



Universidad de las Ciencias Informáticas

Facultad 3 "Turismo y Negocios"

**Título: Ingeniería de Requisitos para el desarrollo del Sistema de
Gestión de Inventario Almacén (SIGIA).
Módulo Nomencladores.**

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

Autoras: Dayana Caridad Tejera Hernández
Leidy Bárbara Sánchez Echevarría

Tutor: Arnaldo Utria Azán
Co-tutora: MSc. Leidy Fernández Sánchez

June de 2007

“Pensar es fácil; actuar, difícil; transformar los pensamientos en actos es lo más difícil.”

Johann W. Goeth.

DECLARACIÓN DE AUTORÍA

Declaro que somos las únicas autoras de este trabajo y autorizamos a la Facultad 3 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 __2007__.

Dayana Caridad Tejera Hernández

Leidy Bárbara Sánchez Echevarría

(Autor)

(Autor)

Arnaldo Utria Azán

(Tutor)

DATOS DE CONTACTO

Curriculum de la Co-tutora.

Leidy Agustina Fernández Sánchez con Nivel de escolaridad: universitario. Máster en Ciencias de la Computación en la especialidad de Automatización de Procesos Termoenergéticos de CEN.

Eventos en los que ha participado

- » Evento de la Mujer Creadora del 2004 (Premio).
- » FORUM Provincial de ETECSA 2006 (Premio relevante)
- » Conferencia Internacional de Ciencias Empresariales. UCLV. Octubre 2006 (Con publicación)
- » IV Convención de Calidad de ETECSA (Con publicación)
- » XII Conferencia y Feria Internacional Informática 2007 (Delegada/Ponente) (Con publicación)

Publicaciones

- » Procedimiento para el desarrollo del proceso de ingeniería de requisitos en un proyecto software (PROCIR). Anuario 2006 Universidad de Cienfuegos y CD IV Convención de calidad de ETECSA. III Taller de Calidad en las TICs. Informática 2007.
- » Gestión del riesgo en la fase de ingeniería de requisitos de un proyecto software. INTERNET, sitio Monografías.com. Diciembre 2006.
- » Gestión de los requisitos de un proyecto software. INTERNET, sitio Monografías.com. Diciembre 2006.
- » Agentes inteligentes para el diagnóstico de patologías ginecológicas. Anuario 2006 Universidad de Cienfuegos y CD IV Convención de calidad de ETECSA.
- » Modelo integral para la gestión de proyectos de SW en la PYME. III Taller de Calidad en las TICs. Informática 2007.

Investigación terminada y logro 2006

Procedimiento para el desarrollo del proceso de ingeniería de requisitos en un proyecto software (PROCIR) Facultad de Ciencias Empresariales de la UCLV 2006.

AGRADECIMIENTOS

A la Revolución y el Comandante por permitirnos esta oportunidad.

A nuestros padres: sin ustedes nada de esto fuera posible, muchos besos.

A Arnaldo nuestro tutor: sencillamente gracias por tu tiempo y tus consejos.

A mi hermano Javier (Leidy): A pesar de fajarnos mucho, te adoro, y ojalá en tu tesis haya un pedacito dedicado a mí. Esto triunfo te lo dedico a tí y además te agradezco por tu inocencia y por tenerme como un ideal. Vaya!! la próxima tesis es la tuya, yo no tengo la culpa de ser la mayor.

A Yohan y Fray: en especial a ustedes por el cariño, el aliento, el tiempo que no pudimos estar con ustedes, por querernos tanto (respectivamente).

A nuestra Cotutora Leidy: por tu dedicación, por la revisión y orientación de esta investigación, por darnos tu apoyo desinteresado, incluso sin conocernos, en fin gracias por tu preciado tiempo y estar siempre ahí.

A Bustio: gracias por todo, gracias por alentarnos y confiar en que sí podíamos, por dedicarnos un pedacito de tu tiempo.

A Keidy, Maikel y Janet por el apoyo, por las revisiones, por las aclaraciones, por orientarnos, muchas gracias a todos.

A todas nuestras amistades: Yudaika (yray), Miguel, Dani, en fin son muchos, gracias por soportarnos.

DEDICATORIA

Muchos dedican a sus padres, incluso a su memoria... yo dedico mis esfuerzos:

A mis dos madres...

Mima, a pesar de no estar aquí presente... Sé que estuvieras tan orgullosa como yo, tu recuerdo es mi impulso...

Mami, tu estas y estarás siempre conmigo... Tu confianza en mí es incluso más fuerte que la mía, eres mi luz.

Leidy

RESUMEN Y PALABRAS CLAVES

La evolución gradual y ascendente del software a nivel mundial, desde sus inicios hasta la actualidad, ha provocado que se perfeccionen cada día las técnicas referentes a su manejo, a lo largo de todo su ciclo desarrollo.

Administrar con equidad y eficacia los requisitos así como el control de sus cambios durante el ciclo de vida de un proyecto de desarrollo de software es una ardua tarea para los ingenieros. Actualmente una de las causas de la disolución de proyectos de software es la baja efectividad en el manejo de sus requisitos. La Ingeniería de Requisitos (IR) es la disciplina de Ingeniería del Software dedicada a mejorar la captura, especificación, validación y la gestión de requisitos del software. Para esto, la IR estudia y define métodos, notaciones y herramientas para garantizar una adecuada definición de los requisitos. La correcta especificación de requisitos determina en gran medida el éxito de un proyecto de desarrollo de software.

El uso de la tecnología en función de la automatización de los procesos en las empresas es de vital importancia para el aumento de su productividad. La Gestión de Inventario de Almacén (GIA) en Cuba se suma a tales avances para el tratamiento de sus procesos. La presente investigación pretende encaminar la IR en función de unificar los criterios, entre clientes y desarrolladores, referentes al tratamiento de la GIA almacén para desarrollar un software que viabilice la ejecución de los procesos de la GIA en el país.

Palabras claves: Ingeniería de Requisitos, elicitación de requisitos, sistema de gestión de inventarios.

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA	7
1.1 Introducción	7
1.2 IS: punto de partida para el análisis de la IR.	7
1.2.1 Ingeniería de Requisitos.....	10
1.2.1.1 Elicitación de Requisitos (ERE)	14
1.2.1.2 Análisis y Negociación de Requisitos (ANR).	17
1.2.1.3 Especificación de Requisitos (EPR)	18
1.2.1.4 Modelado del sistema (MSI).....	18
1.2.1.5 Validación de requisitos (VRE).....	19
1.3 Influencia de una deficiente IR en el desarrollo de un proyecto	19
1.4 Tendencias Actuales de la IR.....	20
1.4.1 Reutilización de requisitos.....	20
1.4.2 Patrones de requisitos (PR)	20
1.5 Lenguajes de Modelado	22
1.6 Metodologías de desarrollo de software	23
1.6.1 Metodología Tradicional: Proceso Unificado.....	24
1.6.2 Rol Analista que propone la metodología RUP.....	27
1.6.2.1 Rol de Analista de Sistemas (AS) que propone la metodología RUP.....	27
1.6.2.2 Rol de Analista de Procesos del Negocio (APN).....	29
1.7 Ingeniería de Software Asistida por Computadora (CASE)	29
1.7.1 Herramientas CASE para el manejo de requisitos.....	29
1.7.1.1 RequisitePro.....	30
1.7.1.2 IRqA.....	31
1.7.2 Herramienta CASE para el desarrollo de software.....	31
1.7.2.1 Rational Rose Enterprise.	32
1.8 Procesos de GIA. Softwares de Gestión cubanos.	33
1.8.1 Almacenes	33
1.8.1.1 Tipos de almacén	34
1.8.2 Gestión de Inventario de Almacén	35
1.8.3 Descripción de algunos Sistemas de Gestión de Inventario de Almacén en Cuba... ..	36
1.8.3.1 Sistema Interhotel (IHMM).....	37
1.8.3.2 Suite ZUN.....	37
1.8.3.3 DRIM (Tecnomática).....	38

1.9 Conclusiones Parciales	38
CAPÍTULO II. DESCRIPCIÓN DE LA SOLUCIÓN PARA EL MÓDULO NOMENCLADORES DE SIGIA	40
2.1 Introducción	40
2.2 Elicitación de Requisitos	40
2.2.1 Descripción de las técnicas utilizadas para el caso de estudio	40
2.2.2 Modelación del Negocio para los procesos de GIA	41
2.2.2.1 Definición de casos de uso y actores del negocio	42
2.2.2.2 Trabajadores y entidades del negocio	44
2.2.2.3 Descripción de los casos de uso del negocio	46
2.2.2.4 Reglas del negocio	49
2.3 Conclusiones Parciales	54
2.4 Especificación, Análisis y Negociación de requisitos para el módulo Nomencladores de SIGIA	55
2.4.1 Requisitos Funcionales	55
2.4.2 Requisitos no funcionales.	58
2.5 Modelado del sistema	61
2.5.1 Patrones de Casos de Uso (PCU) promoviendo la reusabilidad de la experiencia..	62
2.5.2 Descripción de los casos de uso del módulo Nomencladores para el SIGIA	64
2.6 Evaluación de los resultados. Validación de Requisitos de Software	88
2.6.1 Resultados de las Listas de Chequeo para Modelo del Negocio.	88
2.6.2 Resultados de las Listas de Chequeo para Modelado del Sistema.	89
2.6.3 Métricas de aseguramiento de la calidad para levantamiento de requisitos.	89
2.6.4 Validación de Requisitos de Software	95
Carta de Conformidad del cliente con los requisitos.	96
2.7 Conclusiones Parciales.	97
CONCLUSIONES GENERALES	98
RECOMENDACIONES.....	100
REFERENCIAS BIBLIOGRÁFICAS	101
GLOSARIO DE TÉRMINOS	104
ANEXOS.....	105
Anexo I Fases e Hitos (Milestones).....	105
Anexo II Estructura de RUP	105
Anexo III Diagramas de Actividad de los Casos de Uso del Negocio.....	105

INTRODUCCIÓN

El hombre en su desarrollo científico ha incorporado las tecnologías de la informática y las comunicaciones a su vida diaria formando parte indisoluble de la misma. La revolución tecnológica es sorprendentemente innovadora y de constante dinámica evolutiva, debido a la naturaleza creadora del ser humano como fiel exponente dentro del proceso tecnológico, creciente y cambiante de las últimas décadas, en el que se encuentra inmerso.

La demanda social frente a la producción de software asciende a un ritmo acelerado; construir productos que respondan a los requerimientos, la planificación y el presupuesto establecido es tema que implica un proceso de construcción de software organizado. Además, las exigencias con respecto al desarrollo de software de calidad crece en la medida que el tamaño y complejidad de los sistemas es mayor; así como su vinculación con dispositivos de hardware u otros componentes electrónicos. Los sistemas toman mayor connotación arquitectónica y los requerimientos son más elevados en correspondencia a los nuevos avances en el mercado industrial.

El software se ha convertido en el elemento clave de la evolución de los sistemas y productos informáticos. En los pasados 50 años el software ha pasado de ser una resolución de problemas especializada y una herramienta de análisis de información, a ser una industria por sí misma [Pressman05]. La conveniente práctica de la Ingeniería de Software (IS) sustenta la idea de obtener un proceso de construcción de software que descansa sobre una organización de calidad, sistemático y que abarque todo el ciclo de vida del desarrollo de software. *Esto evitaría los errores de un proceso de desarrollo inmaduro, centrado en la etapa de implementación y no en todo el ciclo de vida, además de un software difícil de mantener, ya sea por su inaccesibilidad o por su alto costo* [Fernández06].

El hombre cuando se enfrenta a cualquier problema de la vida diaria primero analiza su naturaleza y el alcance del mismo, así como su influencia a largo o corto plazo en los diferentes momentos de su vida. Luego de fijar el problema en sí, lo divide en partes para resolverlo y poder verificar el cumplimiento de cada una. La construcción de sistemas de software, como la resolución de problemas en la vida diaria, tiene como primer paso determinar lo que se va a desarrollar. Nada influye tanto negativamente en el desarrollo de software como la identificación tardía de algún proceso a construir que no fue detectado en la etapa inicial.

En la actualidad persisten problemas en el desarrollo de software, entre ellos, un inadecuado entendimiento de las necesidades de los usuarios, incapacidad de absorber cambios en los requerimientos e insatisfacciones de los clientes por inaceptable o bajo desempeño del software. Las principales causas de estos son la administración de requerimientos insuficiente, la comunicación ambigua e imprecisa, inconsistencias no detectadas entre requerimientos, diseño y programación, validaciones tardías de los requisitos, enfrentamiento tardío de riesgos y propagación de cambios sin control [Fernández06]. Precisamente, sobre la base de lo planteado, y de acuerdo a estudios realizados se determinó que *los principales factores de fracaso de un proyecto son: falta de información por parte de los usuarios, especificaciones y requisitos incompletos y especificaciones y requisitos cambiantes* [Fernández06]. A raíz de todos estos problemas, se requiere cada vez más perfeccionar el proceso de IR enfocando sus procedimientos y técnicas a la resolución y gestión de los requisitos, de forma sistémica a lo largo del ciclo de vida del desarrollo de software.

En la actualidad la economía de todas las naciones desarrolladas depende del software [Sommerville04]¹. La gestión empresarial y administrativa no se encuentra exenta al mencionado proceso evolutivo del software, lo cual estimula el uso de las mismas en la automatización y modernización de sus procesos y actividades. Todo esto con el simple objetivo de controlar y perfeccionar mediante la tecnología, el manejo de sus recursos y materiales. Sin olvidar además, la marcada importancia de los métodos tradicionales y de organización en los que se basa la gestión empresarial y de mercado; con las cuales se obtienen aún notables resultados y que las nuevas tecnologías vienen a soportar.

Todo lo anterior pone al corriente de la inminente tendencia a la utilización de los recursos informáticos en el desarrollo y modernización de las empresas y la madurez de los procesos de desarrollo de software, en función de la calidad y con un enfoque de Ingeniería².

Las empresas en Cuba se encuentran inmersas dentro de tales avances tecnológicos aunque no en su totalidad. *Las empresas cubanas del software deben que realizar profundas transformaciones, que no es solo el uso de la tecnología más novedosa, sino la ejecución de tareas relativas a la gestión de la calidad,*

¹ Ian Sommerville. Autor, académico y consultor. Actualmente Profesor de la Universidad de Informática de St Andrews, Escocia. Durante 20 años (1986-2006) profesor en el Departamento de Computación de la Universidad de Lancaster (Reino Unido). Especialista en gestión de requisitos y la ingeniería del software en general. Autor Libro "Software Engineering" publicado en 1982 y ya actualmente en su 8va Edición. <http://www.cs.st-andrews.ac.uk/%7Eifs/>

² Enfoque de ingeniería que implica organización disciplinada en la construcción de software.

al nivel de la organización y también al nivel de los proyectos, como parte del perfeccionamiento empresarial [Fernández06].

La GIA es una actividad importante en cualquier organización, permitiendo de esta forma el control de sus medios disponibles. La GIA en sí, abarca todo lo relativo al control y manejo de las existencias de determinados bienes en la cual se aplican métodos y estrategias que pueden hacer más rentable y productivo la tenencia de estos bienes. A su vez permite evaluar los procedimientos de entradas y salidas de dichos productos. El uso de las nuevas tecnologías, en función del manejo de los procesos de GIA en el país merita atención, en pos de gestionarlos y controlarlos eficazmente.

A pesar de existir sistemas informáticos para la GIA en el país, ya sean estos de fabricación nacional o extranjera, centralizan su funcionamiento en un sector determinado; no respondiendo de esta forma a las peculiaridades de las restantes entidades existentes. En algunas empresas, al no contar con el presupuesto necesario para la adquisición de estos sistemas en el exterior adoptan los métodos tradicionales para la GIA. Unido a esto el Ministerio de Finanzas y Precios (MFP) en coordinación con Ministerio de Economía y Planificación (MEP) dictaron legislaciones que avalaban la adopción en el país de una doble moneda; situación que provoca que algunos sistemas de GIA existentes no contemplen los nuevos cambios ocurridos. Los productos por tanto tendrán precios en diferentes monedas, las cuáles no siempre responden a la moneda base de la contabilidad.

La variabilidad en la adopción de uno u otro método (tradicional o informatizado) para el desarrollo de los procesos GIA en el país, desencadena en una serie de aspectos negativos que influyen en su estabilidad económica. El control de los recursos disponibles y de su uso adecuado es fundamental tanto para la economía como para el bienestar del cliente.

La necesidad de vincular los temas de producción de software enfocados a la automatización y mejora de los procesos de GIA en el país provocan la siguiente ***situación problema:***

- El logro de un entendimiento común, de los procesos del negocio de GIA, entre clientes y desarrolladores.
- La captura con exactitud de las necesidades del cliente para finalmente transformarlas en un Sistema de Gestión de Inventarios y Almacén (SIGIA) que responda a todos los procesos del negocio.
- La inclusión de las nuevas legislaciones establecidas por los organismos correspondientes.

- La parametrización de las funcionalidades respecto al tratamiento de los procesos de GIA, de forma tal que se ajuste a las empresas del país.

Se pretende así captar las expectativas actuales, en virtud de minimizar los problemas existentes. Todo enfocado a la captura, con la mayor precisión, de las exigencias del beneficiado y transformarlas así en un producto con la máxima calidad y confiabilidad que el cliente espera. Por último fomentar el análisis del sistema, mediante el uso de una metodología de desarrollo de software, por la capacidad que posee la misma de guiar los esfuerzos y las personas a desarrollar sistemas con la calidad requerida.

En este contexto se impone además que el sistema funcione de forma conjunta con otro que controla la contabilidad. Resultados que serían un tanto difíciles de alcanzar sin una adecuada IS que controle todo el ciclo de vida del producto, y en sus inicios la necesidad de la puesta en práctica de una correcta estrategia de captura de requisitos, mediante las entrevistas y constantes intercambios con los clientes para el entendimiento de los procesos del negocio y la eliminación de posibles errores a la hora de implementar el modelado propuesto.

La Universidad de Ciencias Informáticas (UCI) es convocada para satisfacer las demandas existentes, mediante el desarrollo de los temas más actuales de construcción de software de calidad, en función de la GIA. En el país existen empresas productoras de software involucradas en el proceso de obtención de software de calidad y todos los temas actuales de ingeniería y gestión de software, como el Grupo Electrónica para el Turismo (GET), Empresa Nacional de Desarrollo de Software (DESOFT), Empresa de Software, Desarrollo y Comercialización (SOFTTEL), etc. La UCI se une a estas, contando con una infraestructura productiva en la cual involucra a estudiantes y profesores en estos procesos de producción de software para Cuba y el exterior. Todas estas empresas y las universidades del país poseen un objetivo común: encaminar el desarrollo de las tecnologías y su máximo uso en todas las esferas del país.

Ante la evidente situación con la GIA se plantea el siguiente **problema**: ¿Cómo lograr una visión común entre clientes y desarrolladores de los requisitos de software para el módulo Nomencladores del SIGIA? En vistas a resolver el problema planteado el **objeto de estudio** de la presente investigación es: el proceso de ingeniería de requisitos. Enmarcando como **campo de acción del objeto de estudio** a: las actividades de elicitación, análisis y negociación, especificación de requisitos, modelado del sistema y validación de requisitos aplicadas a los procesos de GIA, en Cuba. Con el propósito de proceder al cumplimiento del problema planteado se precisa como **Objetivo principal**:

- Realizar la captura de requisitos de los procesos de GIA para el módulo Nomencladores del SIGIA.

Para el viable progreso de la investigación y el acatamiento del objetivo presentado se establecen las siguientes **tareas**:

- Revisar y seleccionar la bibliografía acerca del tema de IR para la comprensión y desarrollo exitoso de la investigación.
- Construir la fundamentación teórica y destacando la importancia de la IR y su vinculación para la construcción de software en función de la automatización de los procesos de GIA.
- Caracterizar la metodología de desarrollo de software y la herramienta utilizada.
- Identificar y definir los procesos de negocio para la GIA en Cuba.
- Determinar y fundamentar la/las técnica(s) de identificación o elicitación de requisitos a utilizar.
- Realizar el Modelo del Negocio para la GIA.
- Especificar los requisitos de software para el módulo Nomencladores de SIGIA.
- Estructurar el Modelo del Sistema, para el módulo Nomencladores, en función de los requisitos obtenidos.

Se ha determinado además que se utilizarán las estrategias y métodos de investigación siguientes:

- Exploratoria: teniendo en cuenta que la problemática de la investigación, principalmente lo referente a los procesos de la GIA, está basada en un fenómeno que ocurre sobre una determinada muestra de la población y no están claras las ideas acerca del tema.
- Descriptiva: con el objetivo de describir y destacar lo esencial y más significativo en el problema planteado.

Métodos teóricos.

- Histórico-lógico: para analizar la trayectoria de la temática planteada en el campo de acción del objeto de estudio.
- Analítico-sintético: con el objetivo de comprender la gestión de inventario de los almacenes cubanos.
- Modelación: con el objetivo de realizar el análisis del proceso de desarrollo del sistema que dará solución a la problemática planteada.

Métodos particulares.

- Entrevista: como medio ideal para obtener información acerca del campo de acción en cuestión.
- Encuesta: como método de obtención y comparación de las principales ideas acerca del campo de acción.

Metodologías informáticas a utilizar.

- Metodología de desarrollo de software: Rational Unified Process (RUP), teniendo en cuenta que la gestión de inventarios tiene un gran volumen de procesos y que requiere de gran tiempo de desarrollo para la concepción de un sistema con sus funcionalidades.

Para mayor claridad y comprensión de los contenidos de la investigación se ha determinado que el documento conste de tres capítulos fundamentales.

El Capítulo 1 “Fundamentación Teórica” está compuesto por dos partes fundamentales: en la primera se aborda lo referente al estado del arte del tema que se investiga (IR), tendencias y tecnologías actuales del mismo, las herramientas CASE³ que soportan el desarrollo del tema, la caracterización de la metodología de desarrollo así como el desempeño del rol de analista de sistema dentro de la metodología seleccionada, partiendo de un análisis crítico de bibliografía consultada. En la segunda parte, se analizan los procesos de GIA de forma general, además de diferentes sistemas de GIA existentes. Se ofrecen conclusiones parciales para este capítulo.

En el Capítulo 2 “Descripción de la solución para el módulo Nomencladores de SIGIA”, se amplía la situación problema, se analizan y describen los procesos del negocio. Se presentan los artefactos que se generan como resultado de la aplicación del rol, además la estrategia de captura de requisitos para el perfeccionamiento del modelado del sistema. Se muestra el modelado del sistema para el módulo Nomencladores del SIGIA, sobre la base de los requisitos obtenidos. Se validan los requisitos de Software y se evalúan, en general, los resultados. Se ofrecen conclusiones parciales para el capítulo.

Finalmente se procede a las conclusiones generales de la investigación, las recomendaciones y glosario de términos.

³ Computer Assisted Software Engineering. Un tipo de herramientas que soportan la IS generalmente para el modelado.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

El trabajo de investigación que se presenta en este documento se enmarca en la IR dentro de la disciplina de IS, a partir de la cual se desarrollan las actividades que corresponden a la misma aplicadas al proyecto SIGIA, módulo Nomencladores. El presente capítulo pretende en su primer momento realizar el estado del arte del tema en cuestión partiendo de la IS, apoyado en una metodología para el desarrollo eficiente del producto. En un segundo momento se abordan los conceptos de la GIA; que influyen en la posterior comprensión del resto de la investigación.

1.2 IS: punto de partida para el análisis de la IR.

El software de los ordenadores es actualmente crítico para muchos negocios. Funciona en la mayoría de las modernas industrias, maneja diariamente las transferencias internacionales de billones de dólares, y es un elemento clave en los nuevos productos y servicios que utilizamos [Humphrey02]. Como consecuencia, millones de dólares están siendo gastados en el desarrollo del software, el sustento y la vida de la mayoría de las personas dependen de la efectividad de este desarrollo. El resultado del desarrollo de software ha contribuido a ser más eficientes, eficaces y productivos.

El software es un conjunto de programas y su documentación asociada, tales como sus requisitos, diseño, modelos y manuales de usuarios [Sommerville04].

Un sistema informático está compuesto por hardware y software. En cuanto al hardware, su producción se realiza sistemáticamente y la base de conocimiento para el desarrollo de dicha actividad está claramente definida. Sin embargo, respecto del software, su construcción y resultados han sido históricamente cuestionados debido a los problemas asociados [Pressman97]. En los años 60 debido a la evolución ineludible de los sistemas y la necesidad de traspasar las barreras del uso personal de software por los desarrolladores se destapa “la caja de Pandora de la *crisis del software (...)*” [Bonilla02]⁴ caracterizada por problemas en el desarrollo de software, que provocaban demoras en los plazos de entrega, altos presupuestos, falta de respuesta a los requerimientos de los clientes y/o usuarios, y dificultad para usar,

⁴ Miguel Toro Bonilla. Catedrático de Universidad Área de Lenguajes y Sistemas Informáticos, Sevilla, España. Tesis doctoral titulada “Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información.”

mantener y mejorar los sistemas [Dorfman97]. El conjunto de problemas encontrados en el desarrollo de software de computadoras no se limita al software que “no funciona correctamente” [Pressman05]. Los problemas están en cómo desarrollar software de mayor calidad y la mantenibilidad e integridad de los sistemas sobre la base de la captura eficiente de las necesidades de los clientes y/o usuarios.

El primer reconocimiento público de la existencia de problemas en la producción de software tuvo lugar en la conferencia organizada en 1968 por la Comisión de Ciencias de la OTAN en Garmisch (Alemania) para designar la gran cantidad de problemas que presentaba (y aún presenta) el desarrollo de software y el alto índice de fracasos en los proyectos de desarrollo [Bonilla02]. En la misma surge por primera vez el término de **IS** por Fritz Bauer: “*establecer y usar principios de ingeniería orientados a obtener software de manera económica, que sea fiable y funcione eficientemente sobre máquinas reales*” [Bauer69]. Concepto que proporcionará en años posteriores, según Pressman, la línea base de la Ingeniería del Software. La sofisticación y la complejidad pueden producir resultados deslumbrantes cuando un sistema tiene éxito, pero también pueden suponer grandes problemas para aquellos que deben construir sistemas complejos [Pressman05]. La evolución del software ha sido cambiante durante el transcurso del tiempo lo que ha provocado la necesidad de aplicar un enfoque más disciplinado con descanso sobre una organización de calidad.

La IS es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar (funcionar) y mantenerlos. El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE)⁵ [IEEE93] ha desarrollado una definición más completa [Pressman05]: (1) La aplicación de *un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación* (funcionamiento) y mantenimiento de software; es decir la aplicación de la ingeniería de software. (2). El estudio de enfoques como en (1).

Se preocupa por las teorías, métodos y herramientas para el desarrollo del software profesional. Es una disciplina de la ingeniería que se ocupa de todos los aspectos de producción de software. [Sommerville04], [Pressman05]. Una tecnología multicapa apoyada sobre un compromiso de organización de calidad y la cual puede identificar los métodos (indican cómo construir técnicamente el software), el

⁵ Por sus siglas en inglés (Institute of Electrical and Electronics Engineers). Una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.

proceso (es el fundamento de la Ingeniería de Software, es la unión que mantiene juntas las capas de la tecnología) y las herramientas (soporte automático o semiautomático para el proceso y los métodos).

Pressman presenta tres objetivos de esta disciplina: “Los objetivos claves de la ingeniería de software son definir, crear y aplicar 1) una metodología bien definida, dirigida a un ciclo de vida de planeamiento, desarrollo, y mantenimiento; 2) un conjunto establecido de componentes de software que documenta cada paso en el ciclo de vida y muestra un seguimiento paso a paso, y 3) un conjunto de hitos predecibles que pueden ser revisados a intervalos regulares a través del ciclo de vida del software” [Pressman97].

La IS es el resultado de un proceso llamado Ingeniería de Sistemas (ISm) que no se centra solamente en el software sino en el análisis, diseño y organización de diversos elementos en un sistema para constituir un producto, un servicio o una tecnología para transformación o control de información [Pressman05]. La ISm se preocupa por todos los aspectos de desarrollo de los sistemas de computadora basado incluso el hardware, software e ingeniería del proceso. La Ingeniería de Software es parte de este proceso [Sommerville04].

Los ingenieros del software deben adoptar un acercamiento sistemático y organizado a su trabajo y deben usar las herramientas apropiadas y técnicas que dependen del problema a resolver, las limitaciones de desarrollo y los recursos disponibles [Sommerville04].

Por lo visto anteriormente se concluye que:

- La IS es una parte del proceso de ISm, estudian los sistemas informáticos de forma específica y global respectivamente. En la actualidad, existe una clara definición entre los ingenieros de sistemas y los ingenieros de software. Los primeros piensan en la solución completa, el diagnóstico del problema, la formulación de la solución. Los segundos identifican la arquitectura del software requerida para la implementación de la solución, toman en consideración la interacción con las bases de datos, y elaboran el programa. De esta forma se pueden integrar los procesos de desarrollo de software y hardware como un todo y no como la simple suma de sus partes.
- Tiene un enfoque sistemático y abarca todo el ciclo de vida del desarrollo del software.
- Guía mediante el proceso de desarrollo de software los esfuerzos y las personas en la construcción de un producto de calidad.

Esta rama de la ingeniería proporciona un marco de trabajo para el desarrollo del software combinando los anteriores conceptos para construir productos con la calidad requerida, que respondan a los

requerimientos, la planificación y el presupuesto establecidos. La puesta en práctica de esta disciplina (ingeniería de software),(...) evitaría los errores de un proceso de desarrollo inmaduro, centrado en la etapa de implementación y no en todo el ciclo de vida, además de un software difícil de mantener, ya sea por su inaccesibilidad o por su alto costo [Fernández06].

En la actualidad la IS continúa en expansión y merita cada vez más estudios investigativos con resultados prácticos que mejoren y estandaricen sus políticas.

1.2.1 Ingeniería de Requisitos.

En los seres humanos, la comunicación es un acto propio de su actividad psíquica derivado del lenguaje y del pensamiento, además del desarrollo y manejo de las capacidades psicosociales de relación con el otro. A grandes rasgos: permite al individuo conocer más de sí mismo, de los demás y del medio exterior mediante el intercambio de mensajes principalmente lingüísticos que le permiten influir y ser influidos por las personas que lo rodean.

El contenido de comunicación es muy denso. Abundan las ocasiones para las malas interpretaciones o falta de información. Es muy probable que haya ambigüedad [Pressman05]. La necesidad del flujo e intercambio constante de información mediante la comunicación cliente-desarrolladores⁶ determina el entendimiento del problema y el alcance de lo que se pretende automatizar. Es el punto de inicio de las actividades de desarrollo de software. En la medida que el sistema se hace más complejo se necesitan de métodos, técnicas y herramientas adecuadas que soporten y guíen el buen desempeño de estas actividades iniciales.

En la actualidad persisten problemas en el desarrollo de software, entre ellos, un inadecuado entendimiento de las necesidades de los usuarios, incapacidad de absorber cambios en los requerimientos e insatisfacciones de los clientes por inaceptable o bajo desempeño del software. Las principales causas de estos son la administración de requerimientos insuficiente, comunicación ambigua e imprecisa, inconsistencias no detectadas entre requerimientos, diseño y programación, validaciones tardías de los requisitos, enfrentamiento tardío de riesgos y propagación de cambios sin control [Fernández06].

⁶ En este caso el término cliente-desarrollador se utiliza para delimitar las partes interesado y desarrolladores (analistas y equipo en general).

El entendimiento del concepto de “requisitos de software” influye en la posterior comprensión de los términos abordados en el estado del tema de la IR, en la actualidad. Un requisito es una circunstancia o condición necesaria para algo, DRA⁷.

La IEEE Standard Glossary of Software Engineering Terminology⁸ [IEEE90] define un requerimiento como:

1. Condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
2. Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.
3. Una representación documentada de una condición o capacidad como en 1 o 2.

Un requisito de software es una propiedad que debe ser exhibida para solucionar algún problema en la vida real. Por lo tanto, un requisito de software es una propiedad que debe ser exhibido por el software desarrollado o adaptado para solucionar un problema particular [SWEBOK04]. El término “*requisito*” no es usado en la industria del software en una forma coherente. En algunos casos, un requisito es simplemente una declaración de alto nivel abstracta, de un servicio que el sistema debería proveer o una restricción en el sistema. En el otro extremo, es una definición detallada, formal de una función de sistema. El proceso de encontrar, analizando, documentando y comprobando estos servicios y estas restricciones es la **IR**. Se puede pensar en la ingeniería de requisitos como el proceso de comunicación entre los clientes del software, los usuarios y los desarrolladores. No es simplemente un proceso técnico. El sistema de requisitos es influenciado por los gustos de los usuarios, cosas que lo disgustan y los prejuicios, y por los asuntos políticos y administrativos. Éstas son características humanas fundamentales, y las nuevas tecnologías, como los casos de uso, los escenarios y los métodos formales, sirven de mucha ayuda a nosotros para resolver estos problemas. [Sommerville04].

⁷ Vigésima segunda edición del Diccionario de la Real Academia de la Lengua Española. <http://buscon.rae.es/drae/>

⁸ Se decidió usar la terminología en inglés. Glosario estándar de terminología de Ingeniería del Software. El mismo pertenece a la IEEE.

Se define como un proceso que consiste en el análisis de las necesidades y especificación de comportamiento [Wieringa96]; o sea que facilita el mecanismo apropiado para comprender lo que quiere el cliente, analizando necesidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedad y gestionando los requisitos para que se transformen en un sistema operacional [Dorfman97]. Se preocupa por los objetivos del mundo real, funciones y restricciones en los sistemas de software; así como de la relación entre estos factores para precisar las especificaciones del comportamiento del software y a su evolución a través del tiempo y a lo largo de las familias de software [Nuseibeh00], [Zave97]. De forma general, es el proceso, que mediante el dominio del problema, soporta, guía y obtiene los requisitos del sistema proporcionando la solución al conflicto presente (Figura 1 La IR como puente entre los espacios del problema y la solución. Tomada de Tesis de maestría [Fernández06]).



Figura 1 La IR como puente entre los espacios del problema y la solución. Tomada de Tesis de maestría [Fernández06]

Hubo un tiempo en que la frase “los requisitos dicen lo que el sistema hará y no cómo lo hará” resumía todo lo referido a IR. Hoy, mucho tiempo después, la investigación en IR ha producido un gran volumen de conocimiento que incluye terminología, métodos, lenguajes y herramientas [Zave97]. Jackson afirma que se ubica en el punto de encuentro entre lo informal y lo formal del desarrollo de software. Los programas computacionales son efectivamente construcciones formales, que utilizan métodos matemáticos basados en teoría de tipos, precondiciones y postcondiciones, e invariantes. Pero el mundo de los seres humanos y objetos físicos en el cual los requisitos se ubican es informal, y no puede ser tratado adecuadamente por métodos puramente formales [Jackson 2001]. El factor de éxito determinante en esta etapa es el ser humano por la constante comunicación interpersonal que se produce en la misma, lo que conlleva a la complejidad de este proceso.

La IR es un proceso de descubrimiento, refinamiento, modelado y especificación [Pressman 2005].

Durante la IR es necesario que el desarrollador domine el vocabulario del cliente, es algo así como el establecimiento de un protocolo cliente-desarrollador que unifique los intereses y los traduzca de una parte a otra para la transformación de los mismos en un lenguaje funcional que le sirva al desarrollador (Figura 2 La IR como un proceso de comunicación. (Requisitos-C para definir los requisitos del cliente y Requisitos-D traducidos para el desarrollador)). Alcanzar con eficiencia esta meta sería solo un paso, aunque muy definitivo y difícil, de todo el proceso que comprende la IR. Cabe destacar que administrar con equidad y eficacia los requisitos así como el control de sus cambios durante el ciclo de vida de un proyecto de desarrollo de software es una ardua tarea para los ingenieros. Actualmente una de las causas de la disolución de proyectos de software son por no manejar efectivamente sus requisitos.



Figura 2 La IR como un proceso de comunicación. (Requisitos-C para definir los requisitos del cliente y Requisitos-D traducidos para el desarrollador).

Existe un cúmulo de actividades que soportan y guían la IR para lograr su desarrollo organizado. Diferentes autores defienden unas y eliminan otras actividades en dependencia de sus criterios y valoraciones personales.

El proceso de IR puede ser descrito en 5 pasos distintos [SOM97] [Pressman05]: Identificación de Requisitos, Análisis y Negociación de Requisitos, Especificación de Requisitos, Modelado del Sistema, Validación de Requisitos y Gestión de Requisitos. Pohl define cuatro actividades fundamentales: Elicitación de Requisitos, Negociación de Requisitos, Validación y Verificación de Requisitos, Especificación y Documentación de Requisitos. La propuesta de [Durán00] consta de tres actividades principales, elicitación, análisis y validación.

La adopción de uno u otro criterio está en dependencia de las características del proyecto que se lleva a cabo. Las actividades de la IR no son criterios esquemáticos sino que evolucionan según las experiencias de cada desarrollador en este campo de la IS. Teniendo en cuenta lo específico del criterio de [Pressman05] se optó para esta investigación su selección.

Como se ha podido observar la IR es una etapa fundamental dentro del desarrollo de software, podría llamársele “la etapa del entendimiento y gestión de necesidades” pues en la misma debe lograrse la comunicación fluida y unívoca con el cliente. Capturando las necesidades de los usuarios y/o clientes mediante una serie de actividades importantes de descubrimiento y comunicación. El estudio realizado sobre ingeniería de requisitos en los modelos y normas de calidad de los procesos de software revela que el modelo CMMI [Chrissis, 2003], el estándar SPICE [SPICE, 1998] y la Guía para el Cuerpo de Conocimiento de IS [SWEBOK04], concuerdan en las características siguientes [Fernández06]:

- La IR es un proceso iterativo, ya que al ser un proceso de descubrimiento y comunicación, difícilmente llegará a realizarse en forma lineal.
- Los requisitos no siempre son entregados en su totalidad por los clientes y usuarios, así que los ingenieros de requisitos también deben saber descubrirlos.
- Los límites de las actividades de IR son difíciles de establecer por la misma naturaleza del proceso.
- Los requisitos pueden evolucionar tan rápidamente que pueden cambiar antes de haber concluido el desarrollo del sistema.

1.2.1.1 Elicitación de Requisitos (ERE)

La ERE es la parte de la IR en la que se tiene contacto con los clientes y usuarios y donde deben quedar claros el dominio del problema, las necesidades reales del cliente y usuarios finales y la negociación con estos de los requisitos [Durán00]. O sea, en esta etapa se deben capturar los requisitos que luego serán implementados [Fernández06].

La gestión eficiente de los requerimientos influye de forma determinante en el éxito de los proyectos software; identificando los problemas, enfocando la solución al uso exhaustivo de la tecnología y esta a su vez que se adecue, con mayor precisión, a las necesidades de los interesados.

Cada vez se hace más difícil la tarea de entender lo que el cliente quiere y transformarlo en un producto de calidad. Capturar los requisitos así como gestionar y mantener los cambios de los mismos, a lo largo del ciclo de vida del desarrollo de software es tan importante como implementarlo. Los analistas, principales exponentes y responsables de esta actividad, tienen la misión de ser capaz de tener una visión de conjunto y de componer una especificación de requisitos completa, correcta y consistente.

Los requisitos son el resultado acabado de todo el desarrollo de la IS. Por tanto el tratamiento de los requisitos es directamente proporcional a su estado final, es decir a mejor IR mejores requisitos se obtendrán.

La ERE parece un tanto simple al comienzo. Captar las necesidades que debe cubrir el sistema que se implementa y ajustarlas a un tipo de software; su frecuencia de utilización diariamente y las condiciones en las que debe funcionar para explotar el máximo sus potencialidades estas no son tareas fáciles.

Como soporte y enfocada a la mejora de la identificación precisa y sistemática de los requisitos, se han desarrollado diferentes técnicas de elicitación que serán presentadas en el siguiente subepígrafe.

1.2.1.1.1 Técnicas para la identificación de los requisitos (TER)

La mayoría de los profesionales utiliza una o varias técnicas en dependencia del conocimiento que tenga sobre las mismas o de su popularidad. Sin embargo existen varias técnicas que deben ser estudiadas antes de comenzar esta etapa, para poder realizar una selección adecuada a las características del proyecto que se desarrolla. Esta brecha entre la investigación y la práctica ha merecido especial atención, siendo su eliminación el objetivo principal de diversos eventos y publicaciones creados en los últimos años [Fernández06].

A continuación, se presentan algunas de las TER que se consideran más significativas, de las estudiadas, para finalmente argumentar la técnica seleccionada para el caso de estudio.

a) Entrevista

La entrevista es la técnica más utilizada en la obtención de requisitos de software pues es la forma más natural que tiene el ser humano, de intercambiar criterios durante la comunicación. El objetivo de esta es obtener toda la información posible desde la visión que el entrevistado tiene de los requisitos. Se distinguen tres etapas en el acto de la entrevista: apertura, desarrollo y terminación.

Los analistas necesitan ser sensibles a las dificultades que algunos entrevistados crean durante la entrevista y saber cómo tratar con problemas potenciales. Asimismo, es importante considerar no sólo la información que se adquiere a través de la misma sino también, su relevancia. El entrevistador debe centrar la entrevista cuando esta se desvía.

Resulta útil además planificar la entrevista así como evaluar los resultados una vez realizada esta con el fin de evitar riesgos; no pueden ser improvisadas porque estaría expensa a olvidarse detalles importantes.

Los pasos para planificarla y controlarla pueden ser disímiles y adaptables a las necesidades del proyecto. Esta técnica depende en gran medida de las habilidades del entrevistador, la fluidez de su lenguaje y perspicacia para captar las ideas en un tiempo bastante limitado.

b) Joint Application Development (JAD).

El término en español es Desarrollo Conjunto de Aplicaciones, alternativa de la entrevista. Es una práctica, en grupo, que se desarrolla varios días donde el cliente es partícipe junto al equipo de desarrollo de las actividades expresando los problemas así como las posibles soluciones.

El objetivo es romper las barreras con el cliente que no se sienta excluido del proceso de desarrollo y de esta forma puede dar lugar a una declaración más exacta de los requisitos del sistema, a una comprensión mejor de metas comunes, y a una comisión más fuerte al éxito del nuevo sistema.

Cuando los usuarios participan en el proceso del desarrollo de los sistemas, son más probables a sentir un sentido de la propiedad en los resultados, y la ayuda para el nuevo sistema.

Comparado con métodos tradicionales, JAD es más costoso y puede ser incómodo si el grupo es demasiado grande concerniente al tamaño del proyecto. Muchas compañías encuentran, sin embargo, que JAD permite que los usuarios dominantes participen con eficacia en los requisitos que modelan proceso.

c) Revisión de documentos.

Esta técnica depende de la información almacenada por las entidades acerca de los procesos y términos que se manejan dentro de la misma. Las entidades guardan información referente a sus procesos, los modelos o informes necesarios para el desarrollo de la misma o para rendir cuenta a los organismos superiores. Pueden ser además videoconferencias y otros dependiendo del grado de automatización de las mismas.

Este cúmulo de información es estudiado por los analistas en busca de captar bien todos los procesos para determinar los requisitos asociados a estos, y que luego deben ser verificados por otras TER.

La revisión de documentos no es efectiva por si sola, para la ERE, debe ser vinculada con otra (s) para lograr un resultado efectivo.

d) *Cuestionarios y Checklists*⁹

Esta técnica requiere que el analista conozca el ámbito del problema en el que está trabajando. Consiste en redactar un documento con preguntas cuyas respuestas sean cortas y concretas, o incluso cerradas por unas cuantas opciones en el propio cuestionario (*checklist*). Este cuestionario será cumplimentado por el grupo de personas entrevistadas o simplemente para recoger información en forma independiente de una entrevista [Fernández06].

Existen muchas otras técnicas como son: Proceso de Análisis Jerárquico (AHP), Comparación de terminología y otras que no deben restársele importancia. Lo significativo de todas estas técnicas es que no son efectivas, ciento por ciento, por si solas se hace necesario combinarlas para obtener resultados con un mayor grado de certeza.

e) *Casos de uso.*

En un primer momento se desarrollan como técnica para la definición de requisitos pero es usado además para representar la identificación de requisitos. Los casos de uso permiten mostrar el contorno (actores) y el alcance (requisitos funcionales expresados como casos de uso) de un sistema [Fernández06]. Resultan fáciles de entender para el usuario si se acompañan además de visualizaciones gráficas (diagramas de actividad).

Para el caso de estudio presentado en este documento se ha definido el uso de las técnicas de la *entrevista*, *revisión de documentos* y *casos de uso*, según la selección en el próximo capítulo se profundizará más en esta decisión.

1.2.1.2 *Análisis y Negociación de Requisitos (ANR).*

Los requisitos identificados se agrupan por categorías y se organizan, analizando cada requisito con el resto. Esta actividad provoca la identificación requisitos inexistentes, la completitud y ambigüedad de los actuales. Proporciona al analista el grado de profundidad y aceptabilidad de los requisitos elicitados. De aquí pueden surgir nuevos encuentros con los clientes y el posterior análisis de los resultados.

⁹ Listas de chequeo.

Es necesario potenciar la idea del análisis de los requisitos en un primer momento por el equipo de analistas y en un segundo en coordinación directa con el cliente. Determinar las inconsistencias y potenciar el análisis de los nuevos requisitos que surjan son tareas importantes dentro de esta actividad.

Producto del análisis de las deficiencias encontradas en los requisitos con el cliente puede surgir conflictos con el equipo de analistas. Es muy común por parte de cliente y usuarios proponer requisitos contradictorios desde su punto de vista importante.

Durante la negociación se priorizan los requisitos y se llega a un compromiso sobre el conjunto de requisitos a implementar [Fernández06].

La negociación es una actividad significativa para el desarrollo de la IR y requiere de las habilidades del analista para la interpretación y la abstracción en virtud de ubicar las necesidades del cliente.

1.2.1.3 Especificación de Requisitos (EPR)

La obtención de una EPR de alta calidad es fundamental para asegurar que el software se corresponde con las necesidades del cliente. Sin embargo, obtener una EPR de calidad es difícil [Toval01]. La especificación describe la función y características de un sistema de computación y las restricciones que gobiernan su desarrollo [Pressman05].

Los requerimientos pueden dividirse en dos grupos: requerimientos funcionales y requerimientos no funcionales.

Los requerimientos funcionales (RF) definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Los requerimientos no funcionales (RNF) soportan estas transformaciones. Tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

1.2.1.4 Modelado del sistema (MSI)

Los modelos de sistema ayudan al analista a comprender la funcionalidad del sistema y a su vez hacen más fácil la comunicación con el usuario. Estos modelos representan el sistema desde varias perspectivas. Se manejan 3 perspectivas del sistema: externa, de comportamiento y estructural. La

externa muestra el entorno del sistema; la de comportamiento muestra el comportamiento del sistema y la estructural que muestra el sistema o una arquitectura de datos [Sommerville04].

1.2.1.5 Validación de requisitos (VRE)

La VRE examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencias, sin omisiones, que los errores detectados hayan sido corregidos, y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto [Pressman05].

La VRE no puede hacerse sin la participación y presencia de clientes, usuarios y demás implicados [Fernández06]. Este es una condición determinante en el éxito de esta actividad.

1.3 Influencia de una deficiente IR en el desarrollo de un proyecto

La parte más difícil de construir de un sistema software es decidir qué construir. [...] Ninguna otra parte del trabajo afecta más negativamente al sistema final si se realiza de manera incorrecta. Ninguna otra parte es más difícil de rectificar. Por lo tanto, el desarrollo del software sólo puede ser iniciado cuando se tiene bien establecido lo que se quiere producir [Fernández06]. Uno de los problemas que más contribuyen a la baja calidad de los proyectos de software son las deficiencias en los requisitos; estos por ser la base de todo el sistema son más difíciles de rectificar mientras más avanzado esté su desarrollo.

Otras investigaciones arrojan que [Fernández06]:

- En la fase de requerimientos se introduce más del 55% de los defectos totales detectados en el software.
- De todo el mantenimiento correctivo de software, que demanda el 75% de los recursos totales, más del 80% proviene de la Fase de Requerimientos.
- Un error no detectado en la fase de requerimientos cuya corrección costaría apenas 1 unidad de tiempo o dinero, costará 67 veces más, al detectarlo en la fase de producción.

A pesar de la riqueza de conocimiento de desarrollo, la experiencia, y las herramientas disponibles actualmente, un porcentaje sustancial de proyectos del software dejan de operar, a menudo porque los requisitos no son correctamente determinados y definidos al principio, o no son manejados correctamente durante el desarrollo del proyecto. El éxito de un proyecto depende de la gerencia efectiva de los

requisitos. Los errores en los requisitos son el tipo más común de error en el desarrollo de sistemas y los más costosos para arreglar [Leffingwell03].

Los errores en los requisitos tienen probabilidad de consumir 25 % para un 40 % del presupuesto de proyecto total [Leffingwell03]. Es notable la importancia de realizar una eficaz IR en los proyectos de desarrollo de software. Averiguar correctamente las necesidades de los clientes, disminuye costosas tareas de reingeniería después de la codificación [Kovitz01]. Nuestro reto es entender los problemas de los usuarios en su cultura y su idioma y construir sistemas que respondan a sus necesidades [Leffingwell03].

1.4 Tendencias Actuales de la IR

1.4.1 Reutilización de requisitos

Se coincide con lo planteado por los autores [Nuseibeh00] y [Toval01] que las nuevas tendencias y desafíos de la IR es la reutilización de modelos de requisitos. Esto influirá de forma satisfactoria en la obtención de requisitos consistentes y sin ambigüedades. Se espera que se desarrollen modelos de referencia de requisitos en muchos dominios de aplicación, de manera que se reduzca drásticamente el esfuerzo de desarrollo de nuevas EPRs, facilitando además la selección de paquetes de software.

Los requisitos dentro de dominios similares y/o para similares tareas tienen una mayor probabilidad de ser similares que los componentes software que los implementan. Pero sorprendentemente, las técnicas para recuperar, adaptar y consolidar requisitos reutilizables han recibido relativamente poca atención en relación con todo el trabajo en reutilización del software [Toval01].

1.4.2 Patrones de requisitos (PR)

Los patrones de reutilización no son más que el fruto de la constatación de la repetición de requisitos similares en las prácticas académicas en las que se han utilizado las ideas propuestas en esta tesis. Dichas prácticas académicas tienen en común su dominio de aplicación, sistemas de información para pequeños negocios (gestión de pequeños comercios, gestión de bibliotecas, etc.), por lo que parece natural la repetición de requisitos similares [Durán00]. Finalmente un PR puede ser visto como un conjunto de requisitos reutilizable. Hay que tener en cuenta que un buen proyecto de software no solamente incluye requisitos, sino también, por ejemplo, riesgos derivados de estos requisitos, las pruebas a estos requisitos, etc.

Según [Durán00] y coincidiendo con otros trabajos se exponen a continuación diez heurísticas generales sobre la introducción de la reutilización en el proceso de ingeniería de requisitos.

Básicamente se propone:

- Identificar familias de sistemas en los que los requisitos suelen coincidir.
- Desarrollar requisitos parametrizables abstractos.
- Separar los aspectos específicos de los generales
- Intentar identificar PRs al trabajar en dominios específicos.
- Intentar reutilizar también los procesos de obtención de ciertos tipos de requisitos, es decir las preguntas a realizar a los clientes y usuarios, las consideraciones a tener en cuenta a la hora de especificarlos, etc.

Se han encontrado PRs a un nivel de abstracción más alto. Según [Durán00] se proponen los tres siguientes:

- *PR Especificar*: este PR aconseja describir *cómo* puede el usuario de un sistema seleccionar (*especificar*) una determinada información (para modificarla, eliminarla o consultarla) en un requisito separado y hacer referencia a dicho requisito cuando sea necesario. Por ejemplo, un requisito podría establecer que "el sistema deberá permitir al usuario seleccionar a los clientes por su número de DNI, por sus apellidos o por su número de teléfono" y posteriormente otro que dijera que "el sistema deberá permitir a los usuarios modificar la información correspondientes a los clientes seleccionados mediante el procedimiento descrito en el requisito X (el requisito anterior)" (Figura 3 Aplicación del PR Especificar en los casos de uso).

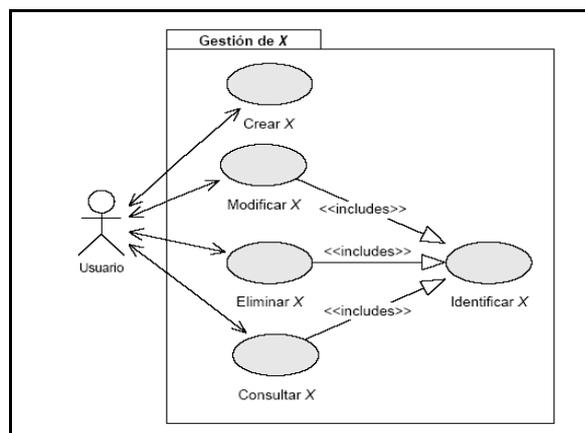


Figura 3 Aplicación del PR Especificar en los casos de uso

- *Patrón Presentación*: este PR recomienda limitarse a indicar qué datos debe solicitar o presentar el sistema sin entrar en detalles concretos de interfaz de usuario.
- *Patrón Priorizar*: este otro PR sugiere que, en el caso de que el usuario desee poder ordenar (*priorizar*) la información presentada por el sistema, se separen las posibles formas de ordenar dicha información en un requisito aparte y se referencie desde los que sea necesario, de forma similar al PR *Especificar*.

Aquí se presentan ejemplos de algunos patrones, ya reconocidos, existe otra gran variedad de patrones de requisitos y las tendencias respecto a su uso continúan en ascenso.

1.5 Lenguajes de Modelado

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad [Jacobson00]. De propósito general, para el modelado orientado a objetos.

Es un lenguaje gráfico para visualizar, especificar y documentar un sistema de software. Ofrece un estándar para describir un panorama del sistema (modelo), incluyendo aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables y aspectos conceptuales como los procesos de negocios y funciones del sistema [Jacobson00]. UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real.

Es importante recalcar que UML es un "lenguaje" para especificar y no un método o un proceso, se utiliza para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir -es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para soportar una metodología de desarrollo de software pero no especifica en sí mismo qué metodología o proceso usar.

UML se divide fundamentalmente en dos partes: vistas y diagramas. Las vistas son una abstracción que muestra un aspecto particular del sistema. Por ejemplo, vista de casos de uso, de diseño, de implementación, de procesos, de implantación. La vista de casos de uso es la más relacionada con la IR y muestra la funcionalidad del sistema desde el punto de vista de un actor externo que interactúa con él.

Los diagramas son una representación gráfica de un conjunto de elementos. Visualizan un sistema desde diferentes perspectivas y se agrupan en tres grupos fundamentales: de estructura estática, de comportamiento y de implementación.

1.6 Metodologías de desarrollo de software

Según [Wieringa96] una *metodología* es el estudio de los métodos; de manera que la metodología de las matemáticas es el estudio de los métodos que se utilizan para encontrar y comprobar verdades matemáticas. También define al método como un medio sistemático para trabajar mediante el cual se obtiene un resultado deseado.

Rechtin en 1997 [Cockburn00] define cuatro tipos específicos para caracterizar a las metodologías:

- Normativas: Basadas en soluciones o secuencias de pasos para trabajar con una disciplina. En el caso específico del desarrollo de software un ejemplo podría ser el la verificación de los diagramas de estado.
- Racionales: Basadas en métodos y técnicas; y son utilizados en el análisis de sistemas y en las disciplinas de ingeniería.
- Participativas: Basadas en los stakeholders¹⁰ y captura aspectos del de los clientes.
- Heurísticas: Basadas en lecciones aprendidas.

Tomando como referencia esta definición, se puede concluir que una *metodología de desarrollo de software*, no es más que el estudio de los métodos más apropiados que se emplean para desarrollar software de manera eficiente; o como precisan otros autores [Jacobson00], define Quién debe hacer Qué, Cuándo, y Cómo debe hacerlo.

Por lo general las metodologías se definen a partir de una serie de parámetros que cada una implementa según sus concepciones. Estos según [Cockburn00] están reflejados en la siguiente figura:

¹⁰ Stakeholders: Son todos los interesados en el proyecto, clientes, desarrolladores, inversionistas, etc.

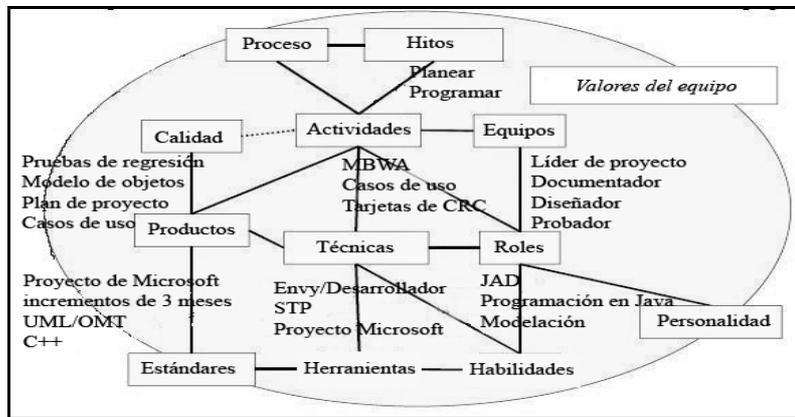


Figura 4 Parámetros que definen una metodología. Tomado de [Cockburn00]

Durante su evolución, las metodologías de desarrollo de software se han separado en dos grupos: las tradicionales (robustas o pesadas) y las ágiles que han surgido recientemente. Las metodologías tradicionales se basan en modelos de proceso que suponen la estabilidad de los requerimientos del cliente que se ha demostrado en la práctica que son bastante inestables; por tanto ha adoptado una rígida planificación para tratar de solucionar dicho problema [Arboleda05].

1.6.1 Metodología Tradicional: Proceso Unificado.

El Dr. Xavier Ferré Grau¹¹ [Ferré05] en su tesis de doctorado afirma que el proceso que está tomando mayor atención en la Ingeniería de Software en la actualidad es el Proceso Unificado y que esto es debido a que sus impulsores son los mejores metodólogos del desarrollo orientado a objetos de la década del 90, James Rumbaugh, Ivar Jacobson y Grady Booch. Sostiene además que adopta un verdadero enfoque iterativo y que es el más aplicado en proyectos reales. Alega que Rational Unified Process (RUP) es una particularización del modelo de proceso representado por el Proceso Unificado.

Junto al Lenguaje Unificado de Modelado (UML) constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

1.6.1.1 Estructura Dinámica del proceso. Fases e iteraciones

Las características de RUP se centran en tres aspectos fundamentales:

¹¹ Licenciado en Informática. Universidad Politécnica de Madrid. Noviembre, 1996. Doctor en Informática. Universidad Politécnica de Madrid. Junio, 2005. Línea de investigación: Integración de la Usabilidad en el Proceso de Desarrollo Software. <http://is.ls.fi.upm.es/xavier/cv.html#titulacion>.

Iterativo e incremental: principio de división en iteraciones pequeñas, tanto para el desarrollo de las funcionalidades, como para la planificación formal del proyecto. El trabajo se divide en piezas pequeñas; cada uno provee un subproducto incremental.

Centrado en la arquitectura: La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, las bases del sistema que son útiles para entenderlo, desarrollarlo y producirlo económicamente.

Guiado por casos de uso: Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso (cómo se llevan a cabo).

RUP divide el proceso de desarrollo en ciclos¹², teniendo un producto al final de cada ciclo, los mismos se dividen en fases que finalizan con un hito (**Anexos I, II**):

- **Concepción o Inicio:** Durante la fase de inicio se define el modelo del negocio y el alcance del proyecto. Se identifican todos los actores y Casos de Uso, y se diseñan los Casos de Uso más esenciales (aproximadamente el 20% del modelo completo). Se desarrolla, un plan de negocio para determinar que recursos deben ser asignados al proyecto.
- **Elaboración:** El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos. En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio.
- **Construcción:** La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las iteraciones sucesivas. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión del producto.

¹² La vida de un sistema transcurre a través de ciclos de desarrollo, desde su nacimiento hasta su muerte, en cada ciclo se repite el proceso unificado de desarrollo, Cada ciclo consta de cuatro fases: Inicio, elaboración, construcción y transición. Cada ciclo concluye con una versión del producto.

- **Transición:** Esta fase en general, ejecuta tareas relacionadas con la configuración, instalación y facilidades de uso del producto.

1.6.1.2 *Estructura Estática del proceso. Flujos de trabajo, actividades, artefactos y roles.*

Un proceso de desarrollo de software define quién hace qué, cómo y cuándo. RUP define cuatro elementos: los roles, que responden a la pregunta ¿Quién?; las actividades, que responden a la pregunta ¿Cómo?; los productos, que responden a la pregunta ¿Qué? y los flujos de trabajo de las disciplinas, que responde a la pregunta ¿Cuándo?

Flujos de trabajo (FT)

Con la enumeración de roles, actividades y artefactos no se define un proceso, se necesita contar con una secuencia de actividades realizadas por los diferentes roles, así como la relación entre los mismos. Un FT es una relación de actividades que producen resultados observables.

Los propuestos por esta metodología son: Modelado del Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Gestión del proyecto, Configuración y Control de Cambios, Entorno. Distribuidos a lo largo de todas las fases que comprende la metodología por la propiedad de esta de abarcar todo el ciclo de vida de desarrollo del software. (Ver Anexo II. Estructura de RUP).

Actividades

Una actividad es una unidad de trabajo que una persona que desempeñe un rol puede ser solicitado a que realice. Las actividades tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto.

Artefactos

Un producto o artefacto es una parte de información que es producido, modificado o usado durante el proceso de desarrollo de software. Son los resultados tangibles del proyecto, las cosas que se van creando y usando hasta obtener el producto final.

Un artefacto puede ser cualquiera de los siguientes:

- Un documento (documento arquitectura del software).
- Un modelo (Modelo de Casos de Uso).

- Un elemento del modelo, un elemento que pertenece a un modelo como un Caso de Uso o un subsistema.

Roles

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas.

Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como su mantenimiento y ser responsable de la producción de un conjunto de artefactos.

RUP define grupos de roles, agrupados por participación en actividades relacionadas. Estos grupos son: Analistas, Desarrolladores, Gestores, de Apoyo y otros.

Analistas:

- Analista de procesos de negocio.
- Diseñador del negocio.
- Analista de sistema.
- Especificador de requisitos.

Durante los dos primeros FT de RUP recae el peso de las actividades de la IR en vistas a capturar las necesidades del cliente y visualizarlas en los requisitos del sistema. La metodología provee de artefactos, roles, y actividades que complementan y guían el desarrollo del modelado del negocio y la administración de los requisitos de forma flexible y adaptable a las necesidades específicas de cada proyecto. Es responsabilidad del equipo de desarrollo identificar lo que se desea construir para satisfacer las expectativas del cliente, en este caso el beneficiado.

1.6.2 Rol Analista que propone la metodología RUP

1.6.2.1 Rol de Analista de Sistemas (AS) que propone la metodología RUP

El AS es imprescindible en cualquier organización, producto a las destrezas que este debe poseer y los beneficios que produce a la misma.

El rol de AS ha sido concebido según RUP [Rational03] como el que guía y coordina la captura de los requerimientos y la modelación de casos de uso, de manera que resume las funcionalidades del sistema y

lo delimita. Para esto la persona que ocupe el rol debe ser un experto identificando y comprendiendo problemas y oportunidades; debe poseer grandes habilidades para comunicarse, así como tener algún dominio del negocio y las tecnologías.

Debe ser una persona versada en tópicos de IS, con amplios conocimientos de técnicas de relevamiento, especificación de requerimientos, y manejo de cambios, que posee bastante conocimiento sobre el negocio en el que opera el Cliente, quien financia el proyecto [Schenone04].

El AS, es la persona que debe establecer la relación con el responsable del área solicitante de sistemas de información y llevar sus requerimientos al procesamiento electrónico. La especialización de este rol requiere que, además de tener una formación integral en informática, cuente con determinadas características personales como ser inteligente, perceptivo, hábil para planear, negociador, fuerte de carácter, analítico, disciplinado, innovador, hábil conversador, buen escucha, sentido de propósito, descubridor de detalles, diplomático, orden mental.

Todas las características personales son importantes, sin embargo, ser un buen escucha y negociador, es tan significativo como la obtención del éxito o fracaso de los sistemas de información, ya que dependiendo de cuán mejor se escuchan las necesidades de los usuarios, podrá definirse el diseño idóneo y satisfacer su petición de acuerdo con sus expectativas.

La capacidad para comunicarse y hacerse entender por sus semejantes es lo primordial para que todas las actividades asociadas a este rol se desarrollen eficazmente. No solo capacidad para interactuar sino también para redactar y mantener los cambios de los requisitos a lo largo de todo el ciclo de vida de la construcción de software.

Tareas del AS.

- Analizar y definir en coordinación con el supervisor de proyectos y el usuario, la situación actual del sistema o procedimiento utilizado, la situación deseable para el mismo y los objetivos del sistema, así como los procesos que intervienen y su normatividad; para confirmar la factibilidad del proyecto.
- Analizar y diseñar en coordinación con el supervisor los elementos necesarios para el proyecto, modificaciones generales para el flujo de información y procedimientos del sistema, para asegurar un control óptimo y seguridad de los datos y uso eficiente de los recursos.

- Analizar y definir en coordinación con el supervisor de proyectos los nuevos requerimientos del sistema y los impactos que las adecuaciones pudieran causar en su operación.
- Participar en las demás tareas inherentes al puesto y las funciones del departamento.

1.6.2.2 Rol de Analista de Procesos del Negocio (APN)

Para el caso específico de los procesos del negocio RUP [Rational03] también propone el rol de APN que sería el encargado de definir la arquitectura del negocio, y en general de liderar la modelación de los casos de uso del negocio determinando cuales son los casos de uso y actores del negocio y como estos interactúan. La persona que ocupe este rol debe tener igualmente excelentes habilidades para comunicarse, pero es fundamental que tenga dominio del negocio.

1.7 Ingeniería de Software Asistida por Computadora (CASE)

CASE es un acrónimo de las siglas en inglés **C**omputer **A**ided **S**oftware **E**ngineering o lo mismo en español que Ingeniería de Software Asistida por Computadora.

1.7.1 Herramientas CASE para el manejo de requisitos

La utilización de la tecnología en función de su propio desarrollo crece vertiginosamente. En la actualidad se cuenta con una gran variedad de herramientas CASE disponibles, para el tratamiento de la IR. Estas herramientas se concentran en capturar requerimientos, administrarlos y producir una especificación de requisitos.

El uso de una herramienta de gestión de requisitos proporciona a la organización [Mcdonald05]:

- Ahorro en costes de especificación y de desarrollo minimizando el impacto de errores.
- Mejora la calidad mediante un adecuado análisis y gestión de los requisitos.
- Facilita la reutilización real.
- Mejora la productividad facilitando la reutilización real desde la especificación.
- Reduce las no-conformidades del sistema.
- Permite controlar la especificación.
- Permite administrar más fácilmente la especificación.
- Ayuda a cumplir con estándares de calidad.
- Proporciona un repositorio no propietario de especificación.
- Permite centralizar toda la información del problema.

- Permite especificar sistemas de una forma estructurada y gráfica.
- Proporciona una trazabilidad completa de la especificación, etc.

La principal ventaja de la utilización de una herramienta CASE, es la mejora de la calidad de los desarrollos realizados y, en segundo término, el aumento de la productividad.

A continuación se explicarán a groso modo las características de estas herramientas CASE (RequisitePro, IrqA).

1.7.1.1 RequisitePro

Es una herramienta centrada en documentos, que almacena los requisitos asociándolos a documentos (aunque también permite guardarlos directamente en la base de datos), mientras que las otras herramientas están orientadas a requisitos. Auxilia especialmente en el control de cambio de requisitos, con trazabilidad para especificaciones de software y pruebas [Mcdonald05].

Esta herramienta según [IBM06] tiene tres lineamientos fundamentales:

- El Poder de una Base de Datos y la Libertad de Microsoft Word.
- Conoce el estado del proyecto y el impacto de los cambios.
- Integra los requisitos con las herramientas y el equipo de trabajo.

Este CASE se integra con aplicaciones para la administración de cambios y con herramientas de modelado de sistemas y de pruebas. Esta integración asegura que los diseñadores conozcan los requerimientos del usuario, del sistema y del software en el momento de su desarrollo. Con esta integración, se pueden conservar juntos todos los requisitos y ser manipulados por todos los miembros del equipo.

Para RequisitePro los atributos de los requisitos son la principal fuente de información para ayudar a planear, comunicar y rastrear las actividades a través del ciclo de vida. Cada proyecto tiene necesidades únicas y se deberán seleccionar las propiedades críticas para asegurar su éxito: prioridad de desarrollo, estado, autor, responsable, relaciones, fecha de registro, fecha última modificación, versión, etc.

Una característica importante es que la curva de aprendizaje de RequisitePro es pequeña, este puede ser basado en el uso de tutoriales [Fernández06].

1.7.1.1.1 Ventajas significativas de Rational RequisitePro

Rational RequisitePro mantiene los equipos de proyectos al día gracias a la creación, análisis y gestión de los requisitos de aplicaciones y casos de uso [IBM06].

- Un producto potente y fácil de utilizar para la gestión de requisitos y casos de uso que propicia una mejor comunicación, mejoras en el trabajo en equipo y reduce el riesgo de los proyectos.
- Combina la interfaz conocida y fácil de utilizar de los documentos de Microsoft Word con potentes funciones de base de datos para conseguir la máxima eficacia en análisis y consulta de requisitos.
- Proporciona a los equipos la posibilidad de comprender el impacto de los cambios.
- Garantiza que todos los componentes del equipo estarán informados de los requisitos más actuales para asegurar la coherencia.
- Proporciona acceso basado en web para los equipos distribuidos.

1.7.1.2 *IRqA*

IRqA es una herramienta de Gestión e IR específicamente diseñada para mejorar la calidad de los sistemas y reducir los errores permitiendo construir EPRs completas y consistentes.¹³

- Soporta actividades más allá de la GR, permitiendo realizar un análisis de la especificación de requisitos, y gestionar las pruebas de aceptación y validación, manteniendo todos los elementos de estas actividades completamente trazados dentro de una misma herramienta.
- Sus funcionalidades textuales y gráficas, potentes pero sencillas a la vez, son idóneas para aquellas organizaciones en busca de una solución única que sea capaz de cubrir todas sus necesidades en el ámbito de la Gestión e IR.
- El enfoque gráfico de IRqA a la IR permite a todo tipo de usuarios (clientes, proveedores, jefes de proyecto, analistas, responsables de pruebas) gestionar información compleja de una manera sencilla y colaborativa.

1.7.2 *Herramienta CASE para el desarrollo de software*

Las herramientas de desarrollo de software diseñan y construyen aplicaciones y dan soporte al desarrollo e implantación de las mismas.

La solución Rational combina las mejores prácticas de IS, herramientas líderes de mercado y servicios profesionales para impulsar la mejoría continua de capacidad del equipo.

¹³ En el sitio web del producto se pueden encontrar mas detalles de esta herramienta <http://www.irqaonline.com/spanish.htm>

Las herramientas Rational unifican el equipo de proyecto, está basada en tres ideas clave: provisión de herramientas integradas para cada miembro del equipo de software, un conjunto robusto de capacidades de unificación del equipo y un compromiso con tecnologías abiertas, extensibles, proporcionando un conjunto robusto de servicios comunes.

Individualmente, las herramientas Rational son líderes en sus respectivas categorías de mercado. Combinadas, ofrecen automatización sin paralelo, a través del ciclo de vida de desarrollo de software. Las herramientas Rational abarcan plataformas Windows, UNIX, Linux y de mainframe, y soportan una multitud de lenguajes, IDEs y ambientes operacionales; para asegurar la interoperabilidad a través de equipos y disciplinas. La plataforma Rational fundamentalmente mejora la manera que las organizaciones tienen de construir software.

El resultado final: una infraestructura de alto desempeño, flexible, posible de extender por una comunidad abundante de desarrolladores y asociados.

1.7.2.1 Rational Rose Enterprise.

Herramienta líder en el mundo de modelación visual para el negocio, análisis de requerimientos y diseño de arquitectura de componentes.

Herramienta de diseño unificada orientada a objeto que usa UML como lenguaje de modelado. Su diseño está orientado al negocio y centrado en los casos de uso. Utiliza un lenguaje estándar facilitando así la comunicación y las capacidades de ingeniería inversa.

IBM Rational Rose Enterprise es uno de los productos más completos de la familia Rational Rose. Al igual que todos los productos de Rational Rose, ofrece un lenguaje de modelado común que agiliza la creación del software.

Incluye también estas funciones:

- Los componentes del modelo se pueden controlar independientemente, lo que permite una gestión y un uso de modelos más granular.
- Permite desarrollo multiusuario.
- Es multiplataforma.
- Permite la integración con modelado de datos.
- Genera documentación del sistema.

- Generación de código en lenguaje Ada, ANSI C++, C++, CORBA, Java y Visual Basic, con funciones configurables de sincronización entre los modelos y el código.
- Soporte para Enterprise Java Beans 2.0.
- Funciones de análisis de calidad de código.
- Modelado en UML para diseñar bases de datos, que integra los requisitos de datos y aplicaciones mediante diseños lógicos y analíticos.
- Integración con otras herramientas de desarrollo de IBM Rational.
- Posibilidad de publicar en la Web modelo e informes para mejorar la comunicación entre los miembros del equipo.

1.8 Procesos de GIA. Softwares de Gestión cubanos.

1.8.1 Almacenes

Un almacén se define como el medio de: "Realizar las operaciones y actividades necesarias para suministrar los materiales o artículos en condiciones óptimas de uso y con oportunidad, de manera que se eviten paralizaciones por falta de ellos o inmovilizaciones de capitales por sobre existencias" [Pardo04].

Función de los Almacenes:

1. Mantener las materias primas a cubierto de incendios, robos y deterioros.
2. Permitir a las personas autorizadas el acceso a las materias almacenadas.
3. Mantener en constante información al departamento de compras, sobre las existencias reales de productos.
4. Llevar en forma minuciosa controles sobre los productos (entradas y salidas).
5. Vigilar que no se agoten los materiales (máximos – mínimos).

El almacenaje es un área crítica para el servicio al cliente, por lo que se requiere un enfoque más activo, donde el servicio al cliente sea medido, evaluado y manejado dentro del almacén. El cliente siempre espera el producto correcto por lo tanto el almacén constituye una herramienta importante para satisfacción del cliente [Pardo04].

1.8.1.1 Tipos de almacén

Existen varios tipos de almacén, entre ellos se encuentran: Terminados, Industriales, Materia Prima, Productos Semielaborados, Piezas Separadas, Piezas de Recambio, Herramientas y Utilaje, Aprovechamiento en General, Productos, Almacenes de Distribución y Depósitos.

- Almacenes Industriales: comprende el conjunto de almacenes de una industria para almacenar las materias primas y los productos terminados. Dentro de estos almacenes industriales tenemos:
 - ✓ Almacén de Materia Prima: almacena las materias primas que intervienen directamente en la composición de los productos terminados.
 - ✓ Almacén de Producto Semielaborados: dedicado al almacenamiento de los materiales que han sufrido algunas transformaciones en el proceso productivo.
 - ✓ Almacén de Piezas de recambio: para almacenar piezas destinadas al servicio de post-venta, con el objeto de efectuar reparaciones.
 - ✓ Almacén de Productos Terminados: destinados al almacenamiento de productos para ser suministrados o entregados a los clientes.
 - ✓ Almacén de Herramientas y Utilaje: controla todos los aspectos relativos al instrumental de producción, tales como herramientas, plantillas, matrices, etc.
 - ✓ Almacén de Aprovechamiento en General: para almacenamiento de insumos que intervienen indirectamente en la fabricación, tal es el caso de combustibles, aceites, lubricantes y material de embalaje.
- Almacén de Distribución: destinados a almacenar y vender artículos, productos, colocados a disposición del consumidor.
- Depósitos: lugar concebido y equipado para las mercaderías colocadas en depósitos por trato entre el depositante y el depositario.

Existen empresas que además clasifican los almacenes en cuanto a la forma de pago al proveedor en:

- Propios: se paga al suministrador el producto por alguna forma de pago establecida en el momento de efectuada la compra.
- Consignación: los productos son pagados al suministrador luego de ser comercializados.

1.8.2 Gestión de Inventario de Almacén

La forma en que una empresa administra físicamente el inventario y los espacios de almacenamiento afectan su grado de eficiencia en costos y tiempo en las operaciones y compras.

Se entiende por Gestión de Inventarios, todo lo relativo al control y manejo de las existencias de determinados bienes, en la cual se aplican métodos y estrategias que pueden hacer rentable y productivo la tenencia de estos bienes y a la vez sirve para evaluar los procedimientos de entradas y salidas de dichos productos.

Un sistema de inventario es un conjunto de políticas y controles que supervisa los niveles de inventario y determina cuáles son los niveles que deben mantenerse, cuándo hay que reabastecer el inventario y de qué tamaño deben ser los pedidos.

La gestión de inventarios se define en la literatura con diferentes acepciones, tales como: "El conjunto de acciones destinadas a minimizar los gastos e incrementar los beneficios originados en el almacenamiento de existencias".

En la Gestión de Inventarios están involucradas tres actividades básicas a saber:

1. **Determinación de las existencias:** La cual se refiere a todos los procesos necesarios para consolidar la información referente a las existencias físicas de los productos a controlar.
2. **Análisis de inventarios:** La cual esta referida a todos los análisis estadísticos que se realicen para establecer si las existencias que fueron previamente determinadas son las que deberíamos tener en nuestra planta, es decir aplicar aquello de que "nada sobra y nada falta", pensando siempre en la rentabilidad que pueden producir estas existencias.
3. **Control de producción:** La cual se refiere a la evaluación de todos los procesos de manufactura realizados en el departamento a controlar, es decir donde hay transformación de materia prima en productos terminados para su comercialización [Repo07].

La función principal de la gestión de inventarios es determinar la cantidad suficiente y tipo de los insumos, productos en proceso y terminados o acabados para satisfacer la demanda del producto, facilitando las operaciones de producción y venta y minimizando los costos al mantenerlos en un nivel óptimo.

Su importancia radica en los siguientes aspectos:

- Optimización de los tiempos. La producción y la entrega por lo general no ocurren de manera instantánea, por lo que se debe contar con existencias del producto a las que se pueda recurrir oportunamente y que la venta real no espere hasta la culminación del proceso de producción.
- Mantenimiento del nivel competitivo. La demanda de los consumidores debe ser satisfecha de manera rápida y completa para evitar que el comprador recurra a la competencia, por lo que no sólo debe contar con un inventario suficiente para satisfacer la demanda del mercado, sino que, además, se debe considerar una cantidad adicional (inventario de seguridad) para las solicitudes inesperadas.
- Protección contra aumentos de precios y escasez de materia prima. Cuando se pronostica un aumento significativo en los precios de las materias primas básicas, se debe almacenar una cantidad suficiente al precio más bajo que predomine en el momento. De la misma forma, si se prevé escasez de materias primas necesarias, es indispensable contar con una reserva para continuar regularmente con las operaciones de producción.

La base de toda empresa comercial es la compra y venta de bienes o servicios; de aquí la importancia del manejo del inventario por parte de la misma. Este manejo contable permitirá a la empresa mantener el control oportunamente, así como también conocer al final del periodo contable un estado confiable de la situación económica de la empresa [Pardo04].

Como punto de arranque para establecer un sistema de inventarios adecuado a las necesidades de la empresa, es preciso conocer la naturaleza de la demanda del material que va a almacenarse. Como se sabe, la función fundamental que cumplen los inventarios es la de actuar como amortiguador entre la disponibilidad de materiales y su demanda, ejercida tanto por los clientes externos como por los internos [Acevedo00], [Repo07].

1.8.3 Descripción de algunos Sistemas de Gestión de Inventario de Almacén en Cuba

En el país existen sistemas de GIA pero los mismos se encuentran centralizados en sectores determinados. Los mismos deben incluir además las últimas resoluciones de la doble moneda emitidas por el MFP y el MEP; situación que provoca la adopción por parte de las empresas de otras medidas organizacionales. Muchas entidades mantienen la gestión de sus inventarios por los métodos tradicionales, es decir, sin explotar las potencialidades de las tecnologías.

A continuación se muestran algunos sistemas de gestión de inventario para el Turismo y para la Empresa de Tecnomática.

1.8.3.1 Sistema Interhotel (IHMM)

Este Sistema de Gestión Hotelera (SGH) abarca las principales áreas de una instalación hotelera, vinculando e integrando en un solo sistema las distintas áreas del mismo (carpeta, ama de llaves, reservas y comercial, relaciones públicas, economía, compras y almacén, mantenimiento, etc.) desarrollado por el Grupo de Electrónica para el Turismo GET en los años 90. En vistas a mejorar las funcionalidades del mismo surge una nueva versión del sistema IHMM2000, con el cual se trabaja actualmente en diferentes instalaciones.

IHMM2000 centraliza su funcionamiento en el sector del Turismo, no abarcando así las especificaciones de los restantes sectores del país.

El sistema de Almacén de IHMM, tiene como objetivo realizar toda la gestión de los almacenes y de los departamentos de cada una de las instalaciones, y permitir un control de los productos tanto en existencias como en valores. Brinda además la posibilidad de realizar varios tipos de operaciones como compras, movimientos entre secciones, gastos, ventas, escandallos de elaboración, despieces, sus respectivas reversiones, y otras como fijaciones de inventario, cambios de código de los productos y rebajas automáticas desde los puntos de ventas. Facilita la consulta de cualquier operación realizada independientemente de su fecha, incluidas las anulaciones de éstas, la historia de los productos, y la obtención de un gran número de listados y resúmenes de múltiple interés.

Permite además la conexión con el módulo de Contabilidad, garantizando la generación automática de los comprobantes, y la facilidad de conocer los valores en existencia, de las operaciones diarias y de las compras a los proveedores. El problema fundamental del sistema de Almacén actual implantado en las instalaciones hoteleras, es que está desarrollado en Foxpro V2.6, con bases de datos .DBF, lo que contribuye a la poca seguridad e integridad de los datos almacenados. Además no tiene implementadas las últimas resoluciones del MEP y del MFP, las cuáles afectan al mismo en su concepción.

1.8.3.2 Suite ZUN

Sistema de Gestión Hotelera integrado por módulos fácilmente adaptables a las particularidades de cualquier hotel, independientemente de la cantidad de habitaciones que posea o la complejidad de su operación.

Suite Zun, con una concepción totalmente nacional, además de pasar a un ambiente moderno y con MS SQL como Base de Datos profesional, ha mejorado significativamente y se ha hecho más amigable, logrando ir desplazando poco a poco la versión de Interhotel como el Sistema de Gestión Hotelera de mayor presencia en las instalaciones hoteleras de Cuba. Aunque con insuficiencias el carácter nacional de Zun y su integración con el Back Office¹⁴, son elementos a su favor en la batalla con los otros SGH presentes en mercado cubano [Salgado05].

1.8.3.3 *DRIM (Tecnomática)*

El sistema (DRIM) constituye la primera versión de la empresa en este ámbito.

DRIM es un sistema para la gestión de almacén. Sus opciones facilitan el tratamiento de la documentación propia de la actividad de los almacenes.

Es importante resaltar que DRIM no tiene vínculo o gestión alguna con la contabilidad, es decir en el sistema no se contabilizan las operaciones del almacén ni se prepara la información para que sea tramitada por un sistema de contabilidad [Pardo04].

1.9 *Conclusiones Parciales*

La industria del software patentiza la idea de mejorar sus procesos en aras de lograr productos de calidad sobre la base de la utilización de metodologías de desarrollo que guíen y soporten la construcción de software, a lo largo de todo su ciclo evolutivo.

Existen problemas asociados a la gestión de los requisitos en el ámbito organizacional y de proyecto, acarreando resultados diferentes a lo esperado. La sustracción de los mismos tiene como fundamento la adecuada IR, el establecimiento organizado e iterativo de todos sus procesos.

El éxito en los requisitos de proyectos anteriores se repite en los nuevos mediante la idea de la reutilización de requisitos y del uso de patrones de requisitos. Las herramientas Case incrementan y facilitan los resultados óptimos en el desarrollo de los diferentes procesos que abarca la IS en la construcción de software de calidad.

-
- ¹⁴ Término en español “la oficina de atrás” para referir aquello que el cliente no ve. Por ejemplo la Contabilidad y finanzas, Emisión de talones / pagarés CxC (Cuentas x Cobrar), Medios Básicos, etc.

La gestión de los inventarios impacta la inversión en capital de trabajo, afectan la fidelidad de los clientes e influyen, positiva o negativamente, en los costos operativos. Constituyen una parte esencial en el buen comportamiento económico de las empresas.

La gestión de los inventarios de almacén en el país requiere de una estandarización respecto a su manejo. La utilización de sistemas de GIA de almacén influye en la eficacia de este tratamiento.

Los sistemas de gestión de inventario existentes están centralizados por sectores o empresas. Es una necesidad la estandarización y uso de un sistema de GIA para todas las empresas.

CAPÍTULO II. DESCRIPCIÓN DE LA SOLUCIÓN PARA EL MÓDULO NOMENCLADORES DE SIGIA

2.1 Introducción

En este capítulo se define la solución al problema planteado, basado en las actividades delimitadas en el campo de acción del objeto de estudio de la investigación. Se sustenta en la fundamentación teórica que centra sus estudios en la IR y en su vinculación con los procesos de la GIA. Partiendo de estos últimos se obtiene la modelación del negocio, que constituye el punto de inicio de las actividades que culminan con la especificación de los requisitos de software y la modelación del sistema para el conjunto de Nomencladores de SIGIA.

2.2 Elicitación de Requisitos

El manejo eficiente de los requerimientos influye de forma determinante en el éxito de los proyectos software; identificando los problemas, enfocando la solución al uso exhaustivo de la tecnología y esta a su vez que se adecue, con mayor precisión, a las necesidades de los interesados.

Captar las necesidades que debe cubrir el sistema que se implementa y ajustarlas a un tipo de software; su frecuencia de utilización diaria y las condiciones en las que debe funcionar para explotar el máximo sus potencialidades, no son tareas fáciles de lograr.

2.2.1 Descripción de las técnicas utilizadas para el caso de estudio

El procedimiento empleado para el estudio de los procesos de GIA tuvo como sustento la revisión de documentos y las entrevistas a los clientes. En un principio estaba concebido que el sistema de gestión abarcaría lo referente al sector del turismo solamente, por la necesidad e importancia para la GIA se decidió extender al resto de las entidades y sectores del país.

Inicialmente la ERE fue centrada en este sector y por tanto las actividades de la misma se adaptarían a sus intereses exclusivamente¹⁵. Las empresas interesadas proporcionaron la documentación que disponían para la preparación en los temas de interés (Manuales de Normas y Procedimientos, Legislaciones, el sistema IHMM2000, Manuales de contabilidad, por mencionar algunos).

¹⁵ La visión de los requisitos del cliente del Turismo no traspasaba el marco de este sector.

Desde aquí, se pueden identificar dos etapas fundamentales en esta captura de requisitos (inicial).¹⁶

Mediante el análisis de la bibliografía y como guía, la experiencia de miembros del equipo de analistas, de conjunto se elaboraron documentos (no formales) con los procesos que meritaban entrevista con el cliente para su esclarecimiento. Esto constituye la primera etapa de la captura de requisitos (inicial).

La segunda etapa comenzó con la organización de entrevistas con el cliente para el mejor entendimiento de los procesos, mediante visitas por parte del equipo a las entidades correspondientes. Todo esto con la previa base del estudio de la documentación.

Al extender los límites y el alcance del nuevo sistema y por la similitud de los procesos, la estrategia de captura de requisitos fue reestructurada pero sobre la base de los requisitos encontrados, documentados y aprobados anteriormente. Se coordinaron entrevistas con los nuevos clientes y se validaron los requisitos obtenidos con anterioridad para que fueran aprobados por estos e incluidos aquellos que por el nuevo alcance debían ser agregados.

2.2.2 Modelación del Negocio para los procesos de GIA.

Uno de los aspectos fundamentales en el desarrollo de software es el entendimiento de los procesos de la entidad a la que se pretende automatizar *“Una buena comprensión de los procesos del negocio es importante para construir los sistemas correctamente”* [Rational 2003].

En otras palabras, antes de automatizar un proceso es necesario comprender correctamente la estructura y funcionamiento de sus partes en el negocio. Un adecuado entendimiento de los procesos del negocio influye positivamente en la comprensión y modelación de los mismos así como la vinculación de los desarrolladores con el vocabulario y términos utilizados por el cliente. Todo en virtud de la familiarización de analistas y desarrolladores con los términos del negocio. La forma para representar estos procesos en el negocio es mediante el Modelo del Negocio (MN).

El estudio de los procesos de GIA en el país arrojó que están bien delimitados unos de otros, se pueden encontrar fácilmente las personas implicadas dentro de cada uno, así como aquellos beneficiados con su resultado, quienes los inicializan y además quienes son las personas que desarrollan las actividades en cada uno de estos procesos. La GIA no es nueva, viene desarrollándose desde hace mucho tiempo y perfeccionándose según la evolución del país y de las empresas.

¹⁶ De ahora en lo adelante inicial para diferenciarla de los requisitos actuales.

Por ende se ha determinado la realización de un Modelo de Negocio capaz de abarcar los procesos de GIA y ser la piedra angular para el desarrollo de las actividades comprendidas dentro del flujo de trabajo de los requerimientos, que propone la metodología seleccionada. El MN es resultado de un conjunto de artefactos que lo complementan como los casos de uso del negocio y sus relaciones con los actores, reglas del negocio, entre otros.

2.2.2.1 Definición de casos de uso y actores del negocio

Con la finalidad de entender y delimitar el alcance de cada uno de los procesos de la GIA en el país; se elabora el MN. En el MN los procesos se identifican como casos de uso del negocio (CUN), generalmente por una relación de 1:1. Cada CUN agrupa una serie de actividades inicializadas por un actor y que responden a un proceso. Actores y casos de uso relacionados correspondientemente satisfacen a procesos del negocio debidamente fundamentados por el analista (Figura 5 Diagrama de casos de uso del negocio).

Un actor del negocio se define como la persona que se beneficia con el proceso del negocio que le corresponde. Responde a un rol dentro del proceso del negocio. Ejemplo: los clientes, los proveedores, las autoridades (legales, reguladoras, y otros), los socios, etc.

Actor	Descripción
Responsable de sección.	Persona facultada para solicitar productos al almacén u otra sección.
Proveedor.	Entidad externa quien provee al almacén de productos previamente pedidos por el jefe de compra.
Cliente.	Son aquellos a quienes el almacén les realiza ventas. No son parte de la entidad.
Personal de Dirección.	Responsable de aprobar las compras solicitadas y de las ventas de productos a terceros. Solicita información al almacén para consolidar.

Tabla 1 Actores del negocio para SIGIA

En el MN de GIA presentado, están involucradas de forma general, las entradas y salidas de productos al almacén, cada uno de ellos con su contrapartida contable respectiva. La descripción de los casos de uso de forma particular, se encuentra en el epígrafe del mismo nombre 2.2.2.3.

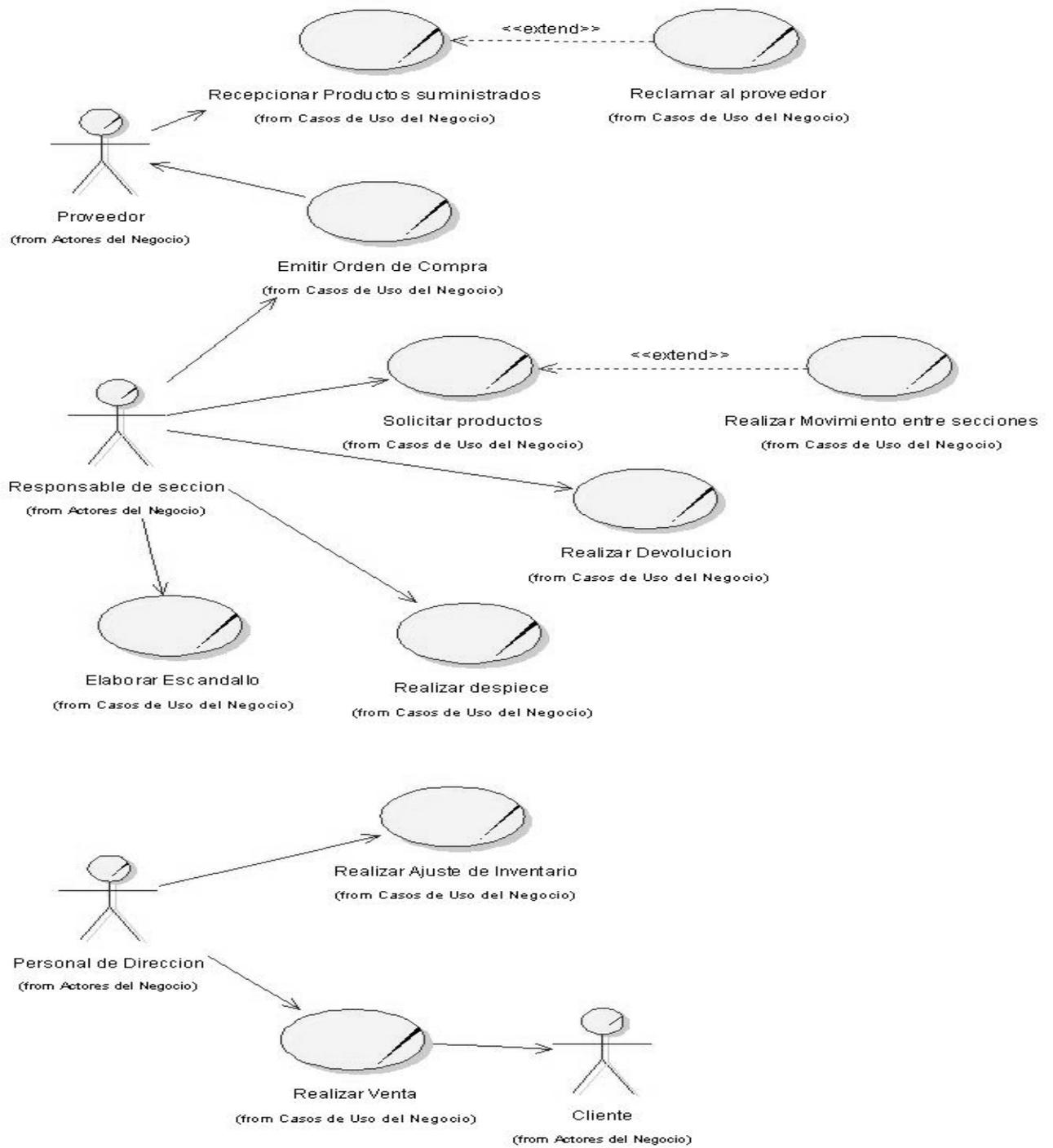


Figura 5 Diagrama de casos de uso del negocio

2.2.2.2 Trabajadores y entidades del negocio

Los trabajadores del negocio son los que llevan a cabo las actividades que conforman el proceso del negocio. Posteriormente pueden pasar a ser los actores del sistema.

Trabajador	Descripción
Almacenero.	Responsable de darle entrada y salida a los productos en el almacén.
Jefe de Compra.	Encargado de gestionar la compra de productos y emitir ordenes de compra al proveedor.
Normador.	Encargado de definir las fichas técnicas además de hacer los pedidos al almacén de los productos para realizar el escandallo o despieces.
Contador.	Empleado del departamento de economía que atiende el almacén, llevando a cabo la contrapartida contable de cada una de las operaciones realizadas en el almacén y sus secciones. Emite los comprobantes contables.

Tabla 2 Trabajadores del negocio para SIGIA

Las entidades del negocio son objetos físicos que controlan la información que perdura en él. Los trabajadores son los encargados de manipularlas durante la ejecución de las actividades que tienen lugar al desarrollar los procesos del negocio, para satisfacer las necesidades de los actores. La relación trabajadores-entidades de SIGIA se muestra en (Figura 6 Modelo de Objetos del Negocio de SIGIA.).

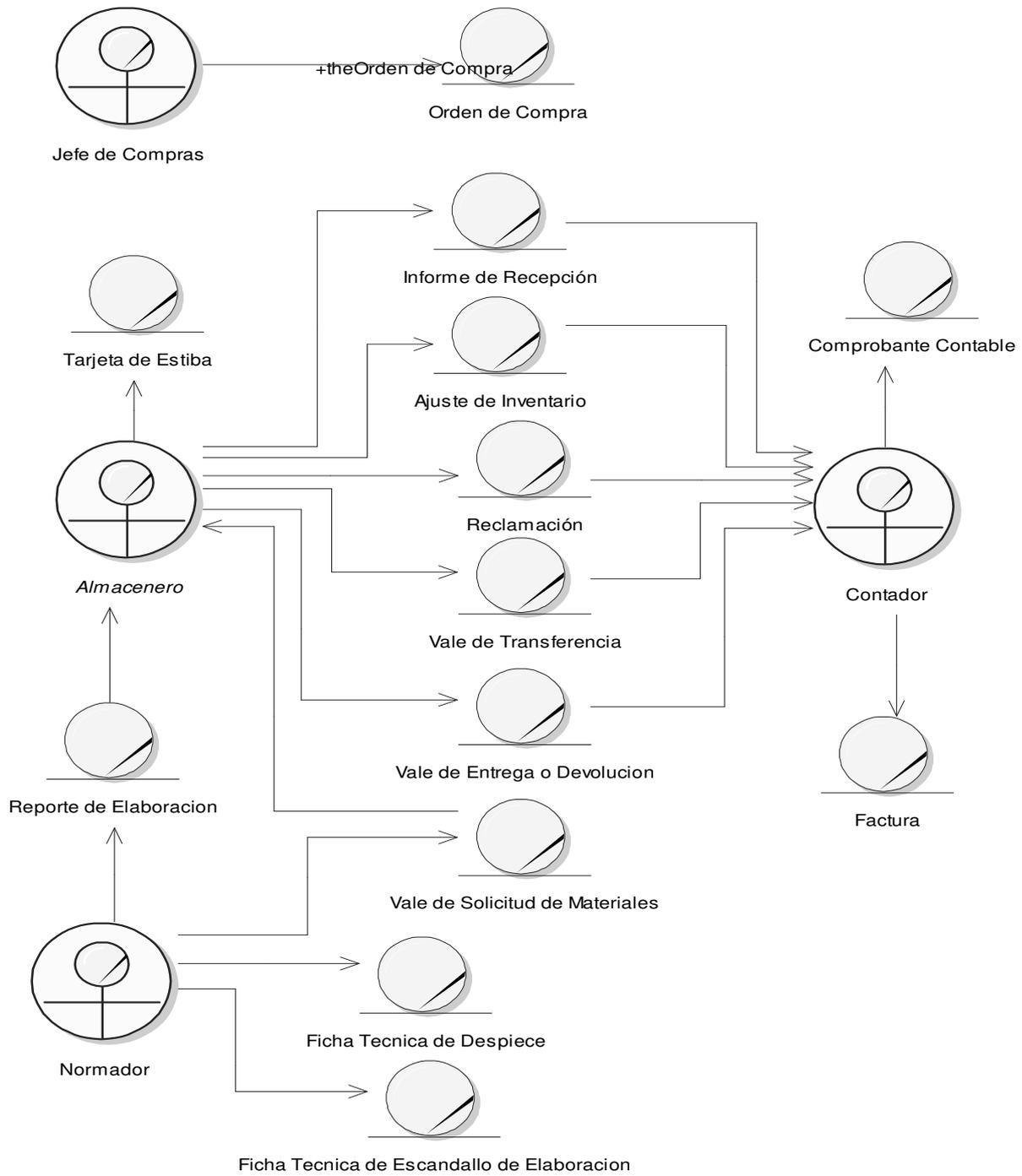


Figura 6 Modelo de Objetos del Negocio de SIGIA.

2.2.2.3 Descripción de los casos de uso del negocio

Caso de Uso:	Emitir orden de compra.
Actores:	Responsable de sección.
Trabajadores:	Jefe de compras.
Resumen:	El caso de uso se inicia cuando el responsable de determinada sección o del propio almacén envían solicitudes de compra de productos al Jefe de Compra, este analiza las solicitudes y decide emitir una Orden de Compra, la cual deberá ser aprobada por el Dpto. de Economía y luego enviada al Proveedor.

Caso de Uso:	Recepcionar productos suministrados.
Actores:	Proveedor (inicia).
Trabajadores:	Almacenero, contador.
Resumen:	El caso de uso se inicia cuando el proveedor entrega los productos al almacén, el almacenero verifica la cantidad y calidad de los mismos, si todo esta en orden se elabora el Informe de Recepción y actualizan las tarjetas de estiba, de existir algún problema se efectúa una reclamación o se revierten las operaciones.

Caso de Uso:	Reclamar al proveedor.
Actores:	Proveedor.
Trabajadores:	Almacenero, contador.
Resumen:	El caso de uso se inicia cuando los productos suministrados por el proveedor no se corresponden con las cantidades en la factura o su calidad no es la requerida y se procede a reclamar al proveedor.

Caso de Uso:	Solicitar productos.
Actores:	Responsable de sección.
Trabajadores:	Almacenero, contador.
Resumen:	El Responsable de la sección destino envía al almacenero un vale de pedido de los productos que necesita y este si no hay suficientes en el almacén remite el pedido a una sección que pueda satisfacerla.

Caso de Uso:	Realizar movimiento entre secciones.
Actores:	Responsable de sección.
Trabajadores:	Contador.
Resumen:	El Responsable de la sección destino envía al almacenero un vale de pedido de los productos que necesita y este si no hay suficientes en el almacén remite el pedido a una sección que pueda satisfacerla.

Caso de Uso:	Realizar devolución.
Actores:	Responsable de sección
Trabajadores:	Almacenero, Contador.
Resumen:	El caso de uso se inicia cuando algún Responsable de sección desea devolver al almacén por algún motivo, productos que previamente sustrajo del mismo. El almacenero recibe los productos confecciona los documentos que avalen la operación y actualiza la Tarjeta de estiba.

Caso de Uso:	Elaborar escandallo.
Actores:	Responsable de sección.
Trabajadores:	Normador, almacenero, contador
Resumen:	El CU se inicia cuando el Responsable de sección (de elaboración) necesita elaborar un producto para ofertar a los clientes e informa al normador el producto a elaborar y la cantidad, este solicita al almacén los productos necesarios para realizar la elaboración a partir de la ficha técnica predefinida, se efectúa la elaboración y se distribuyen por las secciones correspondientes, actualizando la existencias del nuevo producto elaborado en las mismas.

Caso de Uso:	Realizar despiece.
Actores:	Responsable de sección
Trabajadores:	Normador, almacenero, contador.
Resumen:	El CU se inicia cuando el Responsable de Sección (autorizada a realizar despieces) solicita productos para la realización de un despiece. El normador realiza la solicitud al

	almacén, se efectúa el despiece en el área autorizada para ello, se realiza un control de los productos resultantes y se distribuyen los productos a las secciones correspondientes y se actualizan las existencias del producto en las secciones involucradas.
--	---

Caso de Uso:	Realizar ajuste de inventario.
Actores:	Personal de dirección
Trabajadores:	Almacenero, contador
Resumen:	El caso de uso se inicia cuando se detectan anomalías (Faltantes, mermas, roturas, deterioros, etc.) en los productos del almacén por controles realizados ya sea por el Personal de Economía o por el propio Almacenero. Se autoriza por el personal de dirección ajustar el inventario, llevar a la cuenta de gasto el por ciento establecido y aplicar las medidas disciplinarias correspondientes en caso necesario.

Caso de Uso:	Realizar venta.
Actores:	Cliente, personal de dirección.
Trabajadores:	Almacenero, contador.
Resumen:	El caso de uso se inicia cuando algún autorizado del personal de dirección autoriza una venta a un tercero (otra entidad), informa al almacenero la relación de los productos, este despacha los mismos al cliente, los rebaja de su inventario y el contador valida las operaciones contables correspondientes.

Los diagramas de actividad muestran el flujo de las operaciones que responden a los procesos del negocio correspondientes y complementan las descripciones previas. Se elaboró uno por cada caso de uso del negocio viabilizando así la comprensión de los casos de uso modelados (Anexo III **Diagramas de Actividad de los Casos de Uso del Negocio**).

El glosario de términos es otro de los artefactos que se obtiene durante la modelación del negocio. Sirve de complemento para establecer los conceptos que se manejan dentro de la organización. Puede ser redactado o no en dependencia de la determinación del analista; es posible que sea utilizado más tarde durante el proceso de ingeniería de software para agregarle los términos introducidos en la modelación del sistema.

La GIA maneja conceptos que son nuevos para el equipo de desarrollo, razón por la cual se acordó confeccionar el antes mencionado documento de conjunto con el glosario de la investigación (Ver Glosario de Términos).

2.2.2.4 Reglas del negocio

Los procesos del negocio se encuentran sujetos a un conjunto de reglas, que determinan las políticas y la estructura de la información. Una regla es un precepto, principio o máxima. Razón a que se han de ajustar las acciones para que resulten rectas.¹⁷

Una Regla del Negocio es una declaración de política o una condición que debe ser satisfecha dentro de la organización y que la misma no puede ser violada por ninguno de los trabajadores o directivos, en dependencia de los niveles de restricción que se presente en cada proceso.

Se determinó, en vistas a una rápida localización y mayor organización en la redacción de las reglas, confeccionar un identificador estructurado de la siguiente forma: **<RN>.<CU responde dicha regla> ó <G en caso de ser regla general>.<# regla negocio>**.

Emitir Orden de compra (OC).

RN.OC.1 La solicitud de mercancía puede ser realizada únicamente por el jefe de almacén o cualquier responsable de sección.

RN.OC.2 En el caso de compras no habituales se reúne un comité de compras para la aprobación de de las mismas.

RN.OC.3 La única persona que emite la solicitud de productos al proveedor es el Jefe de compras.

RN.OC.4 Tener en cuenta las características de la entidad para emitir las órdenes de compra.

Recepcionar Productos Suministrados (PS).

RN.PS.1 Se debe efectuar el método de recepción a ciegas en la recepción de productos.

RN.PS.2 El Informe de Recepción (IRec) debe reflejar los productos recepcionados teniendo en cuenta la regla anterior.

RN.PS.3 La recepción de productos puede estar amparada factura o conduce.

¹⁷ Concepto según DRA Diccionario de la Real Academia de la Lengua.

RN.PS.4 No es obligatoria la presentación de la OC por parte del Proveedor, en caso de que la traiga pues se agregará su número al IRec para corroborar el pedido.

RN.PS.5 Se identifica cada producto almacenado con una Tarjeta de Estiba (TE).

RN.PS.6 Las TE no pueden contener tachaduras y es obligatoria la firma de la persona que recibe los productos.

RN.PS.7 En caso de no efectuarse una reclamación de un producto en el momento, se podrá realizar 90 días a partir del momento en que se recibe el producto o en el plazo acordado entre las partes involucradas (ver **CU**: Reclamar al Proveedor).

Reclamar al Proveedor (RP).

RN.RP.1 El plazo establecido para efectuar una reclamación debe estar convenido previamente entre el proveedor y la entidad.

RN.RP.2 El plazo en días para efectuar la reclamación es la diferencia entre el día elegido para efectuarla y la fecha del IRec.

RN.RP.3 Solo podrán efectuar reclamaciones de productos que ya fueron entrados al almacén en caso de vicios ocultos.

Solicitar Productos (SP).

RN.SP.1 La sección destino no puede ser una sección de gasto.

RN.SP.2 La fecha de un nuevo pedido de movimiento, no podrá ser menor a la fecha de la última operación efectuada por el almacén ni mayor a los siete días posteriores a los días que se definan en la entidad.

RN.SP.3 El Vale de Solicitud de Materiales (VSM) debe contener la firma del funcionario autorizado de acuerdo con las regulaciones establecidas, a los efectos de su validez.

RN.SP.4 Deben existir muestras de las firmas de los funcionarios autorizados a firmar la solicitud de movimiento y contratos de suministros, las que deben obrar en poder del área de Contabilidad para posibilitar su comparación con la firma de los mencionados documentos.

RN.SP.5 Las anotaciones en la TE deben realizarse inmediatamente después de haber efectuado el movimiento físico de los productos.

RN.SP.6 Confeccionar siempre el modelo de Solicitud de Materiales para efectuar el pre-despacho de los productos y este se archive como soporte del Vale de Salida (VS) o Factura. (Original en el almacén y copia al departamento Económico); cuidando de reflejar en ambos modelos la referencia cruzada.

RN.SP.7 Las reversiones de movimiento requieren de una autorización.

Realizar Movimiento entre secciones (MS).

RN.MS.1 Los productos pertenecientes a Grupos de Insumo sólo podrán ser solicitados al Almacén Central.

RN.MS.2 Las anotaciones en la TE deben realizarse inmediatamente después de haber efectuado el movimiento físico de los productos.

RN.MS.3 Confeccionar siempre el modelo de Solicitud de Materiales para efectuar el pre-despacho de los productos y este se archive como soporte del Vale de Salida (original en el sección origen y copia al departamento Económico); cuidando de reflejar en ambos modelos la referencia cruzada.

Realizar Devolución (RD).

RN.RD.1 El Vale de Entrega o Devolución (VED), en este caso de devolución, debe tener la referencia al VED de salida por donde salieron del almacén los productos a devolver.

RN.RD.2 La cantidad de productos a devolver no debe ser mayor a la cantidad de productos reflejada en el VED de salida.

Elaborar Escandallo (EE).

RN.EE.1 No se pueden solicitar productos para elaborar escandallos desde la sección de almacén central o desde una sección de gasto, solo de las secciones definidas como centros de elaboración.

RN.EE.2 Se deberá reflejar en las Fichas Técnicas (FT) posibles productos sustitutos.

RN.EE.3 La FT debe contener un identificador para cada producto.

RN.EE.4 Se podrán definir tantos sub.-Escandallos dentro de una FT como se deseen, teniendo como única premisa que primero deben definirse las FT de los sub.-Escandallos y luego el producto que los utilizará.

Realizar Despiece (D)

RN.D.1 No se permitirá definir ficha de despiece a un producto que pertenezca a un Grupo de Útiles o que pertenezca a un Grupo de Insumo.

RN.D.4 El producto a despiezar y los productos resultantes no pueden pertenecer a un Grupo/Familia de diferentes monedas.

Realizar Ajuste de Inventario (AI)

RN.AI.1 Un ajuste de inventario se hará por un concepto de ajuste a la vez.

RN.AI.2 En caso que las mermas, roturas o los deterioros de alguno de los productos sean detectados por el responsable de sección debe informar rápidamente de la situación al departamento de economía para que allí se le de una solución al problema.

RN.AI.3 El director de economía es el único que da la autorización de pasar cualquier producto a la cuenta de gastos.

RN.AI.4 Los inventarios realizados en el almacén deben ser supervisados por el empleado de contabilidad.

RN.AI.5 En caso que la merma sea por descongelación se debe pesar el producto.

RN.AI.6 En caso que los que detecten los faltantes sea algún trabajador del negocio estos deben informar de inmediato al director de economía para que este tome las medidas pertinentes.

RN.AI.7 En caso de emitirse acta de responsabilidad, esta debe estar en manos del director de contabilidad.

RN.AI.8 Deben hacerse los expedientes correspondientes en caso de que exista algún faltante en la entidad.

RN.AI.9 Cada vez que se transfiere un producto a la cuenta de gastos se debe dar baja del inventario y el almacenero debe actualizar la TE.

RN.AI.10 Si los productos son ajustados por roturas o deterioro se deben controlar su existencia aun después de ajustados.

RN.AI.11 Los ajustes de inventarios resultantes de los conteos físicos, sean amparados mediante el correspondiente expediente, después de haber sido aprobados por los niveles establecidos. Se dejará evidencia en el almacén de las actas que las Comisiones de Mermas, Roturas y Deterioros o de Destino Final, confeccionen y aprueben.

Realizar Venta (RV)

RN.RV.1 Los productos involucrados en las ventas deberán ser productos estancados o con lento movimiento para la entidad que vende o sin utilidad para la misma.

RN.RV.2 Las ventas se realizaran con previo acuerdo entre las partes involucradas con un incremento o decremento sobre el precio costo de los productos involucrados.

Reglas generales del negocio

RN.G.1 Se deben dominar los niveles de normas de inventario establecidos para cada producto y fiscalizarlos sistemáticamente, así como, no tener productos ociosos o de lento movimiento.

RN.G.2 Realizar comprobaciones sistemáticas a los almacenes y establecer vínculos estrechos con el MINCIN para la aplicación del Programa de Economía de Almacenes.

RN.G.3 Mantener una estrecha vinculación con el área comercial y de explotación para que las compras se correspondan con el Plan y estrategias de Mercadotecnia establecido en la Empresa, en cuanto a gustos de los clientes se refiere. Asesora y controla en este sentido a las UEB Y Complejos.

RN.G.4 La Sección de Carnicería debe subordinarse al área del almacén de la Unidad con el fin de poder ejercer el control sobre el producto, sus mermas, desperdicios y los costos del mismo, hasta su consumo final.

RN.G.5 Controlar lo estipulado en cuanto a stock máximos y mínimos y las rotaciones de los productos.

RN.G.6 Los Contratos de suministros deben contener la firma del funcionario autorizado, de acuerdo con las regulaciones establecidas, a los efectos de su validez y que existan muestras de estas firmas en poder del área económica para posibilitar su comparación con las firmas del mencionado documento.

RN.G.8 No deben coincidir en los mismos funcionarios la facultad de firmar Contratos de Suministros, con la de firmar Informes de Recepción, Vales de Salidas u Orden de Despacho.

RN.G.7 Que el Informe de Recepción se elabore en cada área de almacenaje, sobre la base de las calidades y cantidades físicamente recibidas, independientemente de los datos que aparezcan en la Factura o Conduce.

RN.G.8 Los documentos primarios requieren contener la información de las existencias en unidades físicas, después de efectuada cada anotación en las TE.

RN.G.9 De existir más de un almacén o área de responsabilidad de almacenamiento, resulta necesario que la enumeración consecutiva de los documentos sea individual por área, identificando con siglas o códigos que antecedan a la numeración, el área o almacén a que corresponden.

RN.G.10 Que los productos solo sean despachados mediante los documentos de salida oficialmente establecidos por el Sistema Nacional de Contabilidad (SNC).

RN.G.11 Que el área donde se almacenan los productos cuente con una relación de los nombres, apellidos, cargos y muestras de las firmas de las personas facultadas para solicitar determinados tipos de productos, así como las facultades para aprobar las entregas y transferencias entre almacenes.

RN.G.12 Los productos en existencia se controlan en el almacén a través de la TE, las que se ubicarán junto a cada producto o en lugar cercano al mismo, donde sea más factible su cuidado y manipulación.

RN.G.13 Las anotaciones en las TE se realizan inmediatamente después de haberse efectuado el movimiento físico de los productos.

RN.G.14 Prohibir al personal que labora en los almacenes tenga acceso a los registros contables y a los submayores de inventario.

RN.G.15 La conciliación y cuadro de las existencias en unidades físicas y valores entre las TE y los Submayores de Inventarios, debe ser una tarea sistemática y permanente (minimizar el ciclo de conciliación).

RN.G.16 Que se controlen en el área de almacén los envases retornables, por TE.

RN.G.17 Los productos se roten de acuerdo al tiempo de almacenamiento de los mismos, recordando siempre el principio Primero que entra primero que sale.

RN.G.18 No deben existir documentos originados y destinados al área de almacén, con errores, tachaduras y enmiendas.

2.3 Conclusiones Parciales

De forma general la finalidad del modelado del negocio es describir cada proceso del negocio¹⁸, especificando sus datos, actividades (o tareas), roles (o agentes) y reglas de negocio. Este modelo especifica qué procesos del negocio soportará el sistema.

Todas estas actividades influirán de manera significativa durante todo el ciclo de vida de desarrollo del software y específicamente durante el flujo de trabajo de requerimientos.

Es importante destacar la necesidad de conocer, entender y manipular de forma fluida los términos del negocio que se estudia. Establecer lazos de afinidad con los mismos, en virtud de familiarizar a todos los miembros del equipo de desarrollo con todos estos temas de gestión y control de inventarios de almacén.

El modelo de negocio, junto a todos los artefactos que lo componen, permite a los usuarios, clientes, desarrolladores y otros obtener una visión del negocio y los procesos que son objeto de automatización. Es por lo que RUP ha propuesto los artefactos antes desarrollados que vienen a sustentar y ayudar además, a la gestión de los requisitos; tema que se presentará en los próximos epígrafes.

¹⁸ En este caso el negocio de los inventarios de almacén en Cuba.

2.4 Especificación, Análisis y Negociación de requisitos para el módulo Nomencladores de SIGIA

Producto a la ERE (inicial) se obtuvo una Especificación de Requisitos (ER) también inicial que respondían a las necesidades de los anteriores clientes. Se redactaron en un lenguaje natural y comprensible para todo el equipo pero sin una plantilla formal.

Ante las expectativas de los nuevos clientes se hace un análisis de la ER existente y sobre esta base se trabajó en la ER actual ampliando las expectativas. Se efectuaron un grupo de reuniones internas y no formales por el grupo de analistas para examinar los factores completitud, ambigüedad, complejidad y otros de dicha ER, concluyendo finalmente con la negociación de los resultados con el cliente. Como resultado palpable se generaron nuevos requisitos y refinaron los anteriores sistemáticamente, tanto funcionales como no funcionales, para el módulo Nomencladores de SIGIA.

2.4.1 Requisitos Funcionales

Los RF referentes al módulo que se estudia (Nomencladores) han sido agrupados según el patrón de requisitos, solo por cuestiones de organización, no afectando su funcionalidad y lo logrado en la etapa de negociación.

Los RF son los siguientes:

RF-3	<ul style="list-style-type: none">3. Gestionar Grupo/Familia.<ul style="list-style-type: none">3.1. Registrar Grupo de productos.<ul style="list-style-type: none">3.1.1. Código (2 dígitos).3.1.2. Nombre.3.1.3. Descripción.3.1.4. Estado (Activo o no).3.1.5. Registrar si es insumos.3.2. Registrar Familia de productos.<ul style="list-style-type: none">3.2.1. Código grupo (2 dígitos).3.2.2. Código Familia.3.2.3. Nombre.3.2.4. Descripción.3.2.5. Estado (activo o no).3.2.6. Registrar si es un útil.<ul style="list-style-type: none">3.2.6.1. Por ciento de desgaste en caso de los útiles.3.3. Modificar información de Grupo/Familia de productos.<ul style="list-style-type: none">3.3.1. Verificar dependencias en caso de inactivar el G/F (que no existan productos pertenecientes a este G/F).3.4. Eliminar Grupo/Familia de productos.<ul style="list-style-type: none">3.4.1. Verificar que no existan productos pertenecientes a este G/F.
-------------	--

Seguimiento	CU Nomenclar Grupo/Familia de Productos
--------------------	---

RF-7	<p>7. Gestionar clasificación de sección.</p> <p>7.1. Registrar clasificación de sección</p> <p>7.1.1. Nombre clasificación (central de insumos, central de mantenimiento, central de comestibles, elaboración, carnicería, punto de venta, punto de servicio, gasto, etc.).</p> <p>7.1.2. Código (se asigna automático, 3 dígitos).</p> <p>7.1.3. Descripción.</p> <p>7.1.4. Restricción G/F de productos que puede manejar (dar entrada, dar salida o ambos).</p> <p>7.1.5. Verificar que no coincidan conjugaciones completas de los parámetros para hacer efectivo el registro de una nueva clasificación (esto me garantiza no crear clasificaciones de sección con las mismas características lo cual carece de sentido).</p> <p>7.2. Modificar y/o eliminar clasificación de sección</p> <p>7.2.1. Verificar dependencias.</p> <p>7.2.2. Verificar que no esté activa.</p>
-------------	---

Seguimiento	Nomenclar Clasificación de Secciones
--------------------	--------------------------------------

RF-9	<p>9. Gestionar Sección de inventario.</p> <p>9.1. Registrar sección de inventario.</p> <p>9.1.1. Nombre de sección.</p> <p>9.1.2. Código (Asignar automático).</p> <p>9.1.3. Responsable de sección.</p> <p>9.1.4. Clasificación de sección (según definido en 7).</p> <p>9.1.5. Si se permite Validar Existencias.</p> <p>9.1.6. Si los números de referencia cruzada serán introducirán manuales o automáticos.</p> <p>9.1.7. Forma de valorar las entradas en la sección (Ponderado, FIFO, LIFO).</p> <p>9.1.8. Forma de generar los consecutivos de documentos.</p> <p>9.1.9. Stock máximo y mínimo de productos.</p> <p>9.2. Modificar Información de sección de inventario (todo menos la clasificación).</p> <p>9.3. Eliminar sección de inventario.</p> <p>9.3.1. Verificar que no tenga existencia de productos.</p>
-------------	---

Seguimiento	Nomenclar Secciones de Inventario
--------------------	-----------------------------------

RF-12	<p>12. Gestionar Enlaces Grupo/Familia-Sección-Cuenta contable</p> <p>12.1. Registrar enlace por Grupo/Familia-Sección-Cuenta contable.</p> <p>12.2. Modificar enlace por Grupo/Familia-Sección-Cuenta contable.</p> <p>12.3. Eliminar enlace por Grupo/Familia-Sección-Cuenta contable.</p> <p>12.3.1. Verificar que no hayan operaciones pendientes que afecten este enlace.</p>
--------------	---

Seguimiento	Nomenclar Enlaces Contables Grupo/Familia-Sección
--------------------	---

RF-13	<p>13. Gestionar Unidad de Medida.</p> <p>13.1. Registrar Unidades de Medida con las que se trabajará.</p> <p>13.1.1. Nombre.</p> <p>13.1.2. Código.</p> <p>13.1.3. Métrica (real o entero).</p> <p>13.1.4. Descripción.</p> <p>13.2. Modificar Unidad de Medida.</p> <p>13.3. Eliminar Unidad de Medida.</p> <p>13.3.1. Verificar que no esté en uso.</p>
--------------	---

Seguimiento	Nomenclar Unidades de Medida
--------------------	------------------------------

RF-16	<p>16. Gestionar Especialidad de Proveedor.</p> <p>16.1. Registrar Especialidad de Proveedores.</p> <p>16.1.1. Nombre de Especialidad.</p> <p>16.1.2. Código de Especialidad.</p> <p>16.1.3. Enlazar con Grupo/Familia de productos a los que responde.</p> <p>16.2. Modificar Especialidad de Proveedores.</p> <p>16.3. Eliminar Especialidad de Proveedores.</p>
--------------	--

Seguimiento	Nomenclar Especialidad de Proveedor
--------------------	-------------------------------------

RF-15	<p>15. Gestionar Equivalencias de Unidad de medida.</p> <p>15.1. Registrar equivalencias entre unidades de medidas para los productos que lo requieran.</p> <p>15.1.1. Código unidad menor.</p> <p>15.1.2. Código unidad mayor.</p> <p>15.1.3. Cantidad (UMmayor/UMmenor, según métrica definida).</p> <p>15.2. Modificar equivalencias entre unidades de medidas.</p> <p>15.2.1. Verificar que el producto no se encuentre en uso.</p>
--------------	--

Seguimiento	Registrar Equivalencias de Unidad de Medida de Productos
--------------------	--

RF-19	<p>19. Gestionar información de Producto.</p> <p>19.1. Registrar información de Productos.</p> <p>19.1.1. Código.</p> <p>19.1.2. Grupo/Familia.</p> <p>19.1.3. Nombre.</p> <p>19.1.4. Descripción.</p> <p>19.1.5. Tipo (elaborado, corriente)</p> <p>19.1.6. Unidad de Medida base</p> <p>19.2. Modificar información de Productos.</p> <p>19.2.1. Verificar dependencias</p> <p>19.3. Eliminar Productos.</p>
--------------	--

	19.3.1. Verificar dependencias
Seguimiento	Nomenclar Productos

RF-22	22. Gestionar información de Proveedor 22.1. Registrar información de Proveedores. 22.1.1. Código 22.1.2. Especialidades. 22.1.3. Dirección. 22.1.4. Teléfonos. 22.1.5. Correo electrónico. 22.1.6. Monedas con las que operará. 22.1.7. Enlace a cuentas contables con las que operará. 22.1.8. Activo. 22.1.9. Precio de venta. (el actor lo entra) 22.2. Modificar información de Proveedores. 22.3. Eliminar Proveedores 22.3.1. Verificar dependencias.
Seguimiento	Nomenclar Proveedor

RF-25	18. Gestionar información de Clientes. 18.1. Registrar información de Clientes. 18.1.1. Código. 18.1.2. Nombre. 18.1.3. Dirección. 18.1.4. Teléfonos. 18.1.5. Correo electrónico. 18.1.6. Monedas con las que operará. 18.1.7. Enlace a cuentas contables con las que operará. 18.2. Modificar información de clientes. 18.3. Eliminar Clientes.
Seguimiento	Nomenclar Clientes Terceros

2.4.2 Requisitos no funcionales.

De los requisitos capturados se definieron de conjunto con el Arquitecto de Software los siguientes RNF.

Requerimientos de Hardware

Estaciones de Trabajo

1. Tener periféricos Mouse y Teclado.
2. Tarjeta de red.
3. 64 Mb de memoria RAM.

Servidor Local

1. Tener periféricos Mouse y Teclado.
2. 1Mb de cache L2, 256 Mb de memoria RAM.
3. 2 GB de espacio libre en disco.
4. Tarjeta de red.

Requerimientos de Software

Estaciones de Trabajo

1. Sistema Operativo: Windows 98 o superior, Linux, Unix.
2. Java Virtual Machine (Máquina Virtual de Java) versión 1.6

Servidor Local

1. Sistema Operativo: Windows 98 o superior, Linux, Unix.
2. Gestor de Base de Datos: SQL Server 2000
3. Máquina Virtual de Java versión 1.6

Redes

1. La red existente en las instalaciones debe soportar la transacción de paquetes de información de al menos unas 10 máquinas a la vez.
2. Para hacer más fiable la aplicación debe de estar protegida contra fallos de corriente y de conectividad, para lo que se deberá parametrizar los tiempos para realizar copias de seguridad. Implementar las transacciones de paquetes en la red con el protocolo TCP/ IP que permite la recuperación de los datos.

Seguridad

1. La seguridad se tratará desde las primeras fases de desarrollo del sistema.
2. Se utilizará las reglas de la “programación segura”, se deberá hacer un fuerte tratamiento de excepciones.
3. Parte de la seguridad corre por parte de la tecnología Java utilizada para el desarrollo de la aplicación.
4. Se debe garantizar la seguridad de la Base de Datos, no se permitirá que nadie tenga privilegios de administración sobre la base de datos.
5. Programación de disparadores (Triggers) en la Base de Datos para no permitir la manipulación de los datos directamente en el SGBD.

6. Se deberá utilizar usuarios de base de datos con roles bien definidos para cada una de las operaciones del sistema.
7. La asignación de usuarios y sus opciones sobre el sistema se garantizará desde el módulo de Administración del sistema.
8. Debe quedar constancia de quién, desde donde, y cuando se realizó una operación determinada en el sistema.

Portabilidad, escalabilidad, reusabilidad

1. El sistema debe poder ser utilizado desde cualquier plataforma de software (Sistema Operativo).
2. El sistema debe hacer un uso racional de los recursos de hardware de la máquina, sobre todo en las PC clientes.
3. Debido a los cambios en las condiciones económicas del país, las empresas cubanas toman decisiones continuas que cambian las condiciones en que se desarrollan los procesos, por lo que el sistema debe implementar la forma adaptarse ante el cambio de dichas condiciones.
4. Se implementará la forma de parametrizar elementos importantes para la funcionalidad del sistema.
5. Debe implementar servicios que estén a la escucha del pedido de información de otros sistemas que en el futuro requerirán información.
6. Se desarrollará cada pieza del sistema en forma de componentes (elementos) con el fin de reutilizarlos para futuras versiones del sistema.
7. La documentación de la arquitectura debe ser reutilizable para poder documentarla como una familia de productos.

Restricciones de acuerdo a la estrategia de diseño

1. El diseño de las aplicaciones se hará utilizando la Programación Orientada a Objetos (POO).
2. Se utilizará las tecnologías que brindan los frameworks definidos para cada una de las capas de la aplicación:
 - 2.2. Para la capa de presentación: Swing, aplicación de escritorio.
 - 2.3. Para la capa de lógica del negocio: los objetos del negocio, Framework Spring, la IoC y la programación Orientada a Aspectos.
 - 2.4. Para la capa de Acceso a Datos: Framework Hibérnate.

2.5 Modelado del sistema.

La modelación del sistema constituye una representación principalmente de los requisitos funcionales de software, donde se define la interrelación usuario-sistema como base a sus expectativas. Los artefactos actores, casos de uso y sus descripciones unido a los prototipos de interfaz conforman el Modelo del Sistema.

Los actores definidos para el módulo Nomencladores de SIGIA se muestran en (Tabla 3 **Actores del sistema para el módulo Nomencladores de SIGIA.**).

Actor	Descripción
Personal de Economía	Usuarios del sistema con privilegios de modificar la información de los nomencladores, así como solicitar los reportes de submayores de inventarios u otros necesarios.
Jefe de Compras.	Encargado de registrar en el sistema lo relativo a la gestión con proveedores, clasificar los productos en función de aumentar las rotaciones de los inventarios y lograr una mejor gestión de la compras.

Tabla 3 Actores del sistema para el módulo Nomencladores de SIGIA.

Por otro lado, los casos de uso son las funciones que proporciona un sistema para añadir valor a sus usuarios. Estos se han adoptado casi universalmente para la captura de requisitos de sistemas de software, sin embargo son más que simplemente una herramienta para la captura de requisitos; sino que dirigen todo el proceso de software [Jacobson00].

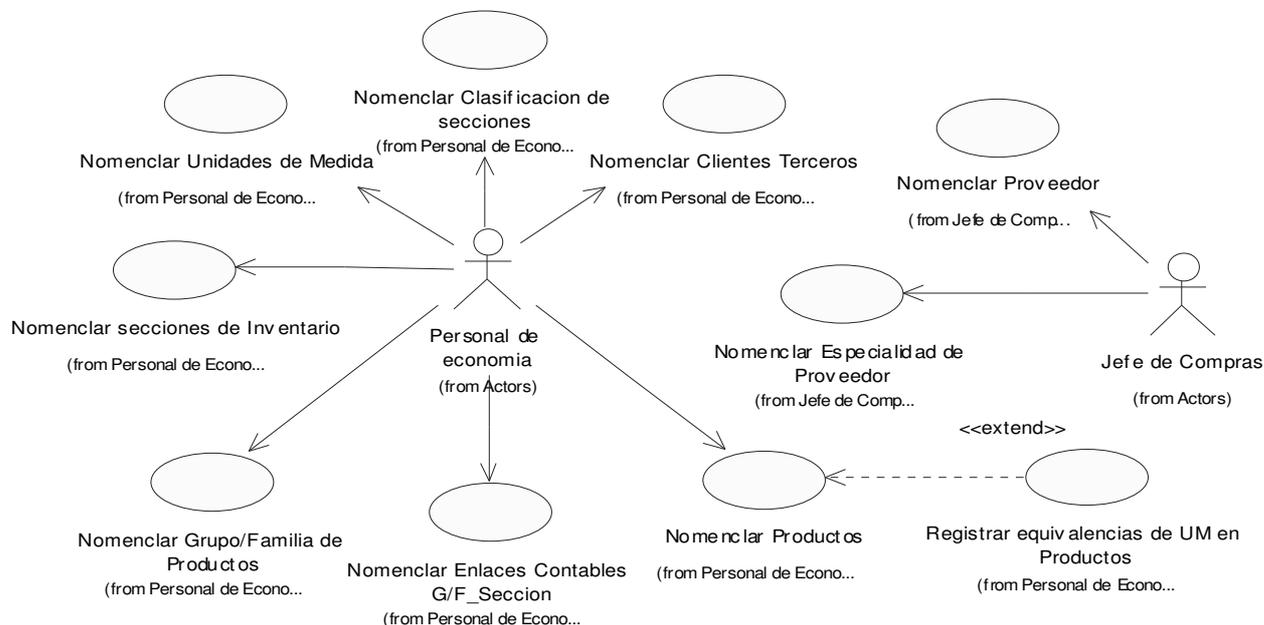


Figura 7 Diagrama de Casos de Uso del Módulo Nomencladores del SIGIA.

Los casos de uso del sistema constituyen una representación de los requerimientos funcionales y determinan mediante su encapsulamiento una funcionalidad específica del sistema. Los patrones de casos de uso constituyen una ayuda para la conformación de estos.

2.5.1 Patrones de Casos de Uso (PCU) promoviendo la reusabilidad de la experiencia.

Reutilización es el término que se le confiere al uso de cualquier tipo de artefacto, o parte del mismo, creado con anterioridad, en un nuevo proyecto [Reuse06].

Una de las ventajas que proporciona la reutilización es el hecho de que se requiere menor inversión inicial, así como que no se requieren cambios sustanciales en la aplicación.

La modelación de casos de uso no difiere de otros tipos de desarrollo en cuanto a que se reutilicen soluciones que se consideren eficientes. De ahí que las exitosas técnicas y diseños que han sido utilizados una y otras vez en los modelos de casos de uso, se formalizan como patrones expresando buenos diseños de modelos de casos de uso [Övergaard04].

Sin embargo es importante aclarar que un PCU no describe un uso particular del sistema, sino que captura una técnica para hacer sostenible, reusable y entendible el modelo [Övergaard04].

Uno de los PCU más utilizados es CRUD¹⁹. Este PCU describe un caso de uso para administrar la información, que abarca las operaciones de crearla, leerla, modificarla y eliminarla. Dicho caso de uso se relaciona con el usuario que maneja dicha información.

Teniendo en cuenta los requisitos funcionales descritos en el epígrafe anterior y utilizando la definición de este PCU se pueden obtener casos de uso para gestionar la información de Proveedores, Productos, Grupo/Familia de Productos, Secciones, Clasificación de Secciones, así como de Clientes Terceros, Equivalencias de Unidad de Medida y Enlaces Contables Grupo/Familia Sección. Todos estos conceptos están reflejados en los requisitos y sobre ellos se realizan acciones de registro, modificación y eliminación. La operación de mostrar la información de los mismos está encapsulada en el requisito 58: Mostrar reportes que faciliten el control de los inventarios. Dicho reporte debe ser parametrizable para que se muestre cualquier tipo de información y en cualquier formato.

En fin se han definido los siguientes casos de uso para el Módulo de Nomencladores del SIGIA a partir del PCU mencionado:

- Nomenclar Especialidad de Proveedor.
- Nomenclar Proveedores.
- Nomenclar Grupo/Familia de productos.
- Nomenclar Unidades de Medida.
- Nomenclar Productos.
- Nomenclar Enlaces Contables Grupo/Familia-Sección.
- Nomenclar Clasificación de Secciones.
- Nomenclar Secciones de Inventario.
- Nomenclar Clientes Terceros.

El otro de los casos de uso de este módulo sería:

- Registrar Equivalencias de Unidad de Medida de productos

Existen otros PCU que modelan la interacción entre casos de uso. Tal es el caso del PCU Extensión o Inclusión Concreto que trae dos especificaciones, una para la inclusión y otra para la extensión. Para el primer caso se proponen dos casos de uso relacionados con una relación de inclusión partiendo del caso

¹⁹ Referente a las siglas en inglés de las operaciones Crear, Leer, Modificar o Actualizar y Eliminar.

de uso base al incluido; el caso de uso base debe ser instanciado directamente y el incluido solo a partir del primero [Övergaard04].

En cuanto a la extensión, el caso de uso extendido es concreto por lo que puede ser instanciado directamente o realizarse a partir del caso de uso base. El caso de uso base puede ser abstracto o concreto [Övergaard04]. Esta variedad del PCU Extensión o Inclusión Concreto puede ser aplicada en el modelo del SIGIA, en el caso específico de los casos de uso Nomenclar Producto y Registrar Equivalencias de Unidad de Medida de Productos (**¡Error! No se encuentra el origen de la referencia.**). Esto se debe a que al registrar un producto es obligatorio Registrar la Unidad de Medida del mismo, pero para el resto de los escenarios que se muestran en este no resulta necesario. De esta forma el caso de uso base sería Nomenclar Productos y Registrar la Unidad de Medida sería el caso de uso extendido.

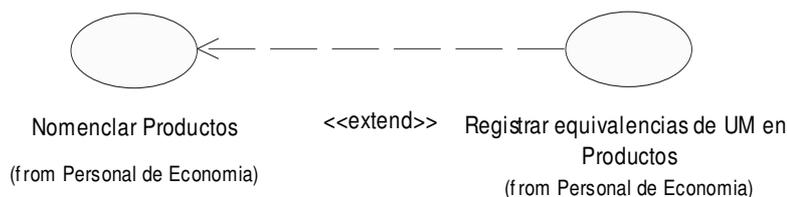


Figura 8 Extensión

2.5.2 Descripción de los casos de uso del módulo Nomencladores para el SIGIA.

Las descripciones detalladas y los prototipos de interfaz de los casos de uso del módulo Nomencladores pueden verse a continuación:

Caso de Uso:	Nomenclar Grupo/Familia de productos.	
Actores:	Personal de Economía	
Propósito	Crear, modificar o eliminar grupos y familias, así como Listar en pantalla los datos de los Grupo/Familia existentes.	
Precondiciones:	<ul style="list-style-type: none"> • El actor debe haberse autenticado correctamente. • Para crear una familia debe haberse creado su grupo previamente. 	
Referencias	RF-3	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “Registrar Grupo”		
Acción del Actor	Respuesta del Sistema	
1. Selecciona la opción Editar Grupo (Figura 9).	2. El sistema muestra una interfaz que pide los datos del grupo y muestra el código del grupo si se definió que se iban a generar automáticamente (Figura 10).	
3. Inserta la información requerida	4. El sistema valida que se haya introducido toda la	

referente al grupo que desea crear.

información requerida.

5. El sistema registra los datos del grupo (nombre, código, descripción, si está activo, si es de insumo) y muestra una lista en la que agrega el nuevo grupo.



Figura 9



Figura 10

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	4 Faltan datos por entrar. 4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) por llenar y regresa al paso 2 del flujo normal de los eventos.
Sección "Modificar Grupo"	
Acción del Actor	Respuesta del Sistema
1. En la misma interfaz de Editar Grupo selecciona el Grupo que desea modificar (Figura 11).	2. El sistema muestra la información referente al grupo seleccionado.
3. Introduce los datos que desea modificar.	4. El sistema valida que no hayan productos con este grupo en caso de que el cambio sea inactivarlo. 5. El sistema valida que se haya introducido toda la

- información requerida.
6. El sistema registra la información modificada del grupo y actualiza la lista.



Figura 11

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>4 Hay productos que pertenecen a este grupo.</p> <p>4.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo y regresa al paso 2 del flujo.</p> <p>5 Faltan datos por entrar.</p> <p>5.a) El sistema muestra un mensaje informando el dato que falta por llenar y regresa al paso 2 del flujo.</p>

Sección "Eliminar Grupo"

Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Grupo selecciona el Grupo que desea eliminar (Figura 11).	2. El sistema muestra la información referente al grupo seleccionado.
3. Verifica que sea ese el grupo que desea eliminar.	<p>4. El sistema valida que el grupo no esté activo</p> <p>4.1. El sistema valida que no haya productos pertenecientes a este</p> <p>4.2. El sistema valida que no haya familias dentro de este grupo.</p>

Flujos Alternos

Acción del Actor	Respuesta del Sistema
<p>3 No es el grupo que desea eliminar</p> <p>3.a) Pasa al paso 1 del flujo.</p>	<p>4 El grupo está activo</p> <p>4.a) El sistema muestra un mensaje informando el problema y dándole al usuario la posibilidad de que la elimine de todas formas si lo desea o cancele la operación.</p> <p>4.1 Hay productos que pertenecen a este grupo</p> <p>4.1.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo</p> <p>4.2 Hay familias que pertenecen a este grupo</p>

	4.2.a) El sistema muestra un mensaje informando el problema y dándole la posibilidad al usuario de que escoja si va a eliminar el grupo con todas las familias de todas formas o si desea cancelar operación.
Sección "Registrar Familia"	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Editar Familia (Figura 12).	2. El sistema muestra una interfaz que pide los datos de la familia y muestra el código de la misma si se definió que se iban a generar automáticamente (Figura 13).
3. Inserta la información requerida referente a la familia que desea crear.	4. El sistema valida que se hayan insertado todos los datos requeridos. 5. El sistema registra los datos de la familia (nombre, código, código del grupo al que pertenece, descripción, si está activo y si es un útil y en este caso, el % de desgaste) y muestra una lista en la que agrega la nueva familia.



Figura 12

The screenshot shows the 'Editar Familia' form. It has the following fields and controls:

- Familia** section:
- Nombre:** Text field containing 'Carne de cerdo'.
- Código:** Text field containing '05'.
- Código de grupo:** Dropdown menu showing '25'.
- Activo:** Checkmark box that is checked.
- Desgaste:** Dropdown menu showing a percentage sign (%).
- Util:** Checkmark box that is unchecked.
- Descripción:** Text area with a vertical scrollbar.
- Right Panel:** A list box containing the text '2504/Carnes/Carne de pollo'.
- Buttons:** 'Crear' and 'Cancel' buttons at the bottom.

Figura 13

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	4 Faltan datos por entrar. 4.a) El sistema muestra un mensaje informando el dato que falta por llenar y regresa al paso 2 del flujo.
Sección "Modificar Familia"	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Familia selecciona la familia que desea eliminar (Figura 14).	2. El sistema muestra la información referente a la familia seleccionada.
3. Introduce los datos que desea modificar.	4. El sistema valida que no hayan productos pertenecientes esta familia en caso de que el cambio sea inactivarlo. 5. El sistema valida que se haya introducido toda la información requerida. 6. El sistema registra la información modificada de la familia y actualiza la lista.

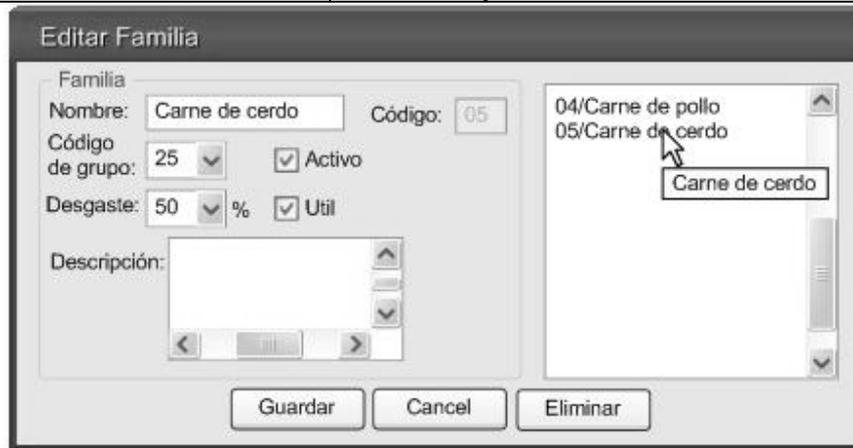


Figura 14

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4 Hay productos que pertenecen a esta familia. 4.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo. 5 Faltan datos por entrar 5.a) El sistema muestra un mensaje informando el dato que falta por llenar y regresa al paso 2 del flujo normal de los eventos.
Sección "Eliminar Familia"	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Familia selecciona la familia que desea eliminar (Figura 14).	2. El sistema muestra la información referente a la familia seleccionada.
3. Verifica que sea esa la familia que desea eliminar.	4. El sistema valida que la familia no esté activa 4.1. El sistema valida que no haya productos

	pertenecientes a esta.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3.1 No es la familia que desea eliminar 3.1.1 Pasa al paso 1 del flujo.	4 La familia está activa. 4.a) El sistema muestra un mensaje informando el problema y dándole al usuario la posibilidad de que la elimine de todas formas si lo desea o cancele la operación. 4.1 Hay productos que pertenecen a esa familia. 4.1.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo.

Sección "Listar Grupo/Familia"	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Listar G/F (Figura 15).	2. El sistema muestra una interfaz que pide el parámetro y el valor para mostrar los G/F (Figura 16).
3. El usuario introduce los datos.	4. El sistema muestra los G/F agrupados según el parámetro definido por el usuario.

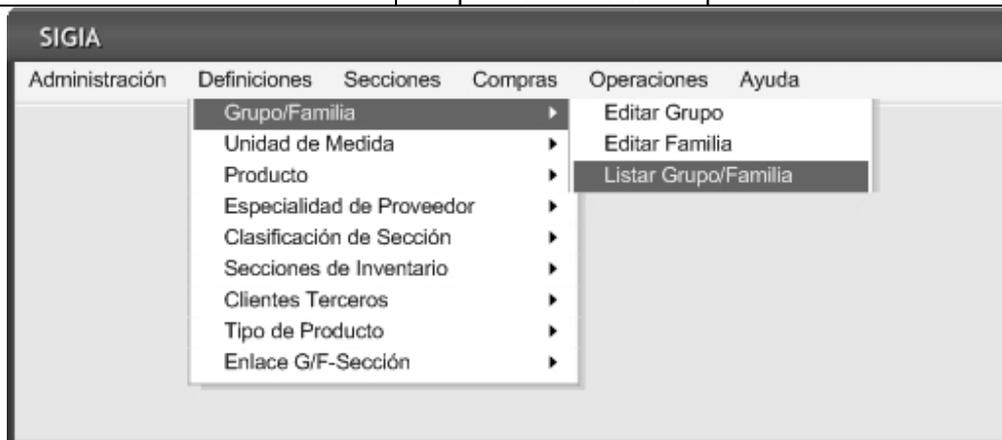


Figura 15



Figura 16

Postcondiciones	➤ El sistema guarda la información de los grupos y las familias creadas
------------------------	---

	o modificadas. ➤ El sistema muestra la información de los G/F existentes.
--	--

Caso de Uso:	Nomenclar Clasificación de Secciones
Actores:	Personal de economía
Propósito	Registrar, modificar o eliminar una clasificación de sección, así como listar las clasificaciones ya registradas.
Precondiciones:	<ul style="list-style-type: none"> • Estén registrados los Grupo/Familia de productos. • Que haya alguna clasificación ya registrada en caso de que se desee modificar o eliminar una clasificación.
Referencias	RF-7
Prioridad	Crítico

Flujo Normal de Eventos	
Sección “Registrar Clasificación de Sección”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Editar Clasificación de Sección (Figura 17)	2. El sistema muestra una interfaz que pide la información para crear una nueva clasificación y muestra el código ya definido si se definió que se iba a asignar automáticamente (Figura 18).
3. Inserta la información.	4. El sistema valida que se haya insertado toda la información requerida. 5. El sistema valida que no haya otras clasificaciones con la misma selección de familias de entrada y de salida. 6. El sistema registra la información referente a dicha clasificación (Nombre [central de insumos, central de mantenimiento, central de comestibles, elaboración, carnicería, punto de venta, punto de servicio, gasto], Código, si está activa o no, descripción, familias de entrada y familias de salida) y muestra una lista en la que agrega la nueva clasificación.



Figura 17



Figura 18

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	4 No se han insertado todos los datos 4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) por introducir y sugiriéndole que lo inserte. 5 Hay otra clasificación igual 5.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo.
Sección "Modificar clasificación de secciones"	
Acción del Actor	Respuesta del Sistema

1. En la interfaz de Editar Clasificación de Secciones selecciona la clasificación que desea modificar (Figura 19).	2. El sistema muestra la información correspondiente a la clasificación seleccionada.
3. Inserta la información que desea modificar, excepto el nombre que no se puede cambiar y el código en caso de que se haya definido que se va a asignar automáticamente.	4. El sistema valida que se haya insertado toda la información requerida. 5. El sistema valida que no haya otras clasificaciones con la misma selección de familias de entrada y de salida. 6. El sistema valida que no hayan operaciones pendientes que incluyan secciones con esa clasificación. 7. El sistema actualiza la información.

Figura 19

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	4 Faltan datos por llenar. 4.a) El sistema muestra un mensaje informándole el/los dato(s) que falta(n) y sugiriéndole que lo(s) introduzca(n). 5 Selección de familias igual a la de otra clasificación. 5.a) El sistema muestra un mensaje informándole que ya esa clasificación existe. 6 Operaciones pendientes. 6.a) El sistema muestra un mensaje informándole que no se puede realizar la operación por este motivo.
Sección "Eliminar clasificación de secciones"	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Clasificación selecciona la clasificación que desee eliminar. (Figura 19)	2. El sistema muestra la información correspondiente a la clasificación seleccionada.

<p>3. Verifica que es la clasificación que quiere eliminar.</p>	<p>4. El sistema valida que la clasificación no esté activa. 4.1. El sistema valida que no haya existencia de productos en secciones con esta clasificación. 4.2. El sistema valida que no hayan operaciones pendientes que incluyan secciones con esa clasificación. 5. El sistema elimina la clasificación seleccionada y elimina la clasificación de la lista.</p>
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
<p>3 No es la clasificación que desea eliminar. 3.a) Pasa al paso 1 del flujo.</p>	<p>4 La clasificación está activa. 4.a) El sistema muestra un mensaje informando esta situación y le da la posibilidad de eliminar de todas formas. 4.1 Hay existencias 4.1.a) El sistema muestra un mensaje informando que no se puede realizar la operación por ese motivo y vuelve al paso 1 del flujo. 4.2. Hay operaciones pendientes. 4.2.a) El sistema muestra un mensaje informando que no se puede efectuar la operación por este motivo y vuelve al paso 1 del flujo.</p>
Sección “Listar Clasificación de sección”	
Acción del Actor	Respuesta del Sistema
<p>1. Selecciona la opción Listar Clasificaciones de Sección (Figura 20).</p>	<p>2. El sistema muestra una interfaz que pide el parámetro y el valor para mostrar las clasificaciones (Figura 21).</p>
<p>3. Introduce los datos.</p>	<p>4. El sistema muestra las clasificaciones agrupados según el parámetro definido por el usuario.</p>



Figura 20

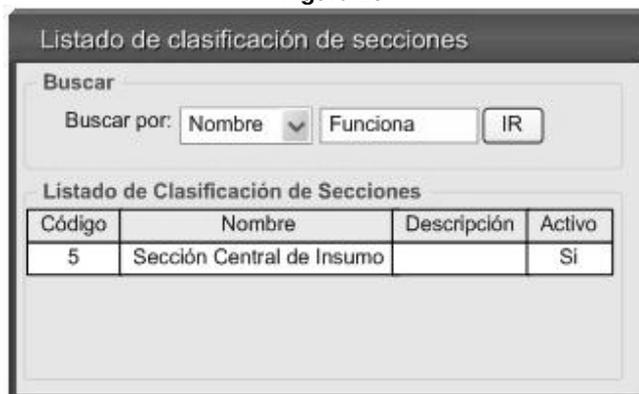


Figura 21

Postcondiciones	<ul style="list-style-type: none"> ➤ El sistema guarda la información de las clasificaciones de sección creadas o modificadas. ➤ El sistema muestra la información de las clasificaciones existentes.
------------------------	---

Caso de Uso:	Nomenclar Secciones de Inventario
Actores:	Personal de economía
Propósito	Registrar, modificar o eliminar una sección de inventario, así como listar las secciones ya existentes.
Precondiciones:	<ul style="list-style-type: none"> • Estén registrada las Clasificaciones de sección. • Que haya alguna sección ya registrada en caso de que se desee modificar o eliminar una sección.
Referencias	RF-9
Prioridad	Crítico
Flujo Normal de Eventos	
Sección "Registrar Sección de inventario"	
Acción del Actor	Respuesta del Sistema

1. Selecciona la opción Editar Sección (Figura 22).	2. El sistema muestra una interfaz que pide la información para crear una nueva sección y muestra el código ya definido si se definió que se iba a asignar automáticamente
3. Introduce la información.	<p>4. El sistema valida que se haya insertado toda la información necesaria.</p> <p>3. El sistema registra la información referente a dicha clasificación (Nombre, código, si está activo, si se va a validar existencia, descripción, responsable de sección, clasificación de sección, forma de valorar las entradas (LIFO, Ponderado y FIFO), forma de generar los consecutivos de documentos (Manual o automático)) y muestra una lista en la que agrega la nueva sección.</p>



Figura 22



Figura 23

Flujos Alternos

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	<p>4. No se ha insertado toda la información.</p> <p>4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) por introducir y sugiriéndole que lo inserte.</p>
--	--

Sección “Modificar Sección de inventario”

Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Sección de Inventario selecciona la sección que desea modificar (Figura 24).	2. El sistema despliega la información correspondiente a la sección seleccionada.
3. El usuario introduce la información que desea modificar, excepto la clasificación que no se puede cambiar y el código en caso de que se haya definido que se va a asignar automáticamente.	<p>4. El sistema valida que se haya introducido toda la información requerida.</p> <p>5. El sistema valida que la sección no esté activada.</p> <p>5.1. El sistema valida que no haya existencias de productos en la sección.</p> <p>5.2. El sistema valida que no hayan operaciones pendientes que incluyan esta sección.</p> <p>6. El sistema guarda la nueva información.</p>



Figura 24

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>4 No se ha insertado toda la información</p> <p>4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) por introducir y sugiriéndole que lo inserte.</p> <p>5 La sección está activa.</p> <p>5.a) El sistema muestra un mensaje informando esta situación y le da la posibilidad de modificarla de todas formas.</p> <p>5.1 Hay existencias</p> <p>5.1.a) El sistema muestra un mensaje informando que no se puede realizar la operación por ese motivo y vuelve al paso 1 del flujo.</p>

	5.2. Hay operaciones pendientes. 5.2.a) El sistema muestra un mensaje informando que no se puede efectuar la operación por este motivo y vuelve al paso 1 del flujo.
Sección “Eliminar Sección de inventario”	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Sección de Inventario selecciona la sección que desea modificar.	2. El sistema despliega la información correspondiente a la sección seleccionada.
3. Verifica que es la sección que desea eliminar.	4. El sistema valida que la sección no esté activada. 4.1. El sistema valida que no haya existencias de productos en la sección. 4.2. El sistema valida que no haya operaciones pendientes que incluyan esta sección. 5. El sistema elimina la sección seleccionada y la quita de la lista.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3 No es la sección que desea eliminar 3.a) Pasa al paso 1 del flujo.	4 La sección está activa. 4.a) El sistema muestra un mensaje informando esta situación y le da la posibilidad de eliminar de todas formas. 4.1 Hay existencias 4.1.a) El sistema muestra un mensaje informando que no se puede realizar la operación por ese motivo y vuelve al paso 1 del flujo. 4.2. Hay operaciones pendientes. 4.2.a) El sistema muestra un mensaje informando que no se puede efectuar la operación por este motivo y vuelve al paso 1 del flujo.
Sección “Listar Secciones de inventario”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Listar Secciones de Inventario (Figura 25).	2. El sistema muestra una interfaz que pide el parámetro y el valor para mostrar las secciones (Figura 26).
3. Introduce los datos.	4. El sistema muestra las secciones agrupadas según el parámetro definido por el usuario.



Figura 25



Figura 26

Postcondiciones	<ul style="list-style-type: none"> ➤ El sistema guarda la información de las secciones creadas o modificadas. ➤ El sistema muestra la información de las secciones existentes.
------------------------	--

Caso de Uso:	Nomenclar Unidad de Medida
Actores:	Personal de Economía
Propósito	Registrar, modificar y eliminar una Unidad de Medida; y listar las unidades ya existentes.
Precondiciones:	<ul style="list-style-type: none"> ➤ El actor debe haberse autenticado correctamente.
Referencias	RF-13
Prioridad	Crítico
Flujo Normal de Eventos	
Sección "Editar Unidad de Medida"	
Acción del Actor	Respuesta del Sistema
4. Selecciona la opción Editar Unidad de Medida (Figura 27).	5. El sistema muestra una interfaz que pide la información para crear una nueva unidad de medida y muestra el

	código si se definió que se iba a asignar automáticamente (Figura 28).
5. Introduce la información.	<p>6. El sistema valida que se haya insertado toda la información necesaria.</p> <p>6. El sistema registra la información referente a dicha unidad de medida (Nombre, código, si está activo, descripción y la métrica (real o entero)) y muestra una lista en la que agrega la nueva unidad de medida.</p>



Figura 27



Figura 28

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	<p>4. No se ha insertado toda la información.</p> <p>4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) por introducir y sugiriéndole que lo inserte.</p>
Sección “Modificar Unidad de Medida”	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Unidad de Medida selecciona la unidad que	2. El sistema despliega la información correspondiente a la unidad de medida seleccionada.

desea modificar (Figura 29).	
3. Introduce la información que desea modificar, excepto el código en caso de que se haya definido que se va a asignar automáticamente.	4. El sistema valida que se haya introducido toda la información requerida. 5. El sistema valida que no haya registro en los históricos de esta unidad de medida. 6. El sistema guarda la nueva información.



Figura 29

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4. No se ha insertado toda la información. 4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) y sugiriéndole que lo(s) inserte. 5 Hay registro en los históricos. 5.a) El sistema muestra un mensaje informando que no se puede efectuar la operación por este motivo.
Sección "Eliminar Unidad de Medida"	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Unidad de Medida selecciona la unidad que desea modificar.	2. El sistema despliega la información correspondiente a la unidad de medida seleccionada.
3. Verifica que es la unidad de medida que desea eliminar.	4. El sistema valida que no haya registro en los históricos de esta unidad de medida. 5. El sistema elimina la unidad de medida seleccionada y la quita de la lista.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4 Hay registro en los históricos. 4.a) El sistema muestra un mensaje informando que no se puede efectuar la operación por este motivo y vuelve al paso 2 del flujo.
Sección "Listar Unidad de Medida"	
Acción del Actor	Respuesta del Sistema

1. Selecciona la opción Listar Unidades de Medida (Figura 30).	2. El sistema muestra una interfaz que pide el parámetro y el valor para mostrar las unidades de medida (Figura 31/Figura 26).
3. Introduce los datos.	4. El sistema muestra las unidades de medida agrupadas según el parámetro definido por el usuario.

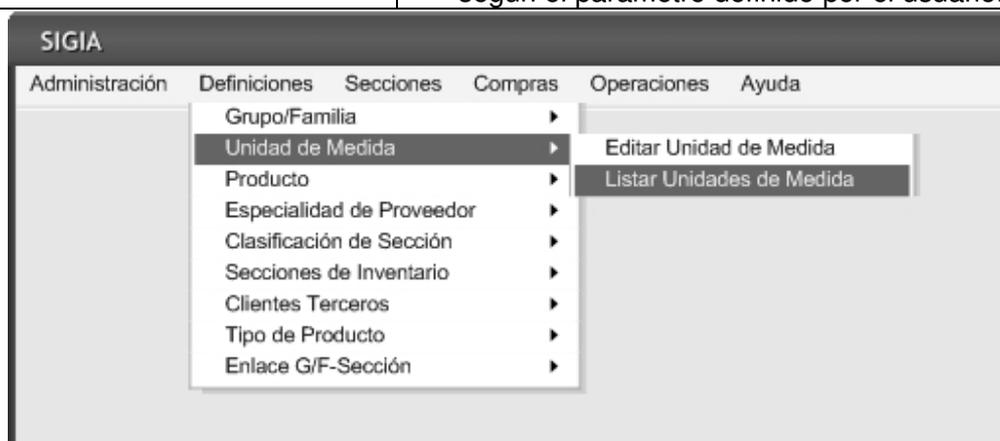


Figura 30

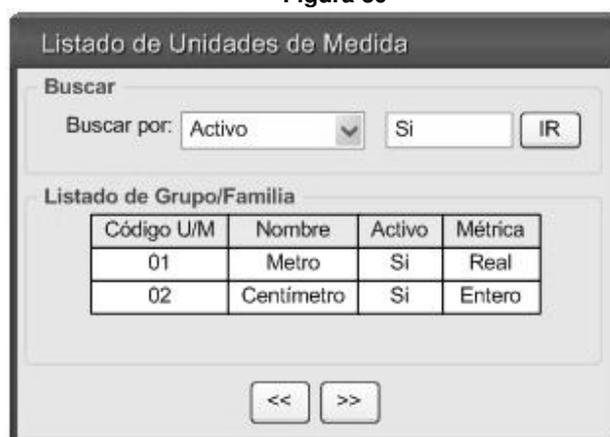


Figura 31

Postcondiciones	<ul style="list-style-type: none"> ➤ El sistema guarda la información de las unidades de medida creadas o modificadas. ➤ El sistema muestra la información de las unidades de medida existentes.
------------------------	--

Caso de Uso:	Nomenclar Producto
Actores:	Personal de Economía
Propósito	Permitir registrar, modificar, eliminar y mostrar la información de un producto.
Precondiciones:	<ul style="list-style-type: none"> • El actor debe haberse autenticado correctamente. • Deben estar creadas los Grupo/Familia. • Deben estar creadas las Unidades de Medida.

Referencias	RF-19
Prioridad	Crítico
Flujo Normal de Eventos	
Sección "Editar Producto"	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Editar Producto (Figura 32).	2. El sistema muestra una interfaz que pide la información para crear un nuevo producto y muestra el código si se definió que se iba a asignar automáticamente (Figura 33).
3. Introduce la información.	4. El sistema valida que se haya insertado toda la información necesaria. 5. El sistema registra la información referente a dicho producto (Nombre, código, grupo/familia, si está activo, descripción, unidad de medida base y el tipo de producto (elaborado o corriente)) y muestra una lista en la que agrega el nuevo producto.



Figura 32

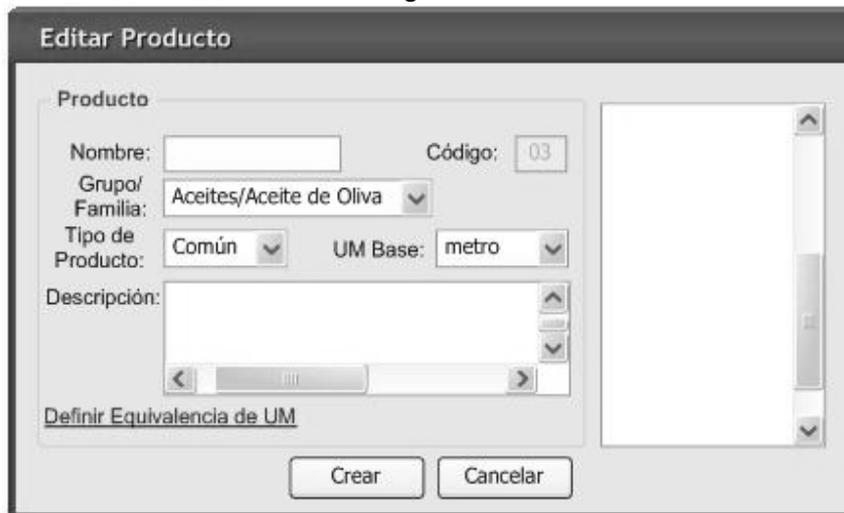


Figura 33

Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4. No se ha insertado toda la información. 4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) y sugiriéndole que lo(s) inserte y regresa al paso 2 del flujo.
Sección "Modificar Producto"	
Acción del Actor	Respuesta del Sistema
1. En la interfaz de Editar Producto selecciona el producto que desea modificar (Figura 34).	2. El sistema despliega la información correspondiente al producto seleccionado.
3. Introduce la información que desea modificar, excepto el código en caso de que se haya definido que se va a asignar automáticamente.	4. El sistema valida que se haya introducido toda la información requerida. 5. El sistema valida que no haya existencias de este producto en ninguna sección.

6. El sistema valida que no haya operaciones pendientes que lo incluyan.
7. El sistema guarda la nueva información.

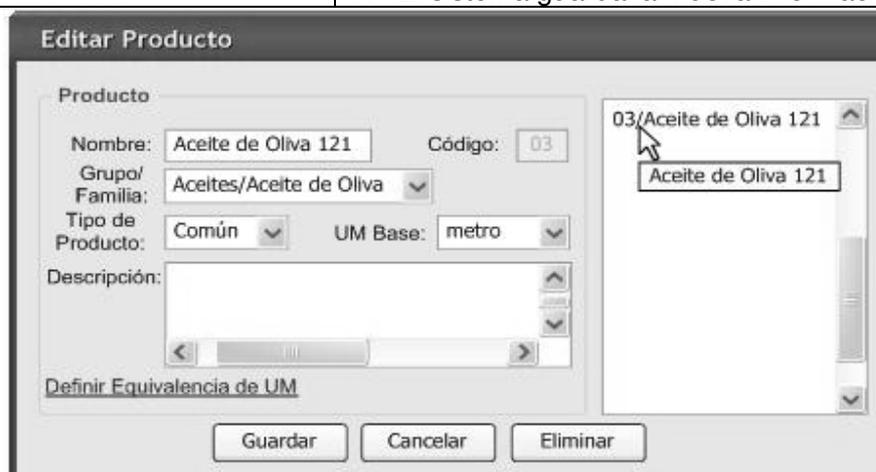


Figura 34

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	<p>4. No se ha insertado toda la información. 4.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) y sugiriéndole que lo(s) inserte. 5 Hay existencias 5.a) El sistema muestra un mensaje de error informando que no se puede realizar la operación por ese motivo y vuelve al paso 2 del flujo. 6 Hay operaciones pendientes 6.a) El sistema muestra un mensaje informando que no se puede realizar dicha operación por ese motivo y vuelve al paso 2 del flujo.</p>

Sección "Eliminar Producto"

Acción del Actor	Respuesta del Sistema
<p>1. En la interfaz de Editar Producto selecciona el producto que desea modificar (Figura 34).</p>	<p>2. El sistema despliega la información correspondiente al producto seleccionado.</p>
<p>3. Verifica que es el producto que desea eliminar.</p>	<p>4. El sistema valida que no haya existencias de este producto en ninguna sección. 5. El sistema valida que no haya operaciones pendientes que lo incluyan. 6. El sistema elimina el producto seleccionado y lo quita de la lista.</p>

Flujos Alternos

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	<p>4 Hay existencias</p> <p>4.a) El sistema muestra un mensaje de error informando que no se puede realizar la operación por ese motivo y vuelve al paso 2 del flujo.</p> <p>5 Hay operaciones pendientes</p> <p>5.a) El sistema muestra un mensaje informando que no se puede realizar dicha operación por ese motivo y vuelve al paso 2 del flujo.</p>
--	--

Sección “Listar Productos”

Acción del Actor	Respuesta del Sistema
5. Selecciona la opción Listar Productos (Figura 35).	6. El sistema muestra una interfaz que pide el parámetro y el valor para mostrar los productos (Figura 26).
7. Introduce los datos.	8. El sistema muestra productos agrupados según el parámetro definido por el usuario.



Figura 35

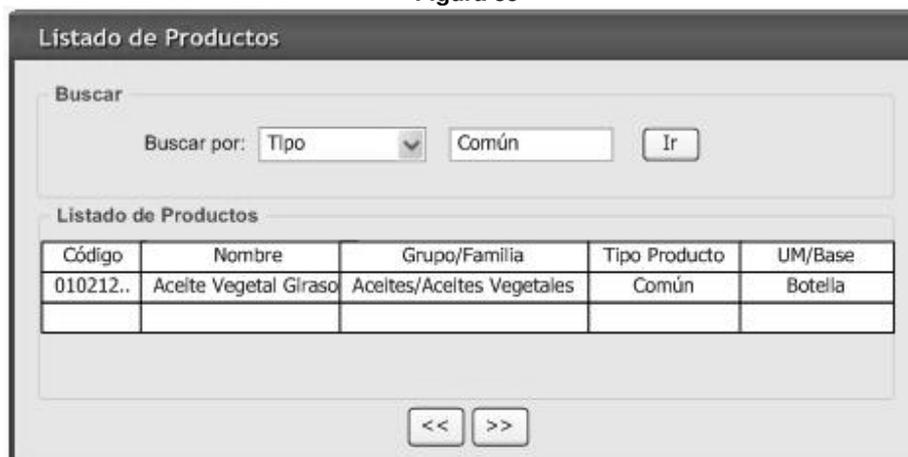


Figura 36

Puntos de Extensión

Selecciona la opción Definir Equivalencia de UM.	Se ejecuta el caso de uso extendido “Registrar Equivalencias de Unidad de Medida en productos”.
--	---

Postcondiciones	Se crea, se modifica, se elimina o se muestra la información de un producto.
Caso de Uso:	Registrar Equivalencias de Unidad de Medida en productos
Actores:	Personal de Economía
Propósito	Registrar o modificar una equivalencia de UM.
Precondiciones:	<ul style="list-style-type: none"> • El actor debe haberse autenticado correctamente. • Deben estar registradas previamente las UM.
Referencias	RF-15
Prioridad	Crítico

Flujo Normal de Eventos

Sección “Editar Equivalencias de U/M en productos”

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una interfaz que pide la información para definir la equivalencia de UM del producto (Figura 37).
2. Introduce la información requerida.	3. El sistema valida que se haya insertado toda la información necesaria. 4. El sistema registra la información referente a dicha equivalencia (UM menor, UM mayor, cantidad).



Figura 37

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	3. No se ha insertado toda la información. 3.a) El sistema muestra un mensaje informando el/los dato(s) que falta(n) y sugiriéndole que lo(s) inserte.

Sección “Modificar Equivalencias de U/M en productos”

Acción del Actor	Respuesta del Sistema
1. En la interfaz Editar Equivalencia de UM, inserta la información que desea modificar (Figura 37).	2. El sistema valida que se haya insertado toda la información necesaria. 3. El sistema valida que el producto no esté en uso. 4. El sistema registra la nueva información referente a dicha equivalencia.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
	2. No se ha insertado toda la información. 2.a) El sistema muestra un mensaje informando el/los

	dato(s) que falta(n) y sugiriéndole que lo(s) inserte. 3. El producto está en uso. 3.a) El sistema muestra un mensaje informando que no se puede realizar la operación por este motivo.
Postcondiciones	Se guarda la equivalencia de unidad de medida de un producto.

2.6 Evaluación de los resultados. Validación de Requisitos de Software

Evaluar los resultados obtenidos ofrece la ventaja de conocer los errores para suprimirlos en procesos de desarrollo de software futuros. Las actividades de la IR presentes en el campo de acción del objeto de estudio tienen su contrapartida evaluativa en la validación de los requisitos de software. Desde el MN como punto de partida de las actividades de IR hasta la modelación del sistema, y los requisitos de software (RF y RNF) producto de estos FT, merecen una evaluación para identificar si son completos, consistentes y correctos.

El equipo de calidad interna del proyecto, basado en las listas de chequeo propuestas por RUP y las métricas de aseguramiento de la calidad, aplicó los parámetros en función de controlar el desempeño de las actividades de la IR desarrolladas en el SIGIA. Fueron aplicadas a los artefactos correspondientes dentro de cada FT, en vistas a obtener un estado cuantificable y un nivel concreto del funcionamiento de las prácticas empleadas.

2.6.1 Resultados de las Listas de Chequeo para Modelo del Negocio.

El MN como la base para la ERE permitió la comprensión de los procesos del negocio de la GIA en el país. Cada caso de uso del negocio responde a un flujo de actividades que produce un resultado de valor para el actor. De conjunto con las técnicas de elicitación empleadas (entrevistas y revisión de documentos) el MN arrojó los resultados concretos respecto al tratamiento de la GIA y su necesidad de automatizarla, de acuerdo a los objetivos económicos y de mercado a los que se encuentran sometidas las empresas cubanas actualmente.

Clientes, usuarios desarrolladores y demás interesados lograron un entendimiento común respecto a los lineamientos y especificidades de la GIA en el país. Se identificaron los procesos del negocio, mediante las entrevistas y revisión de documentos, y se validó con el cliente que los casos de uso abarcaban correctamente todos los procesos de GIA de las empresas.

Los artefactos que complementan este FT (Modelación del Negocio) se desarrollaron y aprobaron por el equipo de calidad interna del proyecto. La integridad y completitud de las descripciones, los diagramas de actividad, de objetos del negocio, las reglas del negocio y de forma general el MN constituyeron la base para la ERE. FT que permitió el paso a las próximas actividades de la IR para el módulo Nomencladores del SIGIA.

2.6.2 Resultados de las Listas de Chequeo para Modelado del Sistema.

El modelado del sistema para módulo de Nomencladores de SIGIA constituyó una forma para validar la consistencia de los requisitos.

- Los requisitos relacionados con este módulo responden al menos a un caso de uso del sistema y todos los casos de uso están soportado por al menos un requisito.
- No existe alto nivel de complejidad en las relaciones entre casos de uso. De diez casos de uso identificados se modeló una sola relación de extensión comprometiendo únicamente dos casos de uso.
- Las descripciones de los casos de uso no definen como el sistema pone en funcionamiento sus tareas internamente, sino limitándose a mostrar la interacción entre el actor y el sistema. Para visualizar mejor estas descripciones se utilizaron prototipos de interfaz por cada una de estas.
- De la actividad de especificación se obtuvo un seguimiento de requisito de 1:1 (los diez requisitos responden a diez casos de usos en el sistema).

Finalmente el modelado del sistema para el módulo de Nomencladores se completó dando paso a las disciplinas restantes del Proceso de Desarrollo de Software del SIGIA.

2.6.3 Métricas de aseguramiento de la calidad para levantamiento de requisitos.

La etapa de levantamiento de los requisitos es de suma importancia pues en la misma se determina las funcionalidades del sistema así como las necesidades del cliente y todo el ambiente factible en que debe desarrollarse el sistema, para que funcione eficientemente. A esta etapa se le aplicaron métricas que evalúan cuatro propiedades: completitud, consistencia, correctitud y complejidad.

El factor de completitud representa el acabado final de todos los elementos, que se encuentren completos. El factor de consistencia define el grado en que los elementos del artefacto representan en forma única un aspecto del problema sin contradicciones, la correctitud determina el grado de adecuación del artefacto para satisfacer los requisitos establecidos. El factor de complejidad permite medir el grado de claridad de los artefactos. Los resultados pueden observarse en la tabla siguiente:

	Factor	Métricas Asociadas	Valor
	Factores de Completitud		

Factor 1. ¿Han sido involucradas todas las áreas funcionales relevantes a las cuales apoyará el sistema?	Métrica 1: Número de áreas funcionales relevantes omitidas.	3%
Factor 2. ¿Han sido involucradas todas las áreas funcionales secundarias a las cuales apoyará el sistema?	Métrica 2: Número de áreas funcionales secundarias omitidas.	5%
Factor 3. ¿Han sido definidos todos los roles relevantes de usuario encargados de generar/ modificar o consultar información?	Métrica 3: Número de roles relevantes omitidos.	0%
Factor 4. ¿Han sido definidos todos los roles secundarios de usuario encargados de generar/modificar o consultar información?	Métrica 4: Número de roles secundarios omitidos.	2%
Factor 5. ¿Han sido considerados todos los sistemas externos con los cuáles interactuará el sistema?	Métrica 5: Número de sistemas externos omitidos.	0%
Factor 6. ¿Se presenta una descripción resumida (descripción de alto nivel) de todos los casos de uso del negocio?	Métrica 6: Número de casos de uso que no tiene descripción resumida	0%
Factor 7. ¿Están definidos todos los requisitos que justifican la funcionalidad del caso de uso?	Métrica 7: Número de requisitos omitidos por caso de uso Métrica 8: Número de casos de uso que tienen requisitos omitidos	0%
Factor 8. ¿Existen requisitos que no han sido considerados en algún caso de uso?	Métrica 9: Número de requisitos que no son considerados en ningún caso de uso.	0%

	Factor 9. ¿Han sido definidos todos los roles de usuario encargados de actividades de soporte/ mantenimiento / auditoría?	Métrica 10: Número de roles de soporte / mantenimiento / auditoría omitidos.	10%
	Factor 10. ¿Se presenta una descripción detallada (descripción extendida esencial) de todos los casos de uso del negocio?	Métrica 11: Número de casos de uso que no poseen una descripción extendida.	0%
	Factor 11. ¿Están todas las acciones del flujo de eventos redactadas en función del responsable?	Métrica 12: Número de acciones del flujo de eventos que no están redactadas en función del responsable. Métrica 13: Número de casos de uso que no describen condiciones de excepción relevantes	0%
	Factor 12. ¿Se describen las condiciones de excepción relevantes que debe contemplar cada flujo de eventos?	Métrica 14: Número de casos de uso que tienen un nombre incorrecto	0%
	Factor 13. ¿Todos los casos de uso del negocio han sido clasificados de acuerdo a su relevancia (primario / secundario / opcional)?	Métrica 15: Número de casos de uso que no representan una interacción observable por un actor	0%
			96%
Factores de Consistencia			
	Factor 14. ¿El nombre dado a los casos de uso es una expresión verbal que	Métrica16: Número de casos de uso que se	0%

	describe alguna funcionalidad relevante en el contexto del usuario?	solapan	
	Factor 15. ¿Representa el caso de uso una interacción observable por un actor?	Métrica17: Número de acciones del flujo de eventos que no se corresponde la definición de las con el responsable.	0%
	Factor 16. ¿No existe solapamiento en la funcionalidad que representan los diferentes casos de uso?	Métrica 18: Número de casos de uso que tienen acciones del flujo de eventos asignados a un responsable que no le corresponde.	0%
	Factor 17. ¿Existen acciones en el flujo de eventos asignadas a un responsable que no le corresponde?	Métrica 19: Grado de adecuación de la descripción del flujo de eventos para un caso de uso Métrica 20: Número de casos de uso que tienen acciones del flujo de eventos asignados a un responsable que no le corresponde.	0%
	Factor 18. ¿Está adecuadamente redactado (en el lenguaje del usuario) el flujo de eventos?	Métrica 21: Grado de adecuación de la descripción del flujo de eventos para un caso de uso Métrica 22: Número de	0%

		casos de uso no aceptados	
	Factor 19. ¿La descripción del flujo de eventos se inicia con la descripción de una acción externa originada por un actor o por una condición interna del sistema claramente identificable?	Métrica 23: Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema	0%
	Factor 20. Si en el caso de uso interviene mas de un actor, ¿existe claridad en cuál de ellos es el actor iniciador?	Métrica 24: Número de casos de uso con más de un actor que no describe cuál es el actor iniciador	0%
	Factor 21. ¿Existe una adecuada separación entre el flujo básico de eventos y los flujos alternos y/o flujos subordinados?	Métrica 25: Número de casos de uso complejos que no tienen separación del flujo básico y de flujos alternos	0%
			100%
Factores de Correctitud			
	Factor 22. ¿Existe para cada caso de uso de negocio por lo menos un usuario responsable?	Métrica 26: Número de casos de uso que no tienen un usuario responsable	0%
	Factor 23. ¿Representa el caso de uso requisitos comprensibles por el usuario?	Métrica 27: Grado en que los requisitos representados por el caso de uso son comprensibles por el usuario Métrica 28: Número de casos de uso en que los requisitos representados no	0%

		son comprensibles por el usuario	
	Factor 24. ¿Se ajusta la representación del diagrama del caso de uso de acuerdo a lo normado en la metodología?	Métrica 29: Grado en que se ajusta el diagrama del caso de uso a la metodología.	0%
	Factor 25. ¿Las interacciones definidas describen la funcionalidad requerida del sistema?	Métrica 30: Grado en que las interacciones definidas describen la funcionalidad solicitada por el usuario Métrica 31: Número de casos de uso que deben ser modificados para adecuarlos a la funcionalidad del sistema	0%
	Factor 26. ¿Las interacciones definidas introducen mejoras al proceso actual?	Métrica 32: Número de casos de uso que deben ser modificados para mejorar el proceso actual	5%
	Factor 27. ¿Se ajusta la representación del diagrama del caso de uso de acuerdo a lo normado en la metodología?	Métrica 33: Grado en que se ajusta el diagrama del caso de uso a la metodología.	0%
			99.16%

El análisis de estos factores con sus respectivas métricas y el porcentaje que representa el aporte de cada atributo a la calidad permitieron validar la efectividad de las actividades desarrolladas.

El factor completitud arrojó un 96%, por tanto la mayoría de los atributos se encuentran completos; por ejemplo en cada caso de uso del negocio se presenta una descripción de alto nivel (resumida), están definidos todos los requisitos que justifican la funcionalidad del caso de uso, no existen requisitos que no

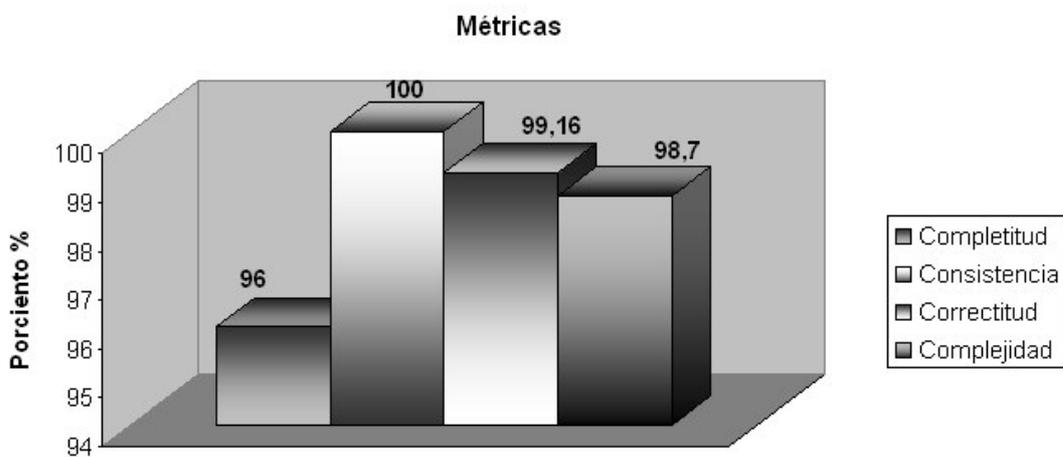
hayan sido considerados en algún caso de uso, entre otros factores importantes para el desarrollo de esta etapa.

En el factor de consistencia se obtuvo un 100%, este factor fue el que más influyó en la calidad final. Todos los artefactos fueron redactados en un lenguaje natural y entendible tanto por el cliente como por el equipo de desarrollo, existe una adecuada separación entre el flujo básico de eventos y los flujos alternos y/o flujos subordinados de los casos de uso presentados, etc. Ninguno de los atributos analizados dentro de este factor de calidad arrojó un resultado desfavorable que influyese en la calidad final de este factor.

El factor correctitud, que influye en el grado de adecuación de los artefactos para satisfacer los requisitos establecidos, mostró un 99.16% de aporte a la calidad de este factor.

Finalmente el factor complejidad que permite conocer la alta claridad en los artefactos desarrollados aportó un 98.7% teniendo como resultado que de forma general los atributos analizados para este factor no contienen una alta complejidad.

Los factores de calidad analizados influyeron positivamente en esta etapa permitiendo que los requisitos capturados para el SIGIA se ajustaran a las necesidades del cliente (expresadas en la Carta de Conformidad con los requisitos emitida por el cliente, Carta de Conformidad del cliente con los requisitos.) y diera paso al resto de las actividades de desarrollo del software.



2.6.4 Validación de Requisitos de Software

La validación de los requisitos examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencias, sin omisiones, que los errores

detectados hayan sido corregidos, y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto [Pressman05].

La finalidad de validar los requisitos de software es explorarlos para asegurar que definen el sistema adecuado (el que el cliente espera). Verifica que las especificaciones de requisitos se corroboran con las necesidades de clientes/usuarios finales y son correctas las interpretaciones por parte del equipo de desarrollo de software. Esta actividad permite no emplear esfuerzos de implementación en requisitos innecesarios o incompletos.

La validación de requisitos para el SIGIA se basó en el método de las revisiones y constó de dos partes fundamentales: la primera, las revisiones a nivel de equipo de analistas y la segunda las revisiones con el propio cliente (sobre el principio de que “valida el cliente” [Fernández06]). El grupo de analistas de conjunto con el equipo de calidad sobre la base de las listas de chequeo controló la especificación efectiva de los requisitos, antes de valorarlos con el cliente. Se confeccionó por cada caso de uso y por cada uno de sus escenarios un prototipo no funcional; esto es una forma más de validar y refinar los requisitos. Estos prototipos se encuentran en la descripción de los casos de uso del sistema.

Una vez especificados debidamente los requisitos se contactó con el cliente para su aprobación mediante un documento formal. Este documento avala por parte del cliente que sus necesidades fueron correctamente analizadas y transformadas en los requisitos para el módulo Nomencladores del SIGIA.

Carta de Conformidad del cliente con los requisitos.



***Vicerrectoría de
Logística.***

Ciudad de la Habana, 16 de Abril del 2006.

“Año 49 de la Revolución”

Mediante el presente documento se expresa la conformidad con los requisitos del Sistema de Gestión de Inventarios y Almacenes. Estos:

- Integran las normas vigentes de contabilidad y las regulaciones establecidas por el Ministerio de Finanzas y Precios.

- En el documento de requisitos presentado a los especialistas se recogieron las principales operaciones de los almacenes, aplicables a la Universidad.
- La información asociada a cada operación se almacena con el formato requerido y establecido por los modelos estándares para dicha operación.

De manera general los requisitos constituyen una propuesta que integra muy bien las funcionalidades de un sistema de inventarios. Avala la presente,

Héctor Iván Rodríguez

Asesor Económico de la Vicerrectoría de Logística.

2.7 Conclusiones Parciales.

La modelación del negocio de la GIA constituyó el punto de partida para que la obtención de los requisitos funcionales y no funcionales del software se ajustara a las necesidades del cliente.

El modelo de casos de uso del sistema del módulo Nomencladores se construyó como la representación para una primera visión de las funcionalidades de los procesos de negocio en el SIGIA.

Los resultados del epígrafe 2.6 muestran la efectividad de las actividades de IR desarrolladas mediante el grado de satisfacción del cliente, objetivo fundamental dentro de todo este proceso.

CONCLUSIONES GENERALES

- El desarrollo de la IR, con sus máximos representantes (analistas de sistema), alcanza en la actualidad un mayor auge a razón de la idea común de unificar esfuerzos en el desarrollo de software de calidad y en función de cumplir las expectativas del máximo beneficiado (el cliente). Desde la perspectiva de la mejora continua de la comunicación cliente – desarrollador mezclando a ambos en los diferentes entornos a los que el software se vincula y por el perfeccionamiento de los procesos en las empresas, desde la óptica de desarrollo tecnológico.
- La IR resulta tan o más importante que la propia implementación del sistema. Antes de comenzar la implementación de un producto, es necesario delimitar lo que se desea desarrollar, para no emplear esfuerzos en vano en funcionalidades que pueden ser innecesarias.
- Las técnicas desarrolladas durante la actividad de elicitación de requisitos para el módulo Nomencladores del SIGIA arrojaron resultados satisfactorios para esta etapa y reflejados en los requisitos obtenidos. La combinación de varias técnicas aumentan la calidad de los requisitos pues se obtiene la información por diferentes vías y puntos de vista.
- La unificación de las actividades de los roles de analista de sistemas y analista de procesos de negocio en un solo rol (analista) en el desarrollo de la IR para el módulo Nomencladores del SIGIA constituyó una vía óptima en el entendimiento de los procesos de GIA. Esta interrelación redujo considerablemente las fallas de comunicación entre cliente-desarrolladores. La responsabilidad recae sobre un individuo, y a medida que existen menos intermediarios más fluida y certera es la información.
- El MN de los procesos de la GIA desarrollado constituyó la base fundamental para el entendimiento de estos, y la identificación de los que serían objeto de automatización. Se exploró el negocio, concluyendo que la definición de todos los conceptos básicos de soporte a las operaciones de GIA estarían recogidos en los Nomencladores (definición de los Grupo/Familia, de los productos por cada sección, etc.)
- Sobre la base de las actividades de IR enmarcadas en el campo de acción del objeto de estudio se resolvieron las contradicciones existentes entre cliente-desarrolladores respecto al entendimiento de los procesos de GIA, en las empresas. Esto permitió una especificación de software acorde a las expectativas del cliente y reflejada en la carta de conformidad con los requisitos emitida por el beneficiado y anexada en este documento.

- La Modelación del Sistema para los Nomencladores responde a los requisitos de software capturados, siendo así el módulo que define los conceptos de productos, proveedores, G/F, etc. para las entidades del país independientemente de su tipo o sector.
- La validación del cliente permitió delimitar las actividades de implementación de acuerdo con sus expectativas y sirvió como constancia final para disminuir los posibles cambios en los requisitos.

RECOMENDACIONES

- Desarrollar la actividad de gestión de requisitos de la IR a lo largo del ciclo de vida del desarrollo de software.
- La utilización de una herramienta CASE (RequisitePro) para la gestión de los requisitos de software de los restantes módulos del SIGIA, por las facilidades que proporciona, y estar integrada dentro de la Herramienta de Rational además de basarse en la metodología de desarrollo de software RUP.
- La reutilización de los requisitos capturados en otros Softwares de Gestión, sobre la base de las experiencias obtenidas. Esto posibilitará la reducción de esfuerzos y tiempo en la actividad de ERE.
- Estudiar y volcar los esfuerzos en virtud de minimizar la curva de errores que provoca una deficiente IR en esta etapa de desarrollo de un proyecto de software.

REFERENCIAS BIBLIOGRÁFICAS

- [Arboleda05] Arboleda Jiménez, Hugo F. “Modelos de ciclo de vida en desarrollo de software”. *Asociación Colombiana de Ingenieros de Sistemas*. [En línea] 4 de Octubre de 2005. [Citado el: 10 de abril de 2007.] <http://www.acis.org.co/index.php?id=552>.
- [Bauer69] Bauer. Dr. F. L “Software Engineering”. *Report on a conference sponsored by the NATO SCIENCE COMMITTEE Garmisch, Germany, 7th to 11th October 1968. January 1969.*
- [Cockburn00] Cockburn, Alistair. “Agile Software Development”, Cockburn*Highsmith Series Editors, 2000.
- [Doftman97]. Thayer, Richard H., y Merlin Dorfman, “Software requirements engineering”, 2da edición, IEEE Computer Society Press.1997.
- [Durán00] Toro Durán Amador: “Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información”, *Tesis Doctoral, septiembre 2000. Departamento de Lenguajes y Sistemas Informáticos Universidad de Sevilla.*
- [Estrada02] Estrada Hugo, Martínez Alicia, Pastor Oscar, Sánchez Juan. Trabajo Investigativo: “Generación de Especificaciones de Requisitos de Software a partir de Modelos de Negocios: un enfoque basado en metas”.Universidad Politécnica de Valencia. 2002
- [Fernández06] Fernández Sánchez. Ing. Leidy, Tesis de maestría: “Procedimiento para el desarrollo del proceso de ingeniería de requisitos en un proyecto software (PROCIR)”.2006.
- [Ferré05] Ferré Grau, Xavier; *Marco de Integración de la Usabilidad en el Proceso de Desarrollo de Software*. Tesis Doctoral. 2005.
- [Granollers04] Granollers Toni:“*MPLu+a. Una Metodología que integra la Ingeniería del Software, la interacción Persona-Ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares.*” *Tesis Doctoral. julio 2004*
- [Humphrey02] Humphrey Watts S, “*Introducción al Proceso de Software Personal (PSP)*”. Addison-Wesley. 2002.
- [IBM06] Sitio Web www.ibm.es 15/03/2007
- [IEEE90] IEEE Standard Glossary of Software Engineering Terminology. IEEE Standard 610.12–1990, Institute of Electrical and Electronics Engineers, 1990.
- [Jackson01] Jackson, M.: *Software Requirements and Specifications*, Addison-Wesley, 2001.

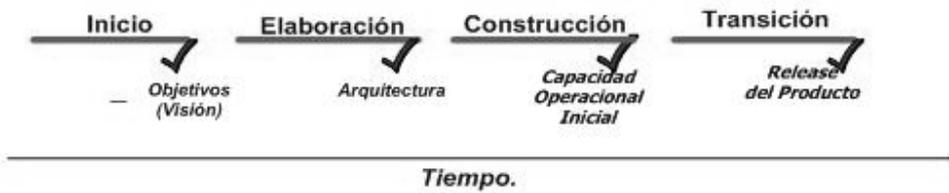
- [Jacobson00] Jacobson Ivar, Booch Grady, Rumbaugh James. “*El proceso Unificado de desarrollo de software*” Félix Varela, 2000, Tomo I.
- [Leffingwell03] Leffingwell. Dean, Widrig Don. “*Managing Software Requirements: A Use Case Approach*”, Second Edition. Addison Wesley, ISBN:0-321-12247-x, pages 544.
- [Matulevicius05] Matulevicius Raimundas. Tesis Doctoral. “Process Support for Requirements Engineering A Requirements Engineering Tool Evaluation Approach.” Trondheim, June 2005, Norwegian Universidad de Ciencia y Tecnología. Facultad de Información, Tecnología, Matemática y Ingeniería Eléctrica.
- [Mcdonald05] Mcdonald Landazuri Bárbara A. “*Definición de Perfiles en Herramientas de Gestión de Requisitos.*” Tesis Doctoral. Departamento de Lenguajes y Sistemas Informáticos e Ingeniería del Software. Facultad de Informática. Universidad Politécnica de Madrid, 2005.
- [Nuseibeh00] Nuseibeh, B. y Easterbrook, S. “*Requirements Engineering: A Roadmap*”. in *22nd International Conference on Software Engineering (ICSE'00)*. 2000. Limerick, Ireland: IEEE Computer Society Press.
- [Övergaard04] Gunnar Övergaard, Karin Palmkvist “*Use Cases Patterns and Blueprints*”. 2004.
- [Pardo04] Pardo Hernández, Eileen; de la Torres Oliva, Suny & Azcanio Hernández, Yania. “*Análisis y diseño de un sistema de gestión y control de almacén*” 2004.
- [Pressman97] Pressman, Roger S. “*Ingeniería del Software: Un enfoque práctico*. Cuarta Edición .McGraw-Hill. Madrid. 1997.
- [Pressman05] Pressman Roger S: “*Ingeniería del Software: Un enfoque práctico*”. Félix Varela. 2005. Vol I.
- [Rational03] Rational Unified Process®. Versión 2003.06.00.
- [Repo07] Disponible en:
<http://10.7.13.22:5801/svn/Inventario/Bibliografia/Tesis%20de%20Inventario/>
 12/04/2007.
- [Reuse06] The Reuse Company. “Presente y Futuro de la Reutilización de Software”. Última Modificación: 20/06/2006.
- [Salgado05] Salgado Febles. Dr. José E. “*Las TICS en el Sector del Turismo.*” Informática Aplicada. Mayo 2005.

- [Schenone04] Schenone Marcelo Hernán. *“Diseño de una Metodología Ágil de Desarrollo de Software.”* Tesis de Grado en Ingeniería en Informática. Facultad de Ingeniería. Universidad de Buenos Aires. 2004.
- [Sommerville04] Sommerville, Ian. *“Software Engineering”*, 7th edition, 2004 ppt <http://www.cs.st-andrews.ac.uk/~ifs/Books/SE7/Presentations/index.html>
- [SWEBOK04] Guide to the Software Engineering Body of Knowledge. IEEE Computer Society Order Number C2330, ISBN 0-7695-2330-7, Library of Congress Number 2005921729. 2004. <http://www.swebok.org>.
- [Thayer97] Thayer, R. H. y Dorfman, M., editores. *“Software Requirements Engineering”*. IEEE Computer Society Press, 2a edición, 1997. Es la 2a edición de [Thayer y Dorfman 1990].
- [Toval01] Toval Ambrosio, Nicolás Joaquín, Moros Begoña: *“SIREN: Un Proceso de Ingeniería de Requisitos Basado en Reutilización”*. Jornadas de Ingeniería de Requisitos Aplicada (JIRA). Sevilla, 11 y 12 de junio de 2001
- [Wieringa96] Wieringa, R. J. *“Requirements Engineering: Frameworks for Understanding”* JohnWiley & Sons, 1996.
- [Zave97] Zave, P., *Classification of Research Efforts in Requirements Engineering*. ACM Computing Surveys, 29(4): 315-321, 1997.

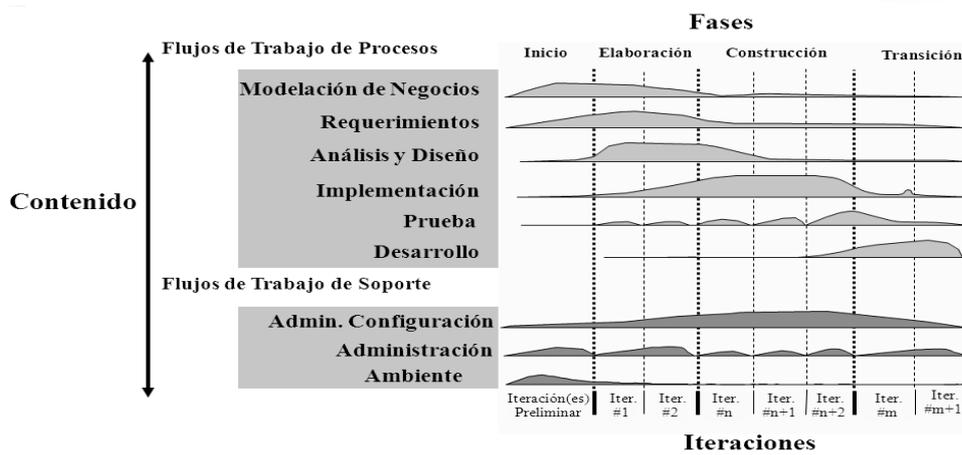
GLOSARIO DE TÉRMINOS

- Ajuste de Inventario (AI)** Ajustar inventario consiste en que al detectar productos deteriorados en el almacén, se le den baja del inventario, lo mismo con la pérdida de materiales o rotura.
- Despiece** Es el proceso contrario al escandallo, este consiste en de un producto obtener varios.
- Escandallo** Consiste en a partir de varios productos obtengo un producto elaborado Ejemplo: Un cake, para su elaboración es necesario la combinación de varios ingredientes: azúcar, harina, huevo, saborizantes, etc.
- Ficha Técnica (FT)** La ficha técnica no es más que la receta de un producto, esta puede ser libre o normada, en la primera solo estarán los productos que llevara el producto resultante, y en la segunda se especificara la medida de cada producto que se necesita.
- Informe de Recepción (IR)** El informe de recepción es un documento donde se recoge todos los datos sobre la entrada de productos al almacén.
- Modelo de Reclamación (R)** Este modelo consiste en un documento donde se llenan todos los datos cuando vas a hacerle una reclamación de uno o varios productos al proveedor.
- Nomenclar** El término nomenclatura, es un término que se emplea para indicar una manera particular de nombrar o clasificar cosas en una rama determinada de alguna materia. En el contexto de esta investigación se refiere a la definición de un concepto, ya sea crearlo, modificarlo, eliminarlo o mostrarlo.
- Stock Mínimo** Existencia mínima recomendada para el producto en el Almacén Central.
- Stock Máximo** Existencia máxima recomendada para el producto en el Almacén Central.
- Tarjeta Estiba (TE)** Esto no es más que una tarjeta que existe en el almacén para cada tipo de producto, en la ella se lleva el control del mismo, registrando aquí cada salida de él desde el almacén.

Anexo I Fases e Hitos (Milestones)

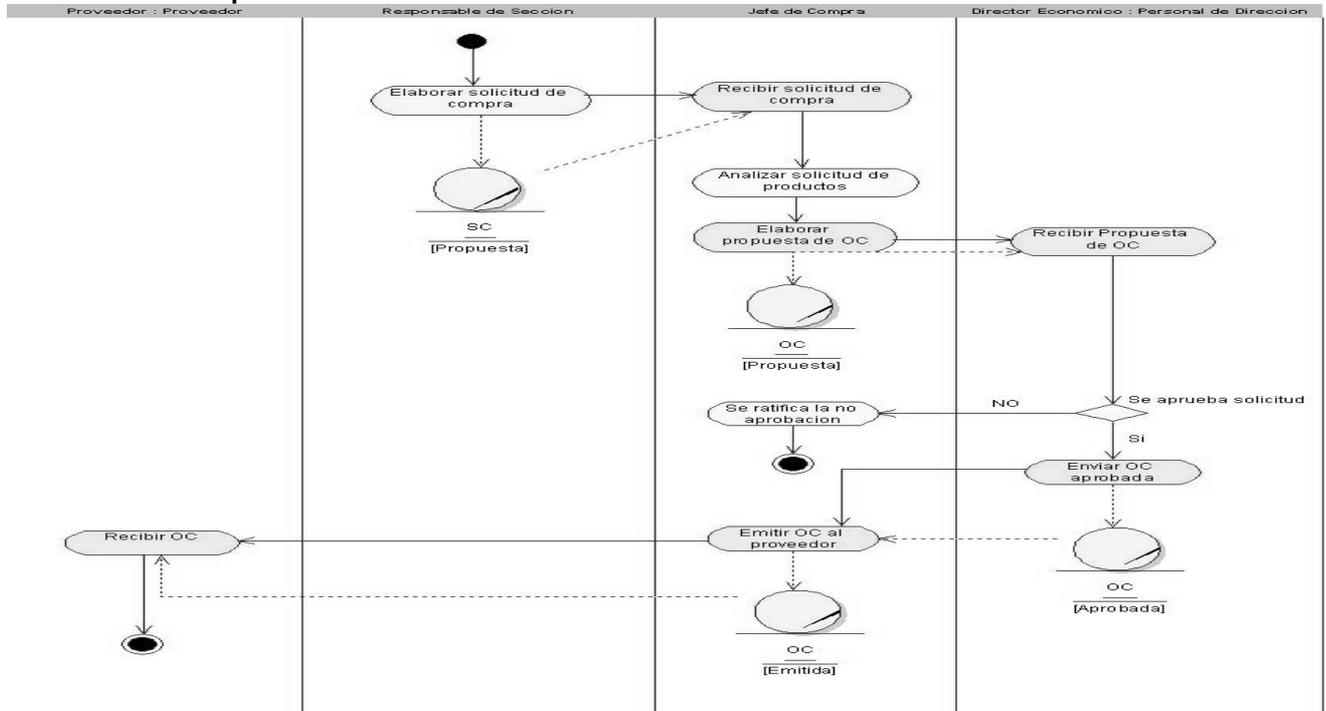


Anexo II Estructura de RUP

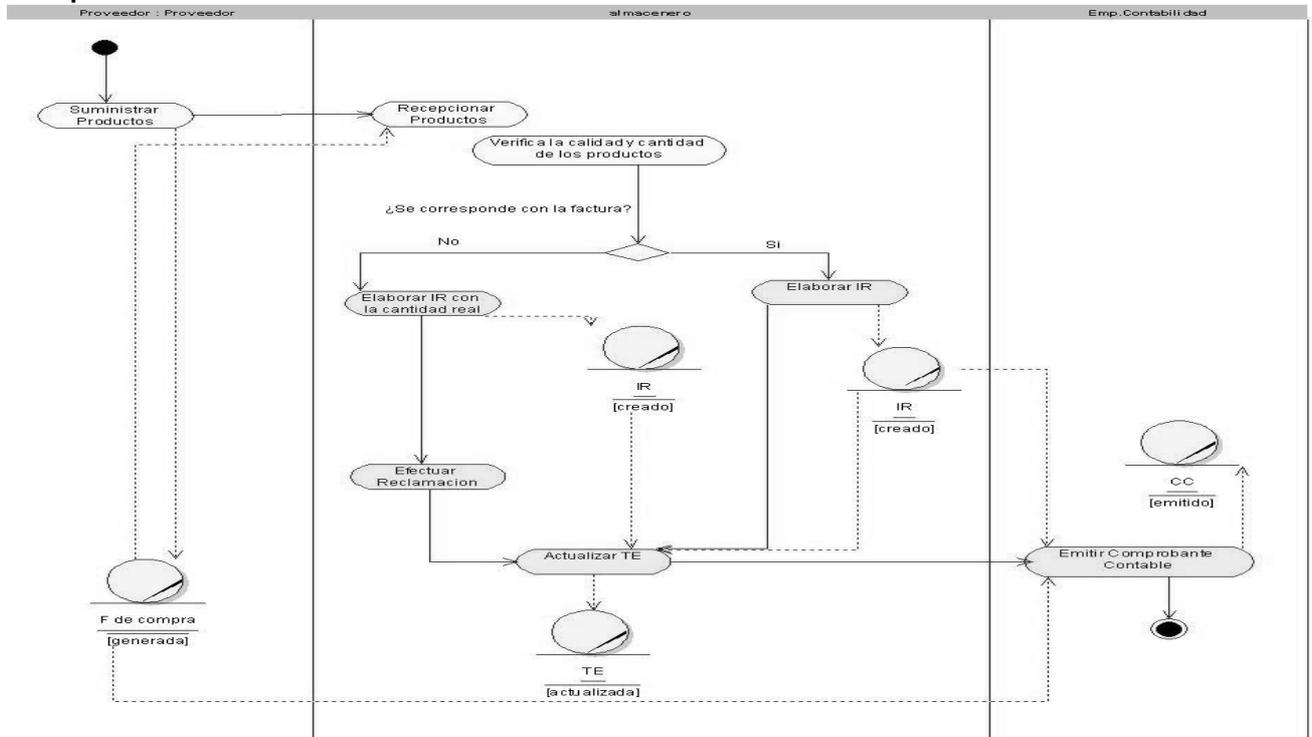


Anexo III Diagramas de Actividad de los Casos de Uso del Negocio

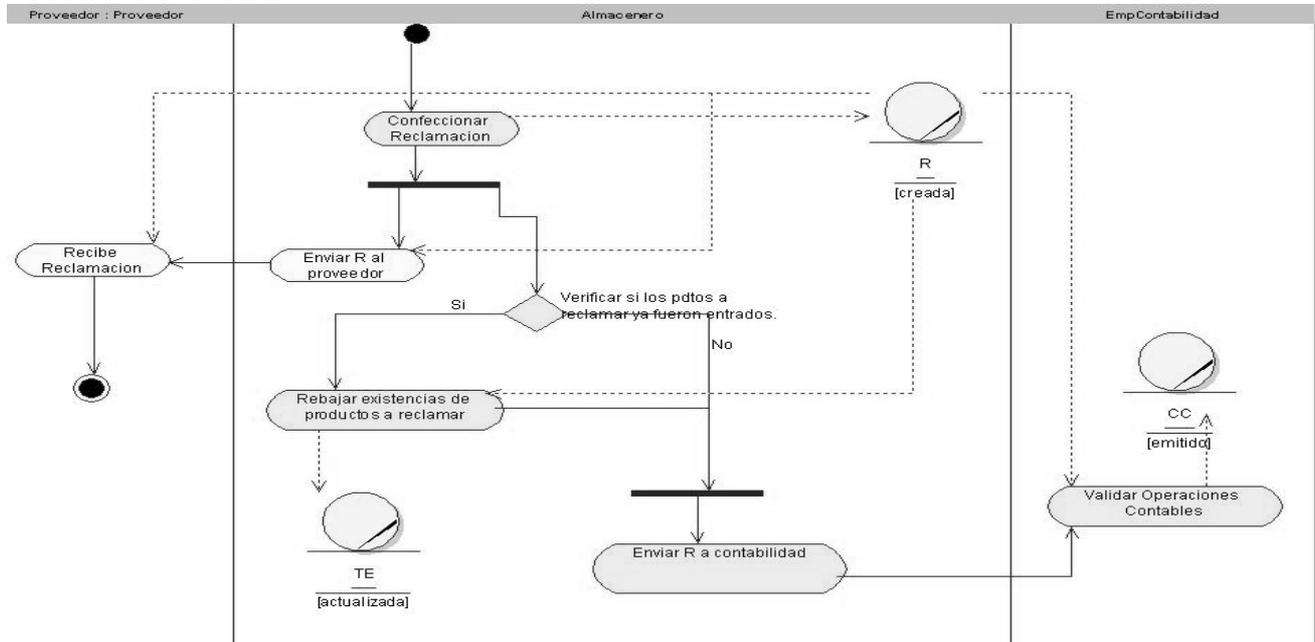
Emitir Orden Compra.



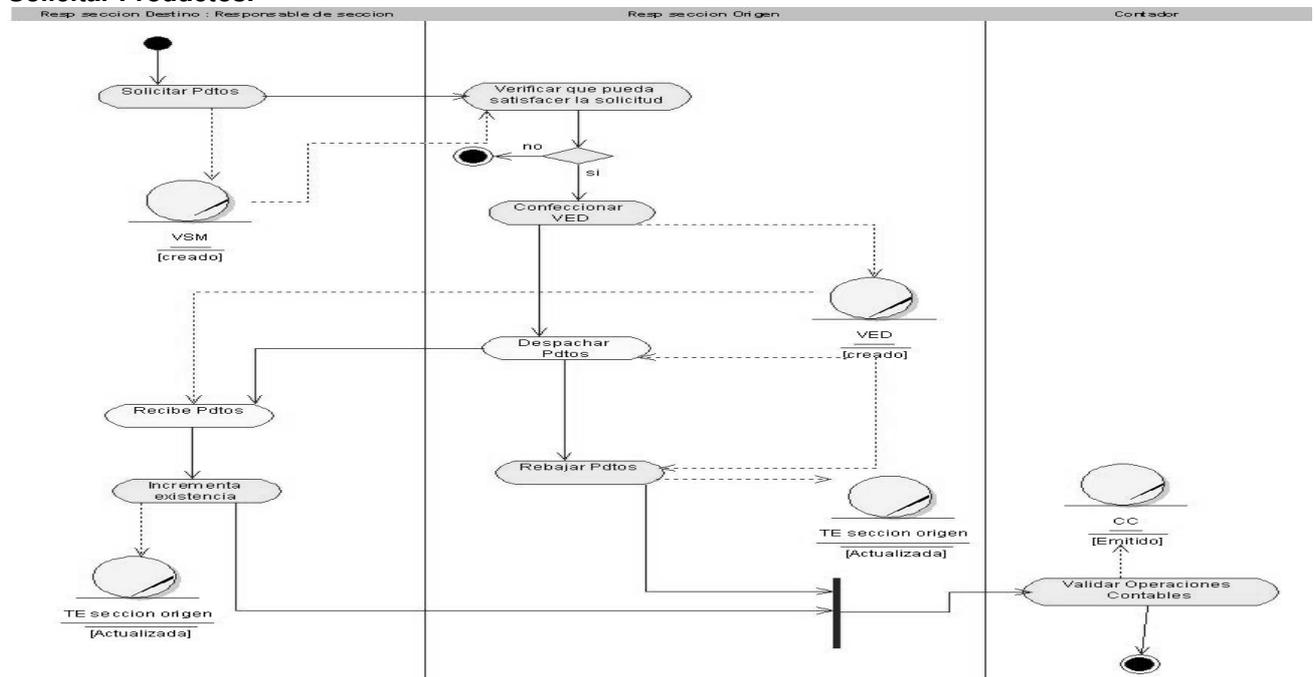
Recepcionar Productos suministrados.



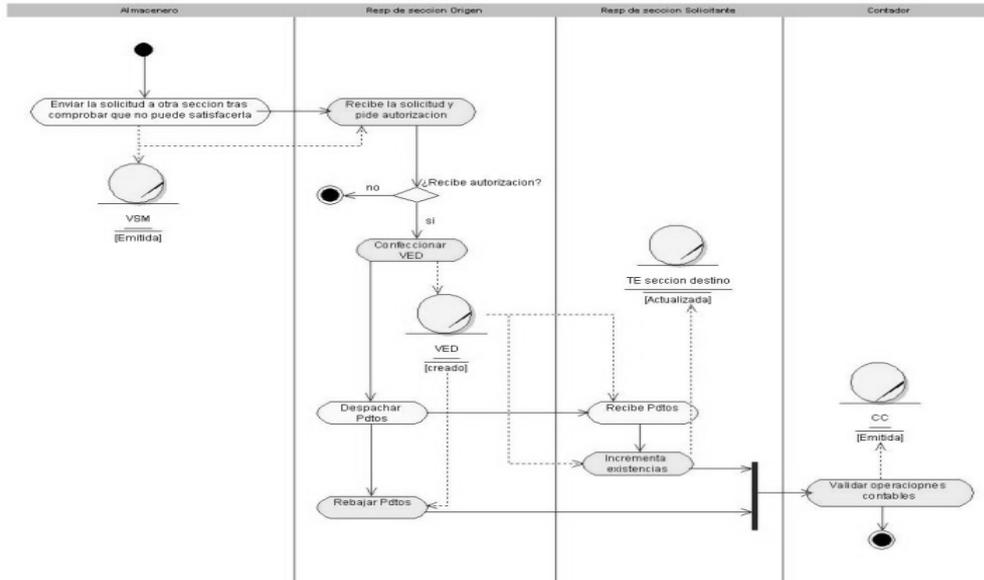
Reclamar al Proveedor.



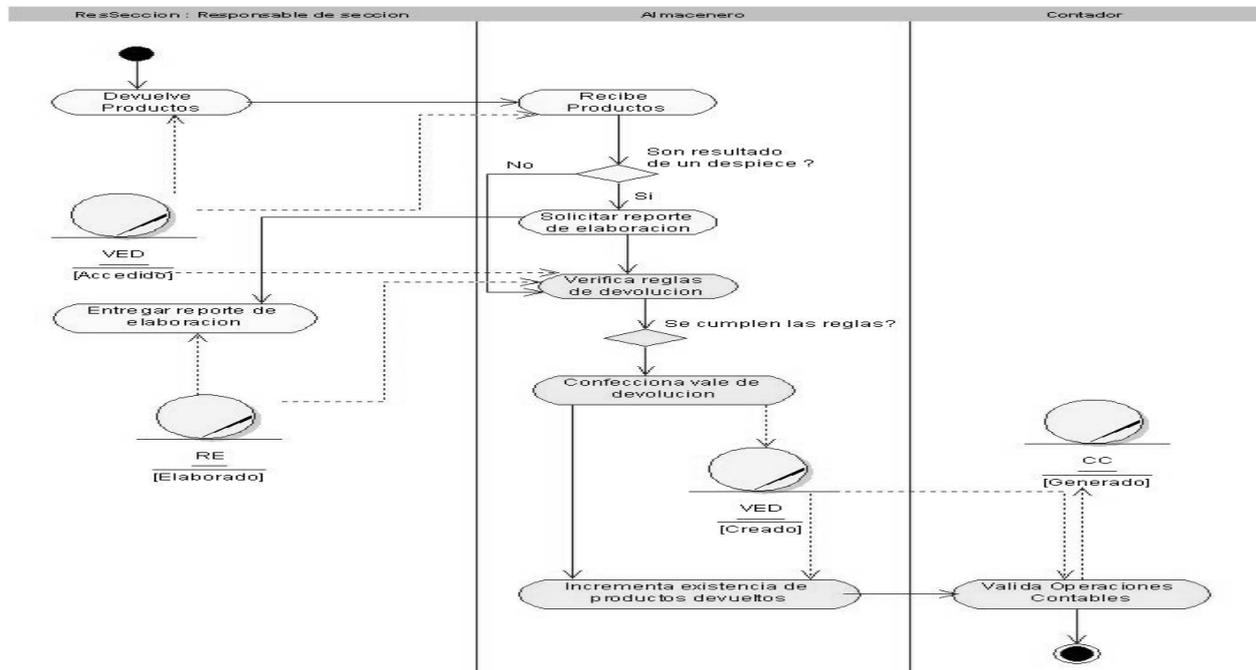
Solicitar Productos.



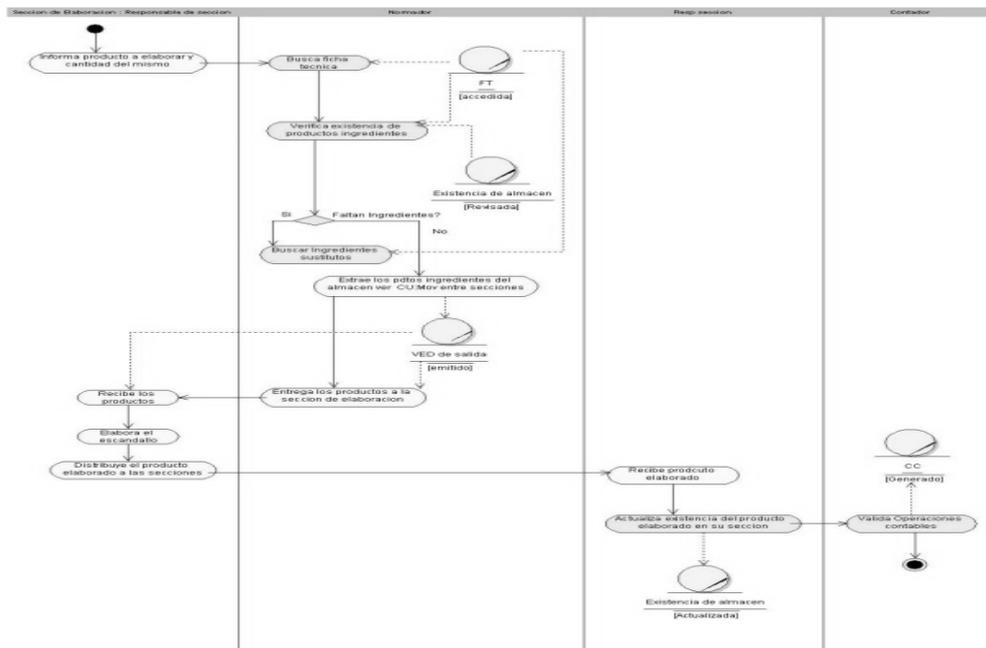
Realizar Movimiento entre secciones.



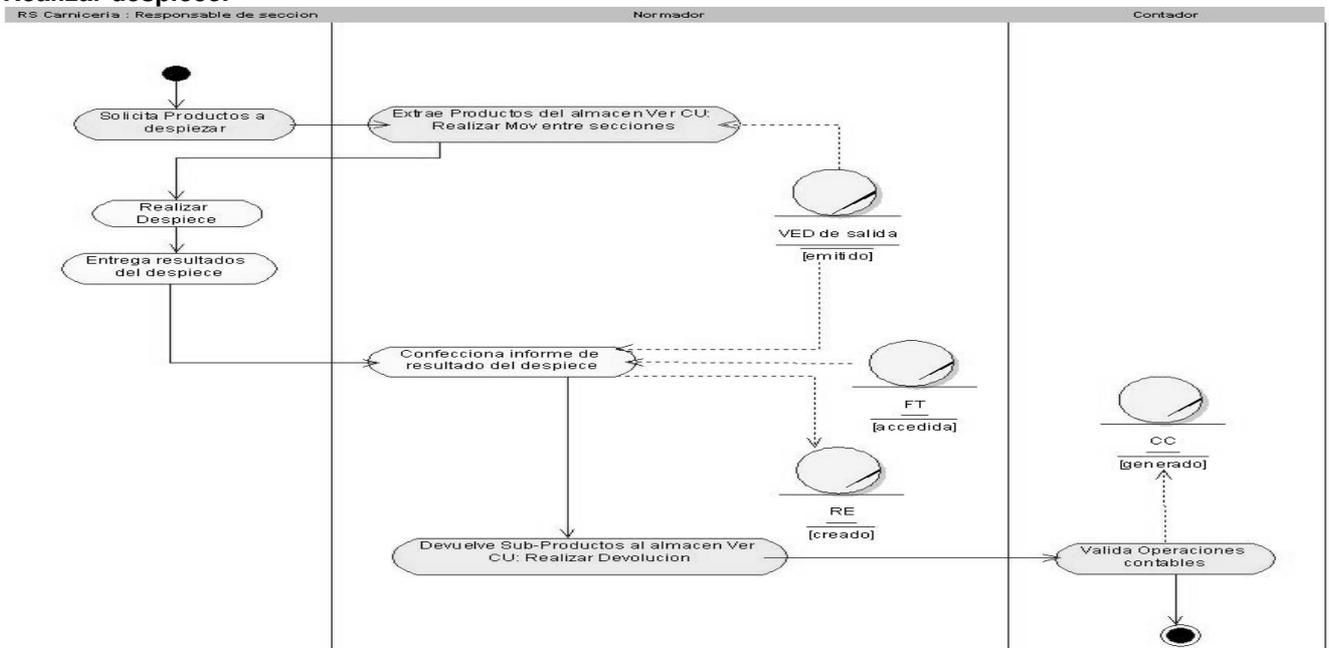
Realizar Devolución.



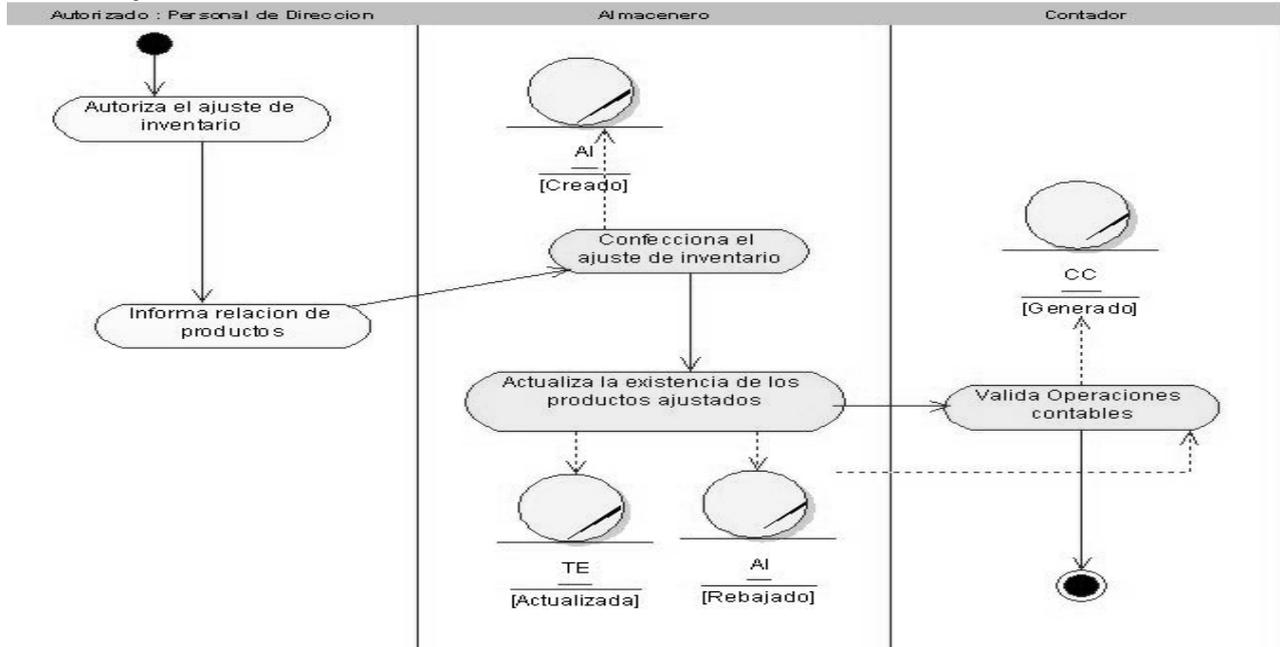
Elaborar escandallo.



Realizar despiece.



Realizar Ajuste de Inventario.



Realizar Venta.

