales además de estándares y restricciones de diseño e implementación (RATIONAL SOFTWARE '03a).

Este artefacto aparece en la propuesta de RUP y su objetivo es registrar los elementos o restricciones con las cuales el negocio y el sistema deben cumplir.

## **Evaluación de la Organización**

La Evaluación de la Organización es un informe del resultado de la evaluación realizada por el Asesor de Transformación Organizacional a los procesos de la DGSP. Se elabora durante la fase de Inicio. Contiene el resultado de evaluar los procesos de negocio y valorarlos en satisfactorios, no satisfactorios o medianamente satisfactorios. La evaluación se desarrolla en base a aspectos definidos en un instrumento elaborado por el Especialista en Transformación Organizacional. Incluye un análisis detallado de todos los problemas detectados, y termina con la valoración de si el negocio está listo, o no, para continuar el flujo de trabajo Modelación del Negocio y comenzar el de Requisitos.

El artefacto Evaluación de la Organización aparece en la propuesta de RUP. Describe el estado actual de la organización en términos de procesos actuales, capacidades y actitudes de las personas, clientes, problemas y áreas de mejora; incluye además, una valoración objetiva del negocio. Este documento se

elaboró en SIGEP con el objetivo de valorar el estado de los procesos del Sistema Penitenciario Venezolano e identificar los que estaban listos para iniciar los flujos de trabajo Modelación de Negocio y Requisitos.

### **Glosario de Términos**

El Glosario de Términos es un documento que se desarrolla desde la fase de Inicio y es mantenido durante todo el ciclo de vida del proyecto. Contiene los términos nuevos, importantes o ambiguos y su definición. Los términos están organizados alfabéticamente y en ocasiones pueden formarse grupos de términos, los cuales se deben organizar según este mismo criterio. Todos los documentos que usan términos que están en el Glosario, deben hacer una referencia a este.

El artefacto Glosario de Términos aparece en la propuesta de RUP y su objetivo es definir los términos más importantes usados en la organización. Se utiliza como un diccionario informal de datos donde las definiciones capturadas sirven para entender otros documentos que se desarrollarán a lo largo del proyecto.

### **Matriz de Seguimiento a los Requisitos**

La Matriz de Seguimiento a los Requisitos es mantenida durante todo el ciclo de vida del proyecto, tiene como propósito mantener información sobre los requisitos, sus atributos y dependencias para ser usados en la evaluación del impacto y gestión de los cambios (RATIONAL SOFTWARE '03a).

Contiene los elementos de seguimiento de traza a los requisitos organizados por tipo y de cada tipo los atributos correspondientes. Cuando se identifica un elemento de seguimiento de traza este se incluye en la matriz y se completan los atributos correspondientes; en actividades posteriores estos atributos deben ser actualizados. Es recomendable mantener el seguimiento a los requisitos mediante una herramienta de software que permita su documentación bajo un esquema organizado y bien definido.

### **Modelo Conceptual**

El Modelo Conceptual es una representación de conceptos del mundo real, no de componentes de software. El objetivo de la creación de este artefacto es aumentar la comprensión del problema y contribuir a esclarecer la terminología o nomenclatura del dominio. Puede verse como un modelo que comunica a los interesados cuáles son los términos importantes y cómo se relacionan entre sí. (JAIME '06) Se representa mediante un diagrama de clases UML y contiene una descripción de los datos.

El artefacto Modelo Conceptual no aparece en RUP como parte del flujo de trabajo Requisitos, aunque en su propuesta está el artefacto entidad del negocio que representa una abstracción de elementos persistentes dentro del negocio. Este diagrama representa las entidades, los atributos que las describen y las relaciones que conectan las entidades.

### **Modelo de Casos de Uso**

El Modelo de Casos de Uso describe la funcionalidad esperada del software en términos de actores y casos de uso. Representa la organización de los casos de uso por sistemas, subsistemas y módulos y las relaciones entre ellos mediante un diagrama UML de casos de uso y contiene la Descripción de las Salidas del Sistema y las Especificaciones de Casos de Uso.

El Modelo de Casos de Uso aparece en la propuesta de RUP y para el SIGEP sólo se le añadió el artefacto Descripción de las Salidas del Sistema.

### **Modelo de Negocio**

El Modelo de Negocio es una representación de los procesos de la organización, las relaciones existentes entre ellos, sus metas, entradas, salidas y reglas de negocio que permiten tener una visión global del negocio. Contiene además la Descripción de Procesos de Negocio.

Este artefacto aparece en la propuesta de RUP. Para el SIGEP el Modelo de Negocio se construye a partir de la Descripción de Procesos de la Organización, el Mapa de Relaciones Internas y el Mapa de Relaciones Externas. A partir de la Descripción de Procesos de la Organización se identifican los procesos, reglas de negocio y metas; el Mapa de Relaciones Externas es útil para identificar las entradas y salidas de los procesos, al igual que el de Relaciones Internas, que es útil además en la identificación de relaciones en los procesos de la organización. La representación del Modelo de Negocio se hace a través de un diagrama de procesos de UML *Business*.

### **Plan de Requisitos**

Describe la estrategia a seguir durante la captura de requisitos, incluye las etapas, los entregables y los participantes. Además refleja el plan de entrevistas, de visitas y la matriz de cooperación donde se establecen las relaciones entre los participantes y sus responsabilidades. (ARIAS y OTROS '06).

El artefacto Plan de Requisitos aparece en RUP pero con el nombre de Plan de administración de Requisitos. Describe los artefactos que se generarán durante la etapa de captura de requisitos y los

mecanismos utilizados para definir, capturar y documentar los requisitos. En el SIGEP el Plan de Requisitos contiene además la manera en que se gestionan los cambios en cada uno de los documentos generados durante este flujo de trabajo.

### **Prototipo de IGU**

El Prototipo de IGU se utiliza para definir y validar los requisitos de software. Este prototipo en SIGEP es una aplicación *web* estática que contiene un grupo de pantallas relacionadas entre sí que no ejecutan ninguna acción, excepto permitir la navegabilidad entre una pantalla y otra. Su objetivo es validar el contenido informativo de las interacciones actor-sistema descritas en los flujos de los casos de uso.

El Prototipo de IGU es un artefacto que aparece en la propuesta de RUP durante el flujo de trabajo de Análisis y Diseño. En SIGEP fue utilizado durante el flujo de trabajo de Requisitos con el objetivo de clarificar requisitos ambiguos o que contaban con una baja definición y como resultado del mismo se obtiene un sistema navegable no funcional.

### **Visión**

El documento Visión se elabora en la fase de Inicio y se usa como base para la Modelación del Negocio. Describe el alcance de la solución y relaciona de modo general las áreas a informatizar y las funcionalidades del futuro sistema, define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.(ARIAS y OTROS '06).

Este artefacto aparece en la propuesta de RUP. La Visión define el conjunto de objetivos a los cuales apunta la modelación del negocio, contiene los principales requisitos previstos para el sistema y proporciona la base contractual para los requisitos técnicos más detallados.

## **2.5.2 Artefactos entregados al Equipo de Desarrollo**

El Equipo de Transformación Organizacional provee al Equipo de Desarrollo de un conjunto de documentos que permiten iniciar la Modelación de Negocio y la Ingeniería de Requisitos. Estos se describen a continuación:

### **Descripción de Procesos de la Organización**

La Descripción de Procesos de la Organización es un artefacto de entrada a la Modelación de Negocio. Contiene la descripción de los procesos globales, subprocesos y actividades que se realizan en la

organización. De los procesos globales detalla el nombre, responsable, misión, objetivos, clientes externos, clientes internos, alcance, productos, indicadores y los subprocesos que los integran. De los subprocesos define la misión, responsable, alcance, clientes internos, clientes externos, entradas, responsables de las entradas, salidas, responsables de las salidas, indicadores, y las actividades contenidas dentro del mismo. Y a su vez, de cada actividad se tiene la descripción, el responsable, la información que se maneja y los documentos de entrada y salida asociados a la misma.

La Descripción de Procesos de la Organización es el artefacto resultante de la transformación organizacional realizada en la DGSP por parte de un equipo multidisciplinario venezolano encargado de llevar a cabo el proceso de transformación dentro de la institución.

El artefacto Descripción de Procesos de la Organización no está en la propuesta de RUP, pero es de vital importancia su consulta porque describe el estado actual de la organización y contribuye a un mejor entendimiento del negocio por parte del equipo de software; todo ello tributa a que la modelación se realice conforme a como se desarrolla el negocio.

### **Diagnóstico de la DGSP**

Es una agrupación lógica de los documentos resultantes del diagnóstico realizado a la DGSP en el 2005, constituye una entrada a la Modelación de Negocio.

### **Documentos Resultantes de la Transformación Organizacional**

El artefacto Documentos Resultantes de la Transformación Organizacional es una agrupación lógica de los documentos producidos durante la transformación. Constituye una entrada a la Modelación de Negocio y agrupa el Informe Resultante de la Transformación Organizacional, la Descripción de Procesos de la Organización, el Mapa de Relaciones Internas, el Mapa de Relaciones Externas y el Resultado del Diagnóstico a la DGSP.

Este artefacto no aparece en la propuesta de RUP pero fue necesario definirlo en SIGEP debido a la necesidad de agrupar todos los artefactos generados por el Equipo de Transformación Organizacional durante la transformación de la DGSP.

### **Informe Resultante de la Transformación Organizacional**

Es un artefacto de entrada a la Modelación del Negocio contiene un informe de los resultados de la Transformación Organizacional: los problemas existentes, los problemas resueltos por la transformación, un resumen de los procesos nuevos y modificados, entre otros aspectos relevantes.

Este artefacto no está en la propuesta de RUP pero fue elaborado por el Equipo de Transformación Organizacional de la DGSP y consultado por el Asesor de Transformación Organizacional del equipo de software para realizar la evaluación de la organización.

### **Legislación Penitenciaria**

Es una agrupación lógica de todas las leyes, códigos, reglamentos y demás normativas que, estando vigentes, regulan el Sistema Penitenciario Venezolano. Constituye una entrada a la Modelación de Negocio.

Para el SIGEP fue necesario definir este artefacto debido a la necesidad de asociar cada proceso con las leyes y reglamentos que debe cumplir, además de que impone reglas de negocio al sistema.

### **Mapa de Relaciones Externas**

El Mapa de Relaciones Externas es una “representación gráfica o tabulada de las relaciones que se establecen entre la organización y los distintos organismos públicos, jurídicos e internacionales a través del envío y/o recepción de la documentación necesaria.” (MEJÍAS '06). Se elabora durante la transformación organizacional y se utiliza en la Modelación del Negocio.

Este artefacto no está en la propuesta de RUP pero fue definido en SIGEP debido a la necesidad de conocer el flujo de información existente entre la DGSP y otras instituciones para determinar cuáles tendrían un impacto en el sistema y sería necesario modelar.

### **Mapa de Relaciones Internas**

El Mapa de Relaciones Internas es una: “representación gráfica o tabulada de las relaciones que se establecen entre los distintos procesos de la organización determinadas por el flujo de información existente.”(MEJÍAS '06). Se elabora durante la transformación organizacional y se utiliza en la Modelación del Negocio.

Este artefacto no está en la propuesta de RUP pero en este caso se propone su utilización debido a la necesidad de conocer el flujo de información y documentación entre los procesos internos que integran la DGSP y así analizar cuales tendrían un impacto para el sistema y sería necesario modelar.

## **2.6 Roles**

A continuación se relacionan los roles propuestos en SIGEP para participar en los flujos de trabajo Modelación del Negocio y Requisitos. De cada rol se define el nombre, una breve descripción de las actividades que realiza y los conocimientos que debe tener.

### **Líder de Proyecto**

#### Descripción

- Participa en el Análisis del Problema.
- Participa en la definición de una metodología de desarrollo.
- Aprueba las tecnologías a utilizar en el desarrollo del proyecto.
- Es responsable de coordinar y organizar las tareas que se asignan a los miembros del equipo de desarrollo.
- Gestiona los recursos y materiales necesarios para el proyecto y para el equipo de desarrollo.

#### Conocimientos

- RUP.
- Tecnologías de desarrollo de software.
- Gestión de proyectos.
- Ingeniería de Software.
- Técnicas para la comunicación y negociación.

### **Analista**

#### Descripción

- Define una estrategia para llevar a cabo los flujos de trabajo Modelación de Negocio y Requisitos.

- Define los artefactos, actividades y las técnicas de recopilación de información que se utilizarán en la Modelación de negocio y en la Ingeniería de Requisitos.
- Supervisa y controla el cumplimiento del procedimiento para desarrollar los flujos de trabajo anteriormente mencionados.
- Define los sistemas, subsistemas y módulos en que se organiza la solución de software.
- Realiza las actividades correspondientes en los flujos de Modelación de Negocio y Requisitos.

#### Conocimientos

- Ingeniería de Requisitos.
- Modelación de procesos de negocio.
- RUP y UML.
- Herramientas de modelado visual.
- Técnicas de recopilación de información.
- Técnicas para la comunicación y negociación.

#### **Revisor Técnico**

##### Descripción

- Chequea que los artefactos generados se ajusten a las pautas y lineamientos establecidos para su confección.
- Elabora las listas de chequeo de los artefactos.
- Realiza las revisiones técnicas formales.

#### Conocimientos

- Calidad de Software.
- Ingeniería de Software.
- Elementos básicos sobre el negocio.
- RUP y UML.

## **Asesor Legal**

### Descripción

- Estudia las leyes y reglamentos vigentes en materia penitenciaria.
- Define la base legal necesaria para llevar a cabo el proyecto.
- Se integra al equipo de desarrollo antes y durante la captura de requisitos para garantizar que todo el trabajo se desarrolle respetando la legalidad penitenciaria.

### Conocimientos

- Jurídicos en materia penitenciaria.

## **Asesor de Transformación Organizacional**

### Descripción

- Analiza la situación existente en la organización.
- Define un instrumento para evaluar procesos.
- Aplica un instrumento para la identificación y evaluación de las áreas de la organización.
- Realiza una evaluación de las condiciones de cada una de los procesos de la organización.

### Conocimientos

- Análisis y diseño de procesos.
- Organización de procesos.

## **Conclusiones parciales**

En el presente capítulo se describió el procedimiento propuesto para desarrollar la Ingeniería de Requisitos en SIGEP. Fueron redefinidos los flujos de Modelación de Negocio y Requisitos propuestos por RUP. En el primero de ellos se definieron dos actividades, en el caso del flujo de trabajo Requisitos, de las seis actividades propuestas por RUP se mantuvieron tres, las cuales, al igual que las del flujo Modelación de Negocio incluyen nuevas tareas de validación, revisión y seguimiento a los requisitos.

El flujo de trabajo Modelación de Negocio de RUP plantea la elaboración de dieciséis artefactos, en el procedimiento propuesto se propone elaborar en este flujo seis artefactos, de ellos dos se mantienen según la plantilla propuesta por RUP, dos han sido redefinidos y dos se han introducido nuevos. En el flujo de trabajo Requisitos se han mantenido cinco de los doce artefactos propuestos por RUP, dos han sido redefinidos y cuatro son nuevas proposiciones en este flujo de trabajo.

Se definieron ocho elementos de seguimiento a los requisitos con sus respectivos atributos y relaciones, se redefinieron tres de los roles propuestos por RUP y se introdujeron dos completamente nuevos.

## CAPÍTULO III VALIDACIÓN DEL PROCEDIMIENTO

### Introducción

Hacer pronósticos mediante métodos matemáticos formales del comportamiento del procedimiento propuesto en el Capítulo II no era posible debido a la insuficiencia de datos al respecto que pudieran servir de punto de partida. En estos casos es útil el uso de métodos que permitan obtener información de los expertos en el tema que se está investigando. La validación del procedimiento propuesto en el Capítulo II se realizó a través de criterios de expertos mediante el uso de técnicas propuestas por el método *Delphi*. *Delphi* es uno de los métodos subjetivos de pronosticación más confiables, su origen se remonta a la década de los 60, fue creado por la *Rand Corporation* con el objetivo de elaborar pronósticos referentes a posibles acontecimientos en varias ramas de la ciencia, la técnica y la política. (CRITERIO DE EXPERTOS: MÉTODO DELPHY. '06)

“... el *Delphi* es la utilización sistemática del juicio intuitivo de un grupo de expertos para obtener un consenso de opiniones informadas” (CRITERIO DE EXPERTOS: MÉTODO DELPHY. '06)

Para la aplicación práctica del método es necesario considerar dos cuestiones fundamentales:

- La selección de los expertos
- La elaboración del cuestionario

A continuación se explica cómo fueron aplicadas estas cuestiones en el presente trabajo.

### 3.1 Selección del grupo de expertos

“Se entiende por experto, tanto al individuo en sí como a un grupo de personas u organizaciones capaces de ofrecer valoraciones conclusivas de un problema en cuestión y hacer recomendaciones respecto a sus momentos fundamentales con un máximo de competencia.” (CRITERIO DE EXPERTOS: MÉTODO DELPHY. '06)

Para la selección del grupo de expertos se realizaron las siguientes actividades:

- Determinación de las áreas del conocimiento que deben dominar los expertos.

Partiendo del problema planteado en la Introducción se determinó que los expertos a consultar debían dominar las siguientes áreas del conocimiento: procesos de desarrollo de software, ingeniería de software, modelado de procesos y estándares internacionales de calidad.

- Elaboración del listado de expertos candidatos.

Luego de determinar las áreas del conocimiento se elaboró un listado de expertos candidatos teniendo en cuenta su experiencia productiva o docente en las áreas identificadas. El listado inicial estaba conformado

por 10 expertos, distribuidos de la siguiente manera: 3 especialistas con varios años de experiencia como líderes de proyecto y en la gestión de requisitos de software, 2 analistas de sistemas en la actualidad, 3 especialistas que han realizado estudios sobre el tema y 2 miembros de CALISOFT que han participado en la evaluación de otros proyectos productivos en la UCI.

- Obtención del consentimiento de los expertos para participar.

El siguiente paso fue la obtención del consentimiento de los expertos para participar en la validación. De los 10 expertos, 7 estuvieron de acuerdo en participar.

- Determinación del coeficiente de conocimiento de los expertos.

Las características de los expertos influyen decisivamente en la confiabilidad de los resultados obtenidos. Estas características son: calificación técnica, capacidad de emitir una decisión al respecto, conocimientos específicos sobre el tema a evaluar, disposición a participar, entre otros. (FEBLES '03) (*CRITERIO DE EXPERTOS: MÉTODO DELPHY.* '06)

Para la selección de los expertos es útil emplear la valoración por competencias. Este método consiste en calcular el coeficiente de competencia (k) del experto a partir de la autovaloración del experto sobre su conocimiento o información sobre el tema (kc) y el coeficiente de argumentación o valoración (ka) mediante la siguiente ecuación (*CRITERIO DE EXPERTOS: MÉTODO DELPHY.* '06)

$$k = (kc + ka)/2$$

El código de interpretación de los coeficientes de competencias es como sigue:

Si  $0,8 < k < 1,0$       coeficiente de competencia alto.

Si  $0,5 < k < 0,8$       coeficiente de competencia medio

Si  $k < 0,5$               coeficiente de competencia bajo

Recomendándose incluir en el grupo a los de coeficiente de competencia alto y medio.

En el Anexo 2 aparece la encuesta aplicada a los expertos para determinar su coeficiente de competencias, los resultados se reflejan en la tabla a continuación.

Experto N°	kc	ka	k
1.	0.9	0.90	0.90
2.	0.9	1.00	0.95
3.	0.8	0.90	0.85
4.	0.8	0.80	0.80
5.	0.6	1.00	0.80

6.	0.6	0.90	0.75
7.	0.8	0.70	0.75

De los 7 expertos encuestados 5 poseen un coeficiente de competencia alto y 2 un coeficiente de competencia medio, por lo que se decidió que todos pueden ser incluidos en la evaluación de la propuesta. En el Anexo 4 se muestra una caracterización del grupo de expertos.

- Selección de los expertos.

Para la selección de los expertos es necesario determinar el número de expertos que debe tener el grupo. No existe una norma generalizada para determinar el número óptimo de expertos, pero hasta 7 expertos el error disminuye exponencialmente, después de 30, aunque el error disminuye lo hace de manera poco significativa y no compensa el incremento de costos y esfuerzo, por lo que se sugiere utilizar un número de expertos en el intervalo de 7 a 30.(CRISTÓBAL y GÓMEZ '00)

Dado que 7 es un número adecuado de expertos y que precisamente esta cantidad fueron los que estuvieron dispuestos a participar en la validación, alcanzando niveles de competencia altos y medios, se decidió que los 7 formaran parte del Comité de Expertos.

### **3.2 Validación de la propuesta**

Para validar el Procedimiento para desarrollar la Ingeniería de Requisitos en SIGEP se aplicó la encuesta mostrada en el Anexo 3, sus objetivos fueron:

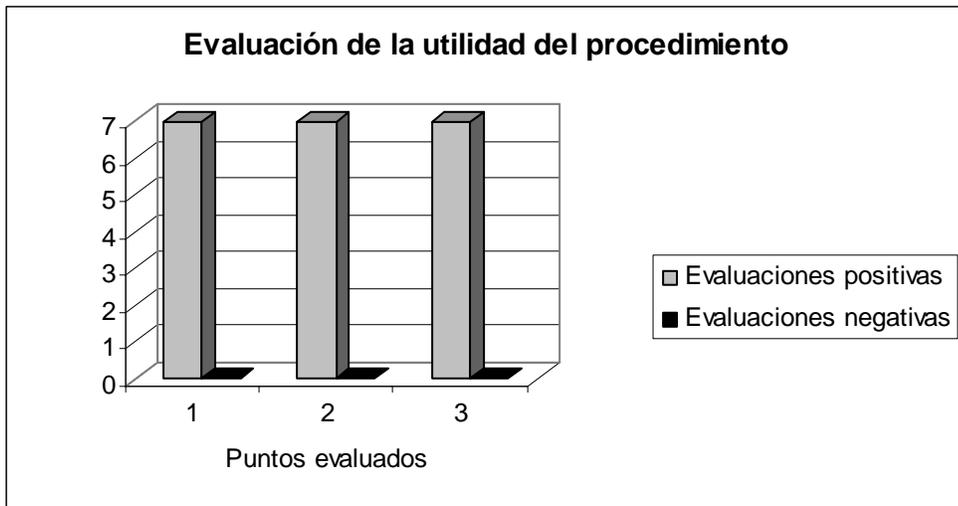
- Determinar la utilidad de la propuesta para resolver el problema planteado en la Introducción del presente trabajo. Para esto se formularon los puntos 1, 2 y 3.
- Determinar la efectividad de la propuesta mediante los criterios emitidos en los puntos 3 y 4.
- Determinar la completitud y corrección del procedimiento, mediante el punto 4.
- Identificar aspectos erróneos y recomendaciones que permitan mejorar la propuesta. Para ello en todos los puntos evaluados los expertos podían hacer las observaciones que estimasen necesarias y el punto 5 estaba dedicado a elaborar una valoración general al respecto.

A continuación se resumen los resultados de la aplicación de la encuesta en cuanto a:

#### Utilidad del procedimiento.

Para valorar este aspecto se tuvieron en cuenta las respuestas de los expertos a los puntos 1, 2 y 3, referidos a la importancia de desarrollar ERS para el éxito de los proyectos de software, a la necesidad de adaptar los procesos para desarrollar la Ingeniería de Requisitos según las características de cada proyecto y a la efectividad del procedimiento propuesto para elaborar y mantener una ERS en SIGEP,

respectivamente. Las respuestas a los puntos podían ser positivas o negativas y el resultado se muestra a continuación.

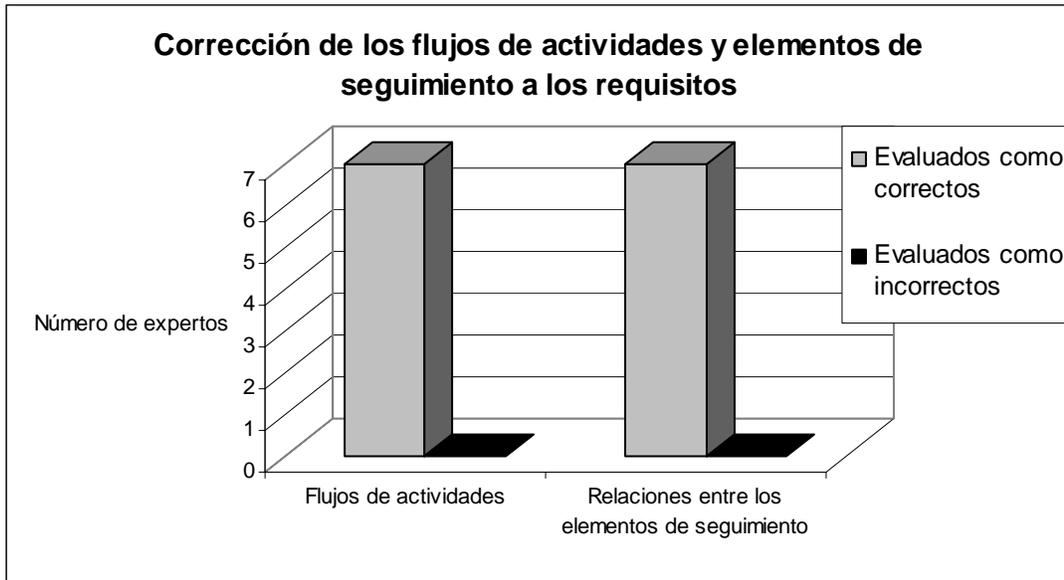


**Gráfico 1. Evaluación de la utilidad del procedimiento.**

#### Corrección y completitud del procedimiento

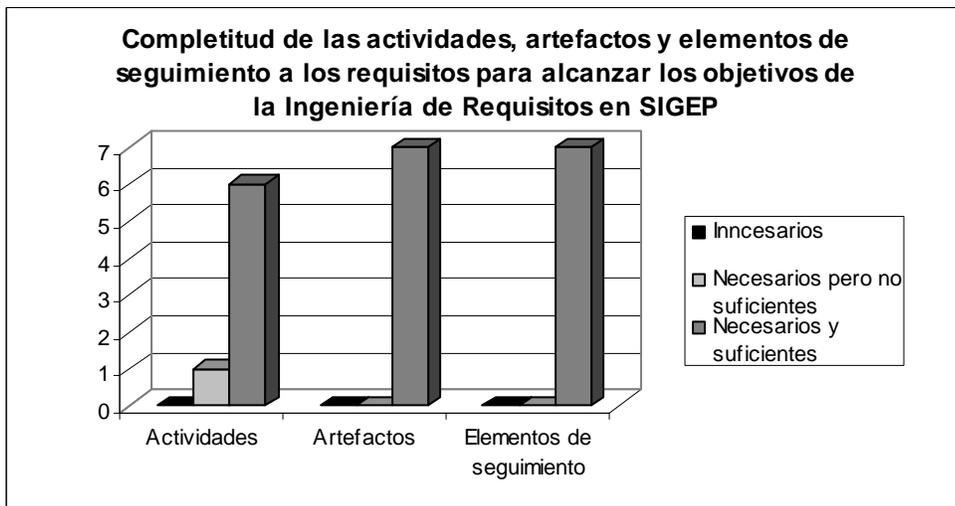
Para obtener una valoración al respecto se plantearon los puntos 3 y 4, este último con cinco incisos a evaluar referidos a la corrección y completitud de las actividades, los flujos de trabajo, los artefactos, los elementos de seguimiento a los requisitos y las relaciones de traza definidas entre ellos.

Las respuestas al punto 3 fueron mostradas anteriormente. En el punto 4 los incisos referidos a la organización de los flujos de trabajo y a las relaciones entre los elementos de seguimiento a los requisitos podían ser evaluados como correctos o incorrectos. El resultado se muestra en el siguiente gráfico.



**Gráfico 2. Corrección de los flujos de actividades y elementos de seguimiento a los requisitos.**

Los incisos del punto 4 referidos a la evaluación de las actividades, los artefactos propuestos y los elementos de seguimiento a los requisitos identificados podían recibir los siguientes valores: innecesarios, necesarios pero no suficientes y, necesarios y suficientes para alcanzar el objetivo propuesto en cada caso. A continuación se muestran los resultados obtenidos:



**Gráfico 3. Completitud de las actividades, artefactos y elementos de seguimiento a los requisitos para alcanzar los objetivos de la Ingeniería de Requisitos en SIGEP.**

## **Conclusiones parciales**

El análisis de los resultados anteriores permite afirmar que de manera general el procedimiento fue evaluado por los expertos como útil, correcto, completo y efectivo para desarrollar la Ingeniería de Requisitos en SIGEP.

En general, el Procedimiento para desarrollar la Ingeniería de Requisitos en SIGEP se describió por los expertos como un proceso que tiene en cuenta las características de los clientes y las condiciones objetivas de la organización a automatizar, enfocándose claramente en los puntos débiles de uno y otro, maduro, bien definido y factible de implementar en el proyecto. El modelado del negocio utilizando una notación orientada a procesos fue valorado por los expertos como muy positivo, así como la inclusión de revisiones técnicas formales y validaciones en varios momentos del proceso.

Las recomendaciones estuvieron alrededor de modificar el conjunto de artefactos propuesto de manera que se pueda generalizar el procedimiento en otros proyectos y realizar mediciones asociadas al tiempo, esfuerzo, errores detectados y corregidos en las actividades, para poder, transcurrido un tiempo, evaluar cuantitativamente la calidad y el desempeño del proceso propuesto y realizar las mejoras necesarias.

## CONCLUSIONES

1. Del análisis de la literatura disponible sobre la Ingeniería de Requisitos y del entorno en el cual se desarrolla SIGEP se determinó que para elaborar y mantener una ERS en este proyecto es necesario definir un procedimiento que se ajuste a las características del mismo, adecuando los flujos de trabajo Modelación de Negocio y Requisitos propuestos por RUP.
2. Partiendo del análisis de los factores que influyen en el desarrollo de la ERS en SIGEP, las buenas prácticas para elaborar ERS consultadas en la bibliografía y la experiencia personal de las autoras durante la primera etapa de captura de requisitos del proyecto se elaboró un procedimiento, el cual mantiene cinco de las doce actividades propuestas por RUP en los flujos de Modelación de Negocio y Requisitos. Estas actividades contemplan nuevas tareas de validación, revisión y seguimiento a los requisitos. De los veintiocho artefactos sugeridos por RUP siete se han mantenido, cuatro han sido redefinidos y seis se han introducido nuevos en estos flujos de trabajo. Se definieron ocho elementos de seguimiento a los requisitos, tres roles de los indicados por RUP fueron rediseñados y dos son nuevos.
3. El procedimiento fue valorado por los expertos consultados como útil, correcto, completo y efectivo para desarrollar la Ingeniería de Requisitos en SIGEP.

## **RECOMENDACIONES**

1. Emplear el procedimiento en la segunda etapa de captura de requisitos del proyecto SIGEP.
2. Definir un conjunto de métricas para evaluar la calidad y desempeño del procedimiento que sirvan como base para la mejora de este.
3. Definir un método para la estimación del esfuerzo necesario en la implementación de los requisitos de software a partir de los casos de uso, salidas del sistema, requisitos no funcionales y reglas de negocio.
4. Evaluar el uso de una herramienta automatizada para mantener el seguimiento a los requisitos de software.

## BIBLIOGRAFÍA

ADOLPH, S. y P. BRAMBLE. *Patterns for effective use case*, 2001. 1.

ARIAS, A. y OTROS. *Proyecto Técnico de Asesoría Especializada, Colaboración Médica Odontológica, Comunicación Institucional y Solución Tecnológica para apoyar la modernización del Sistema Penitenciario de la República Bolivariana de Venezuela*, UCI-DGCRR, 2006. 95.

CASTILLO, R. *Los tres problemas de los proyectos de software. Auditoría y Seguridad*, 2006. septiembre/octubre 2006.

COCKBURN, A. *Writing effective use case*. ADDISON-WESLEY, 2000.

*Concept: Traceability*. Rational Rose, 2003.

CRISTÓBAL, E. y J. GÓMEZ. *DESARROLLO DEL COMERCIO ELECTRÓNICO EN LA GESTIÓN EMPRESARIAL. Análisis de su situación en España*, 2000. [2007]. Disponible en: [http://www.mityc.es/NR/rdonlyres/F3DF486B-FE03-4F7B-A83E-61110AB1DD35/0/15\\_EduardFransi\\_357.pdf](http://www.mityc.es/NR/rdonlyres/F3DF486B-FE03-4F7B-A83E-61110AB1DD35/0/15_EduardFransi_357.pdf)

*CRITERIO DE EXPERTOS: MÉTODO DELPHY.*, 2006. 21 p.

ERIKSSON, H.-E. y M. PENKER. *Business Modeling with UML*. Jhon Wiley & Sons, Inc., 2000. p.

FEBLES, A.: CRIS, CUJAE, 2003. 136. p.

FERNÁNDEZ, G. *Extreme Programming*, Universidad de Castilla-La Mancha, 2002. [Disponible en:

GIL, M. *Presentación en el III Taller de Calidad en las Tecnologías de la Información y las Comunicaciones. Informática 2007*, 2007.

GONZÁLEZ, S. y L. FERNÁNDEZ *Programación Extrema: prácticas, aceptación y controversia. Cultura Científica y Tecnológica*, 2006, mayo-junio 2006.

HURTADO, J. A. y C. BASTARRICA. *Hacia una Línea de Procesos Ágiles Agile SPsL*. Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca, 2005. 140. p.

IEEE. *IEEE Guide for developing System Requirements Specifications*, IEEE, 1998a. 1233-1998: 36.

- . *IEEE Recommended Practice for Software Requirements Specifications*, IEEE, 1998b. 830-1998: 37.
- JACOBS, D. *Requirements Engineering So Things Don't Get Ugly*. *STSC CrossTalk*, 2004. Oct 2004.
- JACOBSON, I.; G. BOOCH, y otros. *El Proceso Unificado de Desarrollo de Software*. Empresa Poligráfica de Holguín, 2004. 435 p.
- JAIME, H. *Guía para desarrollar el Modelo Conceptual*, UCI, 2006. 4.
- LARMAN, C. *UML y patrones*, Prentice Hall, 2002. 520.
- LEISHMAN, T. y D. COOK. *Requirements risks can drown software projects*. *Cross Talk*, 2002. Abril 2002.
- MEJÍAS, S. *Instrumento para la identificación y evaluación de las áreas de la organización*, UCI, 2006. 7.
- MEJÍAS, S. y Y. VEGA. *Guía para desarrollar la Comprensión del Negocio*, Universidad de las Ciencias Informáticas, 2006. 5.
- MONTILVA, J. *Modelado de procesos de software*, 2006.
- MORDEZKI, M.; A. PETERSON, y otros. *PLAN ESTRATÉGICO PARA LA INTERNACIONALIZACIÓN DE LA INDUSTRIA DE TECNOLOGÍA DE LA INFORMACIÓN DE CHILE.*, GECHS: Sociedad Chilena de Software y Servicios A.G CORFO ProChile, 2006.
- ORANTES, S. *Calidad de Software en el uso de Metodologías Ágiles para el Desarrollo de Software*. III Taller de Calidad en las Tecnologías de la Información y las Comunicaciones. Informática 2007., La Habana, 2007. 25 p.
- PERALTA, A. *Metodología SCRUM*, Universidad Ort Uruguay, 2003. [Disponible en:
- PRESSMAN, R. S. *Ingeniería de software: Un enfoque práctico*. 5ta. La Habana, Editorial Félix Varela, 2005. p.

RATIONAL SOFTWARE, C. *Rational Unified Extended Help*, 2003a.

---. *Rational Unified Process: Overview*. HELP, R. U. E., 2003b.

RUIZ, A. *Mesa redonda Desarrollo y Perspectivas del desarrollo de la Industria de Software en Cuba e Iberoamérica*. *Informática 2007*, 2007.

SPENCE, I. y L. PROBASCO. *Traceability Strategies for Managing Requirements with Use Cases*, Rational Software 1998.

TOLIS, C. y A. NILSSON. Using business models in process orientation. en.p.

VILLARROEL, R.; Y. GÓMEZ, y otros. *Evaluación de procesos en una empresa desarrolladora de software usando MoProsoft*. *Informática 2006*. La Habana, Cuba, 2006.

YOUNG, R. Recommended Requirements Gathering Practices *CrossTalk*, 2002.

## ANEXOS

### Anexo 1. Aspectos a considerar para la Evaluación de la Organización. (MEJÍAS '06)

No	ITEMS	No existe	Si existe			Recomendaciones
			Satisfactorio	Medianamente satisfactorio	No satisfactorio	
1	Organigrama.					
2	Plantilla de Trabajadores.					
3	Funciones por unidades organizativas.					
4	Identificación de los procesos.					
5	Clasificación de los procesos en procesos claves, procesos de gestión y procesos de apoyo.					
6	Mapa de relaciones externas de los procesos con las instituciones jurídicas, de seguridad y organismos internacionales.					
7	Mapa de relaciones internas entre los procesos.					
8	Identificación de los productos (servicios) que van a satisfacer las necesidades de los usuarios.					
9	Manual de Calidad que cumpla con el					

	grupo de normas ISO 9000 y contemple la fichas de los procesos e instrucciones en la Gestión de la Calidad.					
10	Fichas e instrucciones de los procesos restantes.					
11	Sistema de Gestión de los Recursos Humanos por Competencias.					
12	Reglamento Interno General de la Institución.					
13	Manual de perfiles de cargo de los puestos de trabajo con sus competencias.					
14	Sistema de Información de cada proceso					
15	Participación de los especialistas en la definición de sus procesos.					

## Anexo 2. Encuesta para determinar el coeficiente de competencia de los expertos

1. Marque con una X, en una escala del 1 al 10 el valor que se corresponde con el grado de conocimiento o información que usted considera que tiene respecto a la Ingeniería de Requisitos. 1 indica que no tiene ningún conocimiento sobre el tema y 10 indica que tiene pleno conocimiento sobre él.

<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>

2. Señale con una X el nivel de influencia que ha tenido cada una de las fuentes indicadas en su conocimiento sobre la Ingeniería de Requisitos.

FUENTES DE ARGUMENTACION	Grado de influencia de cada una de las fuentes en sus criterios.		
	A (alto)	M (medio)	B (bajo)
Análisis teóricos realizados por usted			
Su experiencia obtenida			
Trabajos de autores nacionales			
Trabajos de autores extranjeros			
Su propio conocimiento del estado del problema en el extranjero			
Su intuición			

**Anexo 3. Evaluación del Procedimiento para desarrollar la Ingeniería de Requisitos en el proyecto Sistema de Gestión Penitenciaria (SIGEP)**

1. ¿Considera usted que es importante para el éxito de los proyectos de software mantener una Especificación de Requisitos de Software?

Sí\_\_\_ No\_\_\_

¿Por qué? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

2. ¿Considera usted necesario que cada proyecto adecue los procesos definidos para desarrollar la Ingeniería de Requisitos a sus características?

Sí\_\_\_ No\_\_\_

¿Por qué? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

3. ¿Considera usted que la aplicación del procedimiento propuesto puede ser efectivo en la elaboración y mantenimiento de una Especificación de Requisitos de Software en el proyecto SIGEP?

Sí\_\_\_ No\_\_\_

¿Por qué? \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

4. Evalúe el procedimiento propuesto según los siguientes aspectos.

a. Las actividades propuestas son:

Innecesarias

Necesarias pero no suficientes para desarrollar la Ingeniería de Requisitos

Necesarias y suficientes para desarrollar la Ingeniería de Requisitos

Otras consideraciones al respecto:

---

---

---

---

b. La organización de las actividades en flujos de trabajo es:

Correcta

Incorrecta

c. Los artefactos propuestos son:

Innecesarios

Necesarios pero no suficientes para desarrollar la Ingeniería de Requisitos

Necesarios y suficientes para desarrollar la Ingeniería de Requisitos

Otras consideraciones al respecto:

---

---

---

---

d. Para seguir la traza de los requisitos de software los elementos definidos son:

Innecesarios

Necesarios pero no suficientes

Necesarios y suficientes

Otras consideraciones al respecto:

---

---

---

---

e. Las relaciones establecidas entre los elementos de seguimiento a los requisitos son:

Correctas



#### Anexo 4. Caracterización de los expertos

Experto	Graduado de	Categoría Científica			Eventos en los que ha participado	Experiencia
		Categoría	Aspirante a	Tema		
Experto 1	Ing. Informática		Msc en Bioinformática	Modelo de Referencia para la Ingeniería de Requisitos en proyectos de Bioinformática	Convención Informática 2007	Participación en el proyecto LIMS de calidad de la facultad 6. Asesora de Ingeniería de Requisitos en la segunda fase del proyecto Registros y Notarías
Experto 2	Master en Ciencias Técnicas en el Instituto Politécnico de Kiev	Msc Ejecutivo en Tecnologías de la Información	Doctor en Ciencias	Modelado de Negocios y Gestión de Requisitos	Convención Informática 2007 SEIG 2006 Evento Europeo de Calidad de Empresas Informáticas del Sector de la Energética	Consultor en Procesos en áreas de procesos de informática y telecomunicaciones. Experto en Ingeniería de Requisitos de la Comisión Nacional de Calidad del Software de Cuba Responsable de la redacción de la Norma Cubana de Ingeniería de Requisitos por el Comité de Normalización del MIC. Director del Centro de Desarrollo Integrado de Proyectos Informáticos de la Empresa Correos de Cuba. Autor de la metodología ESTAFETA, para la modelación de procesos de

						negocio y requisitos, utilizada en DESOFT, SOFTEL, Correos de Cuba y en el Proyecto de Informatización de la Empresa Correos de Venezuela.
Experto 3	Ing. Informática		Msc. en Informática Aplicada	Metodología para implantar un proceso de medición de software en la UCI	Convención Informática 2007 Conferencia del SEPLA UCIENCIA	Fue líder del proyecto Gemelos, de la facultad 6
Experto 4	Ing. Informática		Msc. en Gestión de proyectos	Gestión de requisitos a través de procesos	UCIENCIA Convención Informática 2007	Un año de trabajo en CALISOFT desarrollando pruebas de aceptación con los clientes, donde se han validado los requisitos de software.
Experto 5	Ing. Informática		Msc. en Informática Aplicada			Participó en la Captura de requisitos del proyecto Centro de Tratamiento de Análisis de Información de Seguridad Ciudadana (CTAISC) como Analista principal. Participó en la elaboración de la estrategia a seguir para desarrollar la Captura de requisitos en CTAISC. Recibió el curso de postgrado Sistemas de

						Calidad.
Experto 6	Ing. Informática		Msc. en Dirección	Mejora de Procesos de Organizaciones desarrolladoras de software		Tutora de tesis relacionada con la elaboración de un procedimiento para desarrollar la Captura de Requisitos en Registros y Notarías (RN)
Experto 7	Ing. en Sistema Automatizado de Dirección	Msc. en Informática Aplicada			UCIENCIA Convención Informática 2007 Conferencia del SEPGLA	Imparte docencia en la Universidad desde 1992 en temas relacionados con Ingeniería de Software. Ha desarrollado trabajos con universidades extranjeras en Brasil, Bolivia y Canadá. Certificado del curso de Introducción que imparte el SEI. Miembro de la Comisión Nacional de Expertos en Calidad del Software de Cuba.