

Universidad de las Ciencias Informáticas
Facultad # 3



Título:
Ingeniería de Requerimientos aplicada al módulo
Registro y Aprobación del sistema SINAPSIS

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

Autor: Ariel E. Morales Malpica
Ornolis Vázquez Thompson

Tutor: Ing. Daylin María Fariñas González

Mayo, 2009

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores 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 _____.

Ariel Morales Malpica

(Autor)

Ornolis Vázquez Thompson

(Autor)

Daylin María Fariñas González

(Tutor)

AGRADECIMIENTOS

A nuestro Comandante en Jefe Fidel Castro, por ser el gran motor impulsor, en la realización de un sueño, la educación en Cuba.

A la UCI, por influir notablemente en mi formación como profesional y en la madurez de mis acciones.

A mis padres, abuelos, y a la familia en general, por la confianza depositada en mí, por ser mi razón de ser y la luz que me guía por el sendero correcto.

A todos los profes que desde edades muy tempranas colaboraron en mi formación, por el esfuerzo realizado durante toda mi vida estudiantil.

A mi tutora Dailyn, por el gran apoyo que nos ha dado en la realización de este trabajo.

A mi compañero de tesis Ariel, por permitirme desarrollar este trabajo a su lado, y por los años compartidos en la UCI.

A mis amigos, los viejos y nuevos, los que están conmigo desde la secundaria, y aquellos que conocí en la UCI.

A mis compañeros de brigada que compartieron conmigo los años en la universidad, desde el 3101 hasta el actual 3503.

A todos, los que de una forma u otra, han colaborado con la realización de este trabajo.

Omo

A mi tutora Daylin por el apoyo y constancia en la realización de este trabajo. Gracias por aguantar mis alteraciones e inconformidades. A Juan Carlos por compartir la tutoría junto a Daylin y por sus consejos.

A mi compañero de tesis Ornelis por darme la oportunidad de compartir este trabajo tan importante juntos.

Al amigo Polo por estar presente en los momentos buenos y malos. Por ser amigo de mis padres y tenerle ese cariño tan especial a mi mamá. Gracias por la atención que me diste, al igual que Nel, nunca lo olvidaré.

A Tía Coralia, Tío Chucho y todos mis primos por acogerme durante estos cinco años en esa familia tan alegre y buena que han formado.

A Tery, Manuel y Rene, gracias por hacerme uno más de la familia. Por aguantarme cada fin de semana. Por pasar tantos cumpleaños, comidas, playas juntos. Gracias por ser tan especiales.

A Ary por ser mi novia y amiga, por aguantarme tantos berrinches en la realización de este trabajo, por aconsejarme y corregirme y por tu amor, apoyo y comprensión.

A mis padres por depositar tanto amor y confianza en mí, por enseñarme a luchar por mis sueños y estar pendientes en todo momento por la realización de mi tesis.

Ariel

DEDICATORIA

A mis padres Fela y Tato, quienes han sido la mayor inspiración de este trabajo y han hecho de mi un hombre de bien, llevándome por el camino correcto. Los amo mucho.

A mis abuelos Chelo, Nieves y Roberto, por su gran confianza en mí y darme ánimos para seguir adelante. Son lo más grande de mi vida.

A mi hermanito Rafelito, para el cual, quiero ser el mejor ejemplo del mundo.

A mi familia que siempre está pendiente de mí, llenándome de fe para afrontar cualquier reto.

A mi amorcito Daniar, por darme su amor incondicional en los momentos buenos y malos y por convertirme en una mejor persona. Te adoro.

A los amigos que hoy se encuentran aquí, y a los que lamentablemente no pudieron estar, por darme su apoyo en los momentos que más necesité.

A mi brigada 3503, por los momentos inolvidables vividos en el aula y en la producción.

Omo

A mi abuela Mima Cacha por darnos la familia tan especial que tenemos.

A Tío Tito y Tío Jorge por ser tíos, padres y amigos.

A Tatá y Tía Uca por ser tan incondicional en las buenas y malas.

A mis primos Franco, Yuli y Arianna y a mi hermano Jose por ser mis preferidos. Espero que le llegue a muchos de uds. este gran momento.

A mi papá por el apoyo y confianza durante todos mis estudios, mis ideas e inventos que a cada rato se me ocurren. Por enseñarme varias formas de ver la vida, salir adelante y triunfar.

A mi mamá por ser lo más grande que tengo en este mundo. Por ser mi mayor inspiración. Espero que la vida te regale la oportunidad de verte mencionada en la dedicatoria de tesis de tus nietos. Por fin lo logramos, este triunfo también es tuyo.

Ariel

RESUMEN

El presente trabajo aborda la aplicación de la Ingeniería de Requerimientos al proceso de Registro y Aprobación de Proyectos que se lleva a cabo en la administración pública de la República Bolivariana de Venezuela, como parte de la gestión de proyectos que se realiza todos los años en este país. A partir de este proceso se planifican y aprueban los proyectos que serán llevados a cabo, así como el presupuesto con que contará Venezuela para el año próximo, de ahí la importancia que implica su automatización de manera efectiva. El propósito trazado consistió en el desarrollo de los artefactos de software que influyen en la transformación de las necesidades de los clientes y usuarios en requerimientos de software. Para ello fue necesario analizar y especificar los requerimientos identificados previamente. Además, se aplicaron métricas combinadas con un conjunto de listas de chequeo con el objetivo de verificar la calidad de los principales artefactos obtenidos. Los clientes quedaron satisfechos con los servicios brindados, afirmación que se pudo constatar con la aplicación del método de Kano para medir la satisfacción de los mismos.

PALABRAS CLAVE

Ingeniería de Requerimientos, artefactos de software, métricas, método de Kano.

ÍNDICE DE FIGURAS

Figura 1 Flujos de Trabajo y Fases de RUP	18
Figura 2 Flujos de Trabajo y Fases de AUP	20
Figura 3 Elementos gráficos y categorías de un BPD	26
Figura 4 Diagrama del modelo ACSI	44
Figura 5 Clasificación de los requerimientos con el método de Kano.	45
Figura 6 Diagrama del proceso de negocio Formular propuesta	53
Figura 7 Porción del Diagrama de requerimientos del caso de uso Gestionar proyecto	61
Figura 8 Paquetes del modulo Registro y Aprobación de Proyectos	62
Figura 9 Diagrama de casos de uso del paquete Funciones básicas.....	63
Figura 10 Diagrama de casos de uso del paquete Formulación.....	63
Figura 11 Diagrama de casos de uso del paquete Recursos financieros	63
Figura 12 Diagrama de casos de uso del paquete Planificación	64
Figura 13 Diagrama de casos de uso del paquete Revisión	64
Figura 14 Formato de encuesta. Captura de pantalla	75
Figura 15 Clasificación de los requerimientos en función de los resultados obtenidos	77
Figura 16 Diagrama de requerimientos de los casos de uso Actualizar datos presupuestarios, Asignar recursos financieros a Entes y Asignar recursos financieros a Proyectos	105
Figura 17 Diagrama de requerimientos de los casos de uso Aprobar propuesta y Aprobar proyecto.....	106
Figura 18 Diagrama de requerimientos del caso de uso Buscar proyectos y propuestas.....	106
Figura 19 Diagrama de requerimientos del caso de uso Gestionar acción específica	107
Figura 20 Diagrama de requerimientos del caso de uso Gestionar actividad	108
Figura 21 Diagrama de requerimientos de los casos de uso Mostrar enviados a revisión y Mostrar proyectos aprobados	108
Figura 22 Diagrama de requerimientos de los casos de uso Gestionar distribución territorial regional, Gestionar distribución territorial internacional y Gestionar distribución territorial nacional	109

Figura 23 Diagrama de requerimientos de los casos de uso Gestionar imputación presupuestaria y Gestionar asignación de fuente de financiamiento	110
Figura 24 Diagrama de requerimientos de los casos de uso Gestionar indicador inicial y Gestionar indicador esperado.....	111
Figura 25 Diagrama de requerimientos de los casos de uso Gestionar propuesta y Gestionar proyecto	112
Figura 26 Paquete Requerimientos auxiliares	113

CONTENIDO

INTRODUCCIÓN	12
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA.....	17
1.1 INTRODUCCIÓN	17
1.2 METODOLOGÍAS DE DESARROLLO DE SOFTWARE	17
1.3 HERRAMIENTAS DE DESARROLLO DE SOFTWARE.....	22
1.4 LENGUAJES DE MODELADO.....	25
1.6 HERRAMIENTAS PARA EL MODELADO DE PROCESOS	29
1.7 INGENIERÍA DE REQUERIMIENTOS	31
1.7.1 ÉTAPAS FUNDAMENTALES.....	33
1.7.2 TÉCNICAS UTILIZADAS EN LA INGENIERÍA DE REQUERIMIENTOS	38
1.8 MÉTRICAS DE SOFTWARE	41
1.9 MÉTODOS PARA LA MEDICIÓN DE LA SATISFACCIÓN DEL CLIENTE	42
1.10 PATRONES DE CASOS DE USO.....	47
1.11 CONCLUSIONES.....	50
CAPITULO 2 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	51
2.1 INTRODUCCIÓN	51
2.2 MODELADO DE PROCESOS.....	51
2.3 REGLAS DEL NEGOCIO	53
2.4 REQUERIMIENTOS	54
2.2.1 REQUERIMIENTOS FUNCIONALES.....	55
2.2.2 REQUERIMIENTOS NO FUNCIONALES.....	58
2.2.3 ADMINISTRACIÓN DE REQUERIMIENTOS.....	59
2.5 ACTORES DEL SISTEMA.....	61
2.6 DIAGRAMAS DE CASOS DE USO DEL SISTEMA.....	62
2.7 ESPECIFICACIÓN DE CASOS DE USO DEL SISTEMA	64
2.8 CONCLUSIONES.....	66
CAPITULO 3 ANÁLISIS DE LOS RESULTADOS.....	68
3.1 INTRODUCCIÓN	68
3.2 VERIFICACIÓN DE REQUERIMIENTOS.....	68
3.3 VALIDACIÓN DE REQUERIMIENTOS.....	73
3.4 MEDICIÓN DEL GRADO DE SATISFACCIÓN DEL CLIENTE.....	73
3.5 CONCLUSIONES.....	78

CONCLUSIONES GENERALES	79
RECOMENDACIONES	80
BIBLIOGRAFÍA	81
ANEXOS	85
GLOSARIO	119

INTRODUCCIÓN

La República Bolivariana de Venezuela desde el primero de enero de 2006 lleva a cabo el presupuesto por proyectos por disposición de su Presidente, el Comandante Hugo Rafael Chávez Frías en julio del año 2005. A través del presupuesto el gobierno proporciona bienes y servicios para atender las necesidades de la comunidad; cancela servicios que permiten su funcionamiento, paga a proveedores y contratistas, resuelve problemas de pasivos laborales y contractuales con sus trabajadores.

Para llevar a cabo la gestión del presupuesto por proyectos, el Ministerio del Poder Popular para la Planificación y Desarrollo (MPPPD), conjuntamente con el Ministerio de Economía y Finanzas y la Vicepresidencia de la República, decidieron implantar un sistema informático que facilitara el registro, seguimiento y control de todos los proyectos que formarían parte del presupuesto anual de la nación, denominado Nueva Etapa.

A través del sistema antes mencionado se registran todos los proyectos que son formulados y planificados por todos los entes, tanto centralizados como descentralizados. La suma del presupuesto de un año de los proyectos aprobados por la máxima dirección de la administración pública es lo que se conoce como presupuesto anual.

Nueva Etapa no fue desarrollado utilizando una metodología de desarrollo de software, y no cuenta con la documentación necesaria que facilite la comprensión y optimización de dicho sistema. A pesar de ser una aplicación web no está disponible para sus usuarios durante todo el año, por lo que se formulan y planifican los proyectos con estimaciones de costos y tiempos irreales.

Se puede afirmar que actualmente Nueva Etapa no satisface todas las expectativas de los clientes y usuarios, dado que no existe un proceso homogéneo de registro y aprobación de proyectos para todos los órganos y

entes centralizados y descentralizados. Además, no están bien definidos los flujos y niveles de aprobación por los que transita un proyecto. Venezuela es un país revolucionario, en constante transformación, debido a esto, las necesidades de la administración pública son cada vez más exigentes.

Para darle solución a esta problemática se decidió desarrollar el Sistema Nacional Público para el Seguimiento de Inversiones y Sectores (SINAPSIS). Un sistema informático que sustituirá completamente el anterior y se dividirá en 7 módulos, entre los que se encuentra el de Registro y Aprobación de Proyectos.

En el módulo antes mencionado, para minimizar la complejidad y heterogeneidad presentes en el proceso de Registro y Aprobación de Proyectos, se hace necesario, una vez que se comprenda el proceso completamente, obtener un conjunto de requerimientos de software que satisfagan las necesidades de los clientes y usuarios.

El conjunto de requerimientos identificados y correctamente especificados, permitirá una mejor comprensión, tanto de clientes y usuarios como de los desarrolladores, de las funcionalidades y características que estarán presentes en el módulo Registro y Aprobación de Proyectos. Además, ayudará en gran medida a minimizar el impacto de los cambios en las posteriores fases de diseño e implementación del sistema y los clientes quedarán satisfechos con el servicio brindado.

Problema científico

¿Cómo transformar las necesidades del cliente en los requerimientos de software del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS, para garantizar la satisfacción de dicho cliente?

Objeto de estudio

Proceso de Desarrollo de Software.

Campo de acción

Ingeniería de requerimientos del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS.

Objetivo general

Desarrollar los artefactos de software que influyen en la transformación de las necesidades del cliente en los requerimientos de software del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS, para garantizar la satisfacción de dicho cliente.

Hipótesis

Si se desarrollan los artefactos de software que influyen en la transformación de las necesidades del cliente, en los requerimientos de software del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS, entonces se garantizará la satisfacción de dicho cliente.

Tareas investigativas

1. Realizar un estudio de las tendencias actuales sobre el análisis y especificación de requerimientos y tomar posición al respecto.
2. Obtener un listado de todas las necesidades del cliente, que contribuya a identificar los requerimientos de software.
3. Realizar el control y seguimiento de los requerimientos como una manera de mitigar el impacto del cambio en los mismos
4. Clasificar los requerimientos de software y eliminar todas las inconsistencias y ambigüedades que los mismos puedan tener.
5. Realizar la especificación de los requerimientos de software para presentarlos de manera consistente y comprensible.
6. Aplicar métricas de requerimientos de software para evaluar la calidad de los artefactos generados y la satisfacción del cliente.

Métodos y técnicas de investigación a utilizar

Métodos teóricos:

Histórico - Lógico: Para el análisis de la trayectoria de la Ingeniería de Requerimientos y para expresar teóricamente sus elementos fundamentales.

Hipotético - Deductivo: Para la realización de la hipótesis.

Modelación: Para la creación de modelos como abstracciones, con el objetivo de explicar cómo funcionan el proceso de registro y aprobación de proyectos.

Métodos empíricos:

Entrevista: Para la obtención de información sobre los procesos de registro y aprobación de proyectos.

Encuesta: Para la medición de la satisfacción del cliente en función de los requerimientos del sistema.

Estructura de la tesis

El presente trabajo está compuesto por tres capítulos, los cuales están estructurados de la siguiente manera:

En el capítulo 1 se hace un estudio de diferentes metodologías para el desarrollo de software, los lenguajes de modelado y herramientas que pueden utilizarse para modelar los artefactos que proponen dichas metodologías. Se abordan las etapas de la Ingeniería de Requerimientos. Se presentan patrones de casos de uso, métricas y métodos para la medición de la satisfacción del cliente.

En el capítulo 2 se modelan los procesos del negocio haciendo uso de los Diagramas de Procesos de Negocio, que propone BPMN. Se presentan las restricciones o políticas a cumplir por el sistema, un conjunto de requerimientos funcionales y no funcionales, así como la matriz de trazabilidad y diagramas de requerimientos. Se muestran además los actores y diagrama de casos de uso del sistema y la descripción de los mismos.

En el capítulo 3 se lleva a cabo la etapa de verificación y validación de requerimientos. Para la primera de las etapas enunciadas, se aplican métricas para medir la calidad de los requerimientos identificados. Para la segunda, se aplica la técnica de *walkthroughs* combinada con los prototipos de interfaz para conseguir la aceptación por parte de los clientes y usuarios. Por último, se analizan y presentan los resultados de aplicar el método de Kano para medir la satisfacción del cliente.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

El presente capítulo aborda un estudio del estado del arte de las tendencias actuales en cuanto a metodologías, herramientas y patrones de casos de uso utilizados para el análisis de un proyecto software. Abarcando también lenguajes y herramientas de modelado de procesos de negocios, pues, un buen entendimiento de dichos procesos, facilitará un adecuado desarrollo de los requerimientos de software.

Se muestra una vista general de la Ingeniería de Requerimientos, disciplina fundamental de la Ingeniería de Software, así como métricas que ayudarán a determinar si los requerimientos han alcanzado un estado óptimo. Se aborda además sobre métodos para medir la satisfacción del cliente.

1.2 Metodologías de desarrollo de software

Una metodología de desarrollo de software es un conjunto de pasos y procedimientos que deben seguirse para desarrollar software. Su objetivo radica en convertir el desarrollo de software en un proceso formal, con resultados predecibles, que permitan obtener un producto final de alta calidad. La elección de una metodología u otra, está fuertemente determinada por las características del proyecto de software y el equipo de desarrollo, es importante que sea fácil de comprender y de aplicar, además de aportar beneficios tangibles al proyecto.

Proceso Unificado de Desarrollo

El Proceso Unificado de Desarrollo (RUP, Rational Unified Process) es un proceso de desarrollo de software. RUP provee un marco de trabajo genérico que puede especializarse para una gran cantidad de sistemas software.

Se caracteriza por ser:

- **Iterativo e incremental:** Cada fase se desarrolla en iteraciones. Se divide el trabajo en partes más pequeñas o mini-proyectos, donde cada uno es una iteración que resulta en un incremento.

- Centrado en la arquitectura: La arquitectura describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.
- Guiado por los 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. (1)

Como RUP es un proceso define “quién” está haciendo “qué”, “cuándo” y “cómo” para alcanzar un determinado objetivo. (2)

En RUP se han agrupado las actividades en grupos lógicos definiéndose nueve flujos de trabajo principales. Los seis primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo. En la figura 1 se representa el proceso en el que se grafican los flujos de trabajo y las fases y muestra la dinámica expresada en iteraciones y puntos de control.

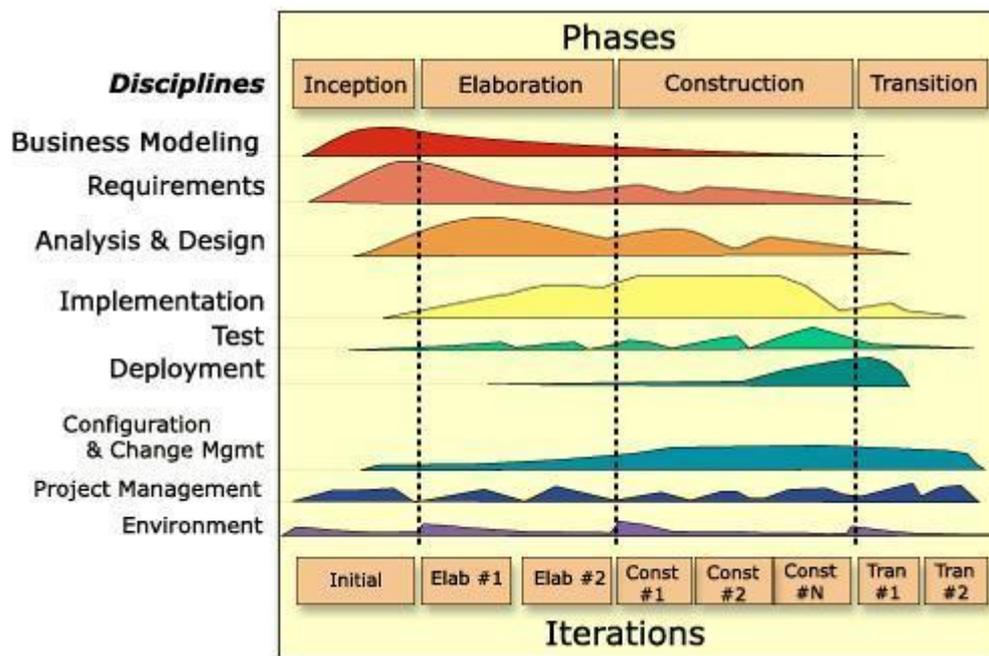


Figura 1 Flujos de Trabajo y Fases de RUP

Cada disciplina describe un grupo de actividades asociadas y artefactos en torno a habilidades comunes. (3)

- Modelamiento del negocio
- Requerimientos
- Análisis y diseño
- Implementación
- Prueba (Testeo)
- Instalación
- Gestión de proyecto
- Gestión de configuración y cambios
- Ambiente

El ciclo de vida del software de RUP es descompuesto en cuatro fases secuenciales, cada una posee varios hitos. Al final de cada fase se realiza una evaluación para determinar que los objetivos de la fase han sido cumplidos. Una evaluación satisfactoria permite al proyecto avanzar a la proxima fase. (4)

- Conceptualización (Concepción o Inicio)
- Elaboración
- Construcción
- Transición

Proceso Unificado Ágil

El Proceso Unificado Ágil (AUP, Agile Unified Process) es una versión simplificada del Proceso Unificado de Desarrollo (RUP), creada por Scott W. Ambler (5). Esta metodología describe de una manera simple y fácil de entender, la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP.

AUP aplica Desarrollo Guiado por Pruebas (Test Driven Development - TDD), Desarrollo Guiado por Modelado Ágil (Agile Model Driven Development -

AMDD), Gestión de Cambios Ágil, y refactorización de base de datos para mejorar la productividad.

Establece un Modelo más simple que el que aparece en RUP por lo que reúne en una única disciplina las disciplinas de Modelado de Negocio, Requerimientos y Análisis y Diseño. (6)

La disciplina de Configuración y Gestión de Cambios es ahora la disciplina Gestión de Configuración. En desarrollo ágil, las actividades de gestión de cambios son típicamente parte de los esfuerzos de gestión de requerimientos, la cual es parte de la disciplina Modelamiento. En la figura 2 se grafican los flujos de trabajo y las fases que propone la metodología.

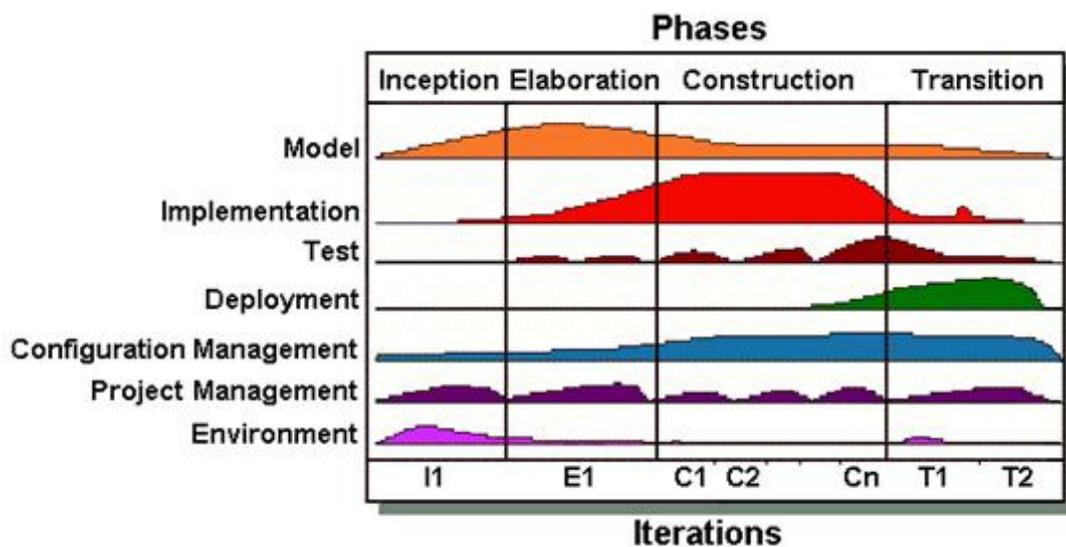


Figura 2 Flujos de Trabajo y Fases de AUP

Las disciplinas de AUP son las siguientes (7):

- Modelamiento
- Implementación
- Prueba
- Despliegue
- Gestión de configuración

- Gestión de proyecto
- Ambiente

Programación Extrema

La Programación Extrema (XP, Extreming Programming) es actualmente un enfoque deliberado y disciplinado para el desarrollo de software. Enfatiza la satisfacción del cliente y promueve el trabajo en equipo. (8)

Se distingue de otras metodologías por su confianza en pruebas automatizadas escritas por programadores y clientes para monitorear el progreso del desarrollo, haciendo posible la posterior evolución del sistema y la captura de defectos tempranamente. Promueve la comunicación oral, permitiendo una estrecha colaboración de programadores con habilidades ordinarias. Está diseñada para trabajar con proyectos que pueden ser construidos por equipos de dos a diez personas. (9)

Análogamente a los casos de usos en otras metodologías, XP define las historias de usuario, que son escritas por los propios clientes, tal y como ven ellos las necesidades del sistema. Proporcionan los detalles sobre la estimación del riesgo y cuánto tiempo conllevará la implementación de dicha historia de usuario. Son similares al empleo de escenarios, con la excepción de que no se limitan a la descripción de la interfaz de usuario.

También conducirán el proceso de creación de las pruebas de aceptación. (10)

Entre las prácticas principales de esta metodología, se encuentran:

- Diseño simple: El sistema debería ser diseñado tan simple como sea posible en cualquier momento dado.
- Pruebas de unidad: Los programadores continuamente escriben pruebas de unidad, las cuales deben correr perfectamente para que el desarrollo continúe.
- Programación en pares: Todo el código es escrito por dos programadores en una máquina.

- Re-fabricación: Los desarrolladores reestructuran el sistema sin cambiar su comportamiento, para agregar simplicidad y flexibilidad al código.

El ciclo de vida de un proyecto XP ideal consta de las siguientes fases: (9)

- Planeación
- Iteraciones
- Producción
- Mantenimiento
- Muerte

1.3 Herramientas de desarrollo de software

Las herramientas CASE (Computer Aided Software Engineering) son diversas aplicaciones informáticas que permitirán organizar y manejar la información de un proyecto de software. Permitiendo que los sistemas (especialmente los complejos), se tornen más flexibles y comprensibles. Con estas herramientas los analistas serán capaces de representar complejos procesos de negocios.

Visual Paradigm

Visual Paradigm for UML (VP-UML), es una poderosa herramienta CASE para UML, designada para un amplio número de usuarios, incluyendo ingenieros de software, analistas de sistemas, analistas del negocio, arquitectos de sistemas y otros, interesados en construir sistemas de software usando el enfoque Orientado a Objetos.

Una de sus características principales es la interfaz de recursos céntricos, que permite el acceso a las opciones de modelado fácilmente, sin necesidad de desplegar el menú o ir a la barra de herramientas. Esta interfaz asegura que el diseñador está creando los diagramas con la sintaxis correcta más rápidamente (11). Por otra parte su integración con entornos de desarrollo como Eclipse, facilita la generación de código y la ingeniería inversa.

VP-UML soporta más de 20 tipos de diagramas incluyendo especificaciones como UML, BPMN, SysML, ERD y más. Además provee chequeo de la sintaxis

de UML. Incorporando gestos del mouse dentro del modelado, permite a los usuarios dibujar elementos y representar acciones.

Para diagramas complejos, VP-UML posee una opción que permite ordenar los elementos del diagrama seleccionando la más acorde automáticamente, se denomina disposición automática (auto layout).

Permite la generación de reportes a formatos como HTML, PDF, Word y otros, como parte de la documentación los diagramas. Project Publisher exporta el proyecto a páginas web interactivas que pueden ser leídas por cualquier navegador sin necesidad de *plug-in*. Report Writer puede extraer datos de los modelos, organizarlos y generar un reporte sistemático.

La herramienta facilita una excelente interoperabilidad mediante el intercambio de diagramas UML y modelos vía XMI. Además se puede importar diagramas creados previamente por IBM Rational Rose y ERWin.

La facilidad de Instant Reverse hace posible aplicar ingeniería inversa a diferentes tipos de ficheros fuentes y binarios y llevarlos al modelado, inversa a esta característica se encuentra la facilidad Instant Generator que permite generar código a partir de los diagramas, soportando varios lenguajes como Java, C#, VB.NET, PHP, ActionScript, C++, Python. Es una herramienta con licencia comercial y en la Universidad de las Ciencias Informáticas se tiene permisos para su utilización.

Enterprise Architect

Enterprise Architect (EA) es una herramienta CASE para el diseño y construcción de sistemas de software. EA soporta la especificación de UML 2.0, que describe un lenguaje visual por el cual se pueden definir mapas o modelos de un proyecto. (12)

Es una herramienta que abarca integralmente el ciclo de vida, cubriendo el desarrollo de software desde la captura de los requerimientos, a través de las

etapas de análisis, modelos de diseño, prueba y finalmente el mantenimiento y re-uso. (13)

Algunas de las características fundamentales es que permite crear todos los diagramas y modelos especificados en UML para un amplio alcance de objetivos. Es posible organizar los elementos en diagramas y paquetes. Permite además documentar los elementos que se han creado y la generación de código para el software que se está construyendo. Es capaz de realizar la ingeniería inversa a partir de un código existente en varios lenguajes. Además, es una herramienta con licencia comercial de la cual no se tiene permisos en la Universidad de las Ciencias Informáticas.

Rational Rose

Es una de las más poderosas herramientas de modelado visual para el análisis y diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de construirlo. Cubre todo el ciclo de vida de un proyecto.

Rational Rose permite la generación de código automático a partir de modelos e ingeniería inversa, a partir del código genera el modelo, así como mantiene la sincronización entre el modelo y el código generado. Además es perfectamente integrable con cualquiera de las otras herramientas de desarrollo de software de IBM, como por ejemplo ClearCase, una herramienta de control de versiones de IBM.

Rational Rose además ofrece modelado UML para el diseño de bases de datos, con capacidad para representar a la integración de los datos y requerimientos de las aplicaciones a través de diseños lógicos y físicos. Actualmente solamente está soportado para Windows (14). Es una herramienta con licencia comercial de la cual, no se tiene permisos para su utilización en la Universidad de las Ciencias Informáticas

1.4 Lenguajes de modelado

Un modelo de proceso de negocio describe cómo funciona el negocio, es decir, describe las actividades involucradas en el negocio y la manera en que se relacionan unas con otras e interactúan con los recursos necesarios para lograr la meta del proceso.

Los modelos de procesos de negocio (MPN) tienen un amplio rango de usos tales como el soporte a la re-ingeniería de procesos, la simulación o sirven como base para el desarrollo de sistemas que automatizan dichos procesos. Los MPN pueden ser creados o presentados usando diversos lenguajes, que son bastante diferentes entre sí, dado que cada uno tiene una manera diferente de ver los procesos dependiendo del propósito para el cual fueron creados.

Notación de Modelado de Procesos de Negocio

La Notación de Modelado de Procesos de Negocio (BPMN, Business Process Modeling Notation) es un estándar de notación desarrollado por Business Process Management Initiative (BPMI). Esta especificación, representa más de dos años de esfuerzo de BPMI Notation Working Group. (15) . Actualmente es mantenida por el Object Management Group (OMG) desde que ambas organizaciones se unieron en el 2005. La versión más actual de BPMN es la 1.1 y un proceso de mayor revisión para BPMN 2.0 está en progreso. (16)

El objetivo principal de BPMN es proveer una notación que sea fácilmente comprensible por todos los usuarios del negocio, por los analistas que crean los primeros modelos, por los desarrolladores técnicos, responsables de implementar la tecnología que realizara dichos procesos, y finalmente, a las personas del negocio, los cuales gestionan y monitorean los procesos. Esta notación, crea un puente para la normalización de las diferencias entre el diseño de los procesos de negocio y el proceso de implementación.

BPMN define un Diagrama de Procesos de Negocio (BPD, Business Process Diagram), basado en una técnica de diagramas de flujo adaptados para la creación gráfica de modelos.

Un BPD esta hecho de un conjunto de elementos gráficos, organizados en categorías básicas:

- Objetos de flujo.
- Objetos de conexión
- Calles
- Artefactos

Cada una de estas categorías posee un conjunto de elementos gráficos. Los objetos de flujo pueden ser evento, actividad o decisión. Los objetos de conexión pueden ser, flujo de secuencia, flujo de mensaje y asociación. Las calles pueden ser simple (pool) o poseer varias pistas (lanes), con uno o varios participantes respectivamente y finalmente, los artefactos pueden ser datos, grupo o anotación. Todo lo descrito anteriormente se muestra en la siguiente figura.

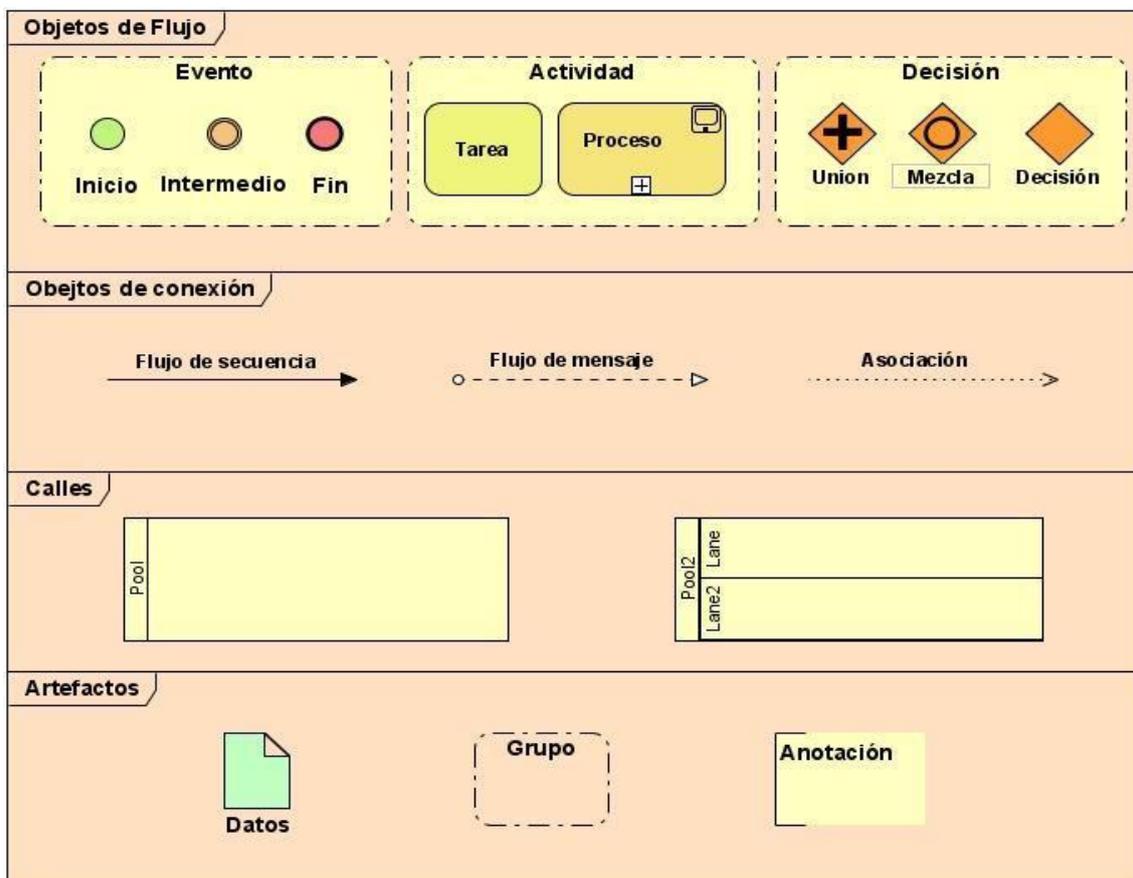


Figura 3 Elementos gráficos y categorías de un BPD

IDEF0

IDEF0 es un método diseñado para modelar decisiones, acciones y actividades de una organización o sistema. Los efectivos modelos de IDEF0 permiten organizar el análisis de un sistema y facilita la comunicación entre el analista y el cliente. Como herramienta de análisis, IDEF0 asiste al analista para identificar que funciones se llevan a cabo, lo que se necesita para llevarlas a cabo, lo que se hace bien y mal en los procesos que se estén modelando.

(17)

IDEF0 tiene las características siguientes:

- Es comprensivo y expresivo, capaz, gráficamente, de representar una variedad amplia de negocio, de fabricación y de otros tipos de operaciones de la empresa a cualquier nivel del detalle.
- Es un lenguaje coherente y simple, que preveé la expresión rigurosa y exacta, y promueve la consistencia del uso y de la interpretación.

Puede ser generado por una gran variedad de herramientas gráficas en computadores. Muchos productos comerciales apoyan específicamente el desarrollo y el análisis de los diagramas IDEF0 y de sus modelos.

Lenguaje de Modelado de Sistemas

El Lenguaje de Modelado de Sistemas (SysML, System Modeling Language) es un nuevo lenguaje de modelado de propósito general para la ingeniería de sistemas. Soporta la especificación, análisis, diseño, verificación y validación de un amplio rango de complejos sistemas; que pueden incluir hardware, software, información, procesos, personal y facilidades. SysML puede ser adaptado para modelar aplicaciones de un dominio específico, como automovilísticas, aeroespaciales, comunicaciones y sistemas de información.

Es efectivo particularmente al especificar requerimientos, estructura, comportamiento y restricciones de las propiedades del sistema. El lenguaje está orientado a soportar múltiples procesos y enfoques como el estructurado,

orientado a objetos y otros, pero cada metodología puede imponer restricciones adicionales en como un tipo de artefacto o diagrama debe ser usado.

SysML reutiliza un subconjunto de UML 2.0, manteniendo intactos algunos diagramas y redefiniendo otros, a su vez que incorpora nuevos diagramas. Entre los más importantes se encuentra el diagrama de requerimientos, en el cual se pueden representar los requerimientos en formato gráfico, tabular o estructura de árbol.

Un requerimiento estándar incluye propiedades para especificar su identificador único y el texto del requerimiento. Por otra parte también se representan las relaciones entre requerimientos, que incluyen herencia, derivación, refinación y otras. (18)

Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (UML, Unified Modeling Language) se define como un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software. (19)

El UML es un estándar incipiente de la industria para construir modelos orientados a objetos. Es un sistema de notación elegante, expresivo y flexible, que entre otras cosas incluye el significado de sus notaciones. UML no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos, ni le indica cual proceso de desarrollo adoptar. (20)

UML permite la implementación de diversos tipos de diagramas, dentro de los cuales se encuentra el diagrama de casos de uso, que modela la funcionalidad del sistema según lo perciben los usuarios externos, llamados actores. Un caso de uso es una unidad coherente de funcionalidad, expresada como transacción entre los actores y el sistema. (19)

1.6 Herramientas para el modelado de procesos

Las herramientas para el modelado de procesos de negocio proveen facilidades a los usuarios para representar las complejas semánticas de dichos procesos. Permiten además obtener una vista estática del proceso en su conjunto.

Business Process Visual Architech 2.0

Business Process Visual ARCHITECT (BP-VA) es una herramienta de modelado visual que provee el soporte más robusto para Business Process Modeling Notation (BPMN) y constituye parte de la suite de Visual Paradigm. Posee un conjunto de características que hacen el modelado de los procesos del negocio un trabajo fácil, rápido y mejor; a continuación se detallan algunas de ellas (21):

- Creación de formas (elementos del diagrama) con recursos céntricos y gestos del mouse.
- Elaboración de diagramas de procesos para subprocessos.
- Formación de diagramas de subprocessos a partir de formas seleccionadas.
- Detección inteligente de conexiones inválidas entre formas.
- Reordenación fácil de trabajadores de las calles.
- Elección de diferentes estilos de conexión
- Simple clic inteligente en los recursos de conexión (22)

En BP-VA los recursos y las facilidades del modelado están bien organizados en ventanas plegables.

Muchas operaciones en el modelado pueden ser llevadas a cabo rápidamente mediante la edición en el lugar, menús contextuales o por medio de la tabla de propiedades.

Para hacer el modelado más eficiente, BP-VA adopta la interfaz de recursos céntricos, en donde botones de contexto sensitivo (referidos como recursos)

pueden ser mostrados rodeando el elemento activo del diagrama. Cada recurso brinda una funcionalidad que se realiza frecuentemente, como crear una conexión a una nueva o existente forma, abrir la especificación del modelo, entre otras. De esta forma, se ahorra tiempo considerable y movimientos del mouse al acceder al menú o la barra de herramientas para obtener la funcionalidad deseada.

BP-VA ofrece validación inteligente de la notación mientras se editan los diagramas, para asegurar que siempre se esté en el camino correcto.

Con BP-VA es posible además, exportar los diagramas a formato de imagen JPG o archivo XML (21).

BPWin 4.1

BPWin es una potente herramienta de modelamiento utilizada para analizar, documentar y mejorar los procesos de negocio de una organización (actividades). Permite documentar de manera clara las actividades necesarias, cómo se realizan y qué recursos consumen, lo cual proporciona una visión exacta, no solo de qué es lo que hace nuestra organización, sino si lo hace de forma eficiente.

Ofrece un conjunto de bondades que lo hacen más utilizable en el mundo de los procesos y la ingeniería. Algunos de sus beneficios son los siguientes:

- Interfaz Intuitiva
- Diseño Automatizado de Procesos
- Propiedades Definidas por el Usuario
- Técnicas de Integración a través del IDEF0, IDEF3 y DFD.
- Métrica y Análisis de Costos

Permite modelar los diagramas de contexto, crear sus actividades y establecer sus relaciones mediante flechas. Cada uno de dichos elementos a su vez, pueden ser documentados mediante textos. El diagrama de contexto puede ser descompuesto fácilmente mediante pocas acciones. (23)

Microsoft Office Visio 2007

Office Visio 2007 facilita a los profesionales empresariales la visualización, análisis y comunicación de información, sistemas y procesos complejos. Mediante los diagramas de aspecto profesional de Office Visio 2007, se puede mejorar la comprensión de sistemas y procesos, entender mejor la información compleja y utilizar esos conocimientos para tomar mejores decisiones para la empresa.

Office Visio 2007 permite documentar, diseñar y comprender de forma visual el estado de los sistemas y procesos empresariales con una gran variedad de diagramas, diagramas de flujo de proceso empresarial, diagramas de red, diagramas de flujo de trabajo, modelos de bases de datos y diagramas de software, entre otros (24).

Facilita y agiliza la creación de diagramas, tan solo con abrir una plantilla, arrastrar formas hasta los dibujos y aplicar temas para lograr un acabado perfecto. Los diagramas están organizados en plantillas según su tipo.

La característica Autoconexión se ocupa de conectar las formas de manera automática, las distribuye de modo uniforme y las alinea con precisión, todo ello con un par de clics de mouse.

Permite además exportar a los formatos PDF y XML Paper Specification (XPS) (25).

1.7 Ingeniería de Requerimientos

Para lograr un mejor entendimiento de lo que es la ingeniería de requerimientos (IR) y lo que representa, es necesario primero comprender que son los requerimientos, cómo están definidos y para qué sirven. Son varias las definiciones que existen actualmente, entre las que se pueden citar:

- es un atributo necesario en un sistema, una declaración que identifica la capacidad, características, o factor de calidad de un sistema a fin de que tenga valor y utilidad a un cliente o usuario. (26)
- una condición o capacidad necesaria dada por un usuario con el objetivo de resolver un problema o alcanzar un objetivo (27)
- expresan las necesidades y restricciones atribuibles a un producto de software que contribuye a la solución de algún problema del mundo real (28)

Los requerimientos pueden ser clasificados como funcionales o no funcionales. Los requerimientos funcionales describen qué es lo que el sistema debe hacer para dar soporte a las funciones y objetivos del usuario. Deben responder a las siguientes preguntas (28):

- ¿Cómo las entradas son transformadas en salidas?
- ¿Quién inicia y recibe información específica?
- ¿Qué información debe estar disponible para que cada función sea ejecutada?

Los requerimientos no funcionales imponen restricciones de cómo los requerimientos funcionales deben ser implementados (28).

Una vez definidos los requerimientos se puede comprender mejor qué es y qué abarca la IR. Para dar respuesta a las interrogantes anteriormente formuladas hay que remitirse a las distintas definiciones dadas por los autores e instituciones más citados en la IR:

- La Ingeniería de Requerimientos del software es un proceso de descubrimiento, refinamiento, modelado y especificación. Se refina en detalle los requerimientos del sistema y el papel asignado al software. Se crean modelos de los requerimientos de datos, flujos de información y control, y del comportamiento operativo. Se analizan alternativas y el modelo completo del análisis es creado. (29)

- La Ingeniería de Requerimientos se ocupa de la elicitación, análisis, especificación y validación de requerimientos de software (30)
- Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo. (27)
- Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal. (27)

1.7.1 Etapas fundamentales

La IR está formada por una serie de etapas bien diferenciadas, las cuales son nombradas y descritas a continuación:

- Elicitación
- Análisis
- Especificación
- Verificación y validación
- Administración

Elicitación de requerimientos

La captura de requerimientos es la actividad mediante la cual el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema (31). Es fundamentalmente una actividad humana, y es en donde los *stakeholders* se identifican y comienzan a establecerse las relaciones entre el equipo desarrollador y el cliente. A este proceso también se lo conoce como “Captura de requerimientos”, “Descubrimiento de requerimientos” y “Adquisición de requerimientos”.

En esta etapa se acude a varias fuentes de información. Algunos autores consideran que la principal fuente de información son los usuarios de una organización. Otros utilizan en cambio a los expertos del dominio, de los cuales obtiene información de forma oral. Existen también los que elicitan a las personas, pero no solamente a los usuarios o clientes sino también a cualquier

otro beneficiario. Algunos siguen la línea de elicitar desde los expertos del dominio. En sentido general, la fuente mencionada por casi todos los actores son las personas. Sin embargo las personas no son las fuentes más adecuadas. Todo lo anteriormente expuesto queda perfectamente evidenciado en (32).

Análisis de requerimientos

Esta es la etapa de derivación de requerimientos del sistema de software a través de la observación de sistemas existentes, discusión con usuarios y proveedores de información potenciales, análisis de tareas y así sucesivamente. Esto puede involucrar el desarrollo de uno o más modelos diferentes del sistema, que ayudan al analista a entender el sistema que será especificado. También pueden desarrollarse prototipos para ayudar a entender requerimientos. (33)

En la etapa de análisis de requerimientos se trata, precisamente, de analizar la información recibida desde los usuarios, para distinguir las necesidades de tareas, los requerimientos funcionales, atributos de calidad, soluciones sugeridas, de información extraña. (34) En esta etapa además, se clasifican los requerimientos identificados y se eliminan todas las inconsistencias y ambigüedades que los mismos posean.

Se define a la etapa de análisis de requerimientos como la actividad de transformar requerimientos informales en requerimientos técnicos mediante el aseguramiento de que los mismos reflejan los atributos de calidad de los requerimientos y que expresan las necesidades de los clientes. El análisis es una actividad iterativa. Los pasos del proceso deberán ser repetidos una cierta cantidad de veces, existe una constante comunicación entre la consulta de los clientes, los usuarios finales y los desarrolladores. (26)

Especificación de requerimientos.

En esta actividad se detalla el documento de Especificación de Requerimientos de Software (ERS). Conjuntamente se puede realizar un diseño de alto nivel

para ayudarse a descubrir errores en la definición de los requerimientos, los cuales deben ser corregidos.

Las especificaciones forman un puente entre requerimientos y diseño. Sus entradas son el documento de requerimientos y la información de los usuarios y desarrolladores. La especificación de un sistema de software describe aspectos funcionales y de calidad. (33)

Los clientes y usuarios utilizan la Especificación de Requerimientos para comparar si lo que se está proponiendo, coincide con las necesidades de la empresa. Los analistas y programadores la utilizan para determinar el producto que debe desarrollarse. El personal de pruebas elaborará las pruebas funcionales y de sistemas en base a este documento. Para el administrador del proyecto sirve como referencia y control de la evolución del sistema. (34)

Verificación y validación de requerimientos

Muchos de los autores utilizan a veces los términos verificación y validación como sinónimos y en la mayoría de la bibliografía sobre Ingeniería de Requerimientos pueden aparecer unidos. Generalmente la verificación y validación suele asociarse a los distintos tipos de técnicas de prueba, principalmente sobre el producto final.

Existen varios estándares como el 1012 de la IEEE de 1997 (Standard for Software Verification and Validation Plans), el 1059 (Guide for Software Verification and Validation Plans) que considera la verificación y validación como dos actividades prácticamente inseparables. En el glosario de la IEEE de 1990 aparecen ambos términos en una única entrada:

Verificación y validación (1): *el proceso de determinar si los requerimientos para un sistema o componente son completos y correctos, los productos de cada fase de desarrollo satisfacen los requerimientos o condiciones impuestas por la fase previa y el sistema o componente final es acorde con los requerimientos especificados.*

Curiosamente más adelante en el mismo glosario aparecen también los términos pero como entradas separadas y no coincide completamente con la definición anterior. En este caso obvian por completo los aspectos relativos a los requerimientos:

Verificación (2): *(a) el proceso de evaluar un sistema o componente para determinar si los productos de una fase de desarrollo dada satisfacen las condiciones impuestas al comienzo de la fase. (b) prueba formal de la corrección de un programa.*

Validación (2): *el proceso de evaluar un sistema o componente durante o al final del proceso de desarrollo para determinar si satisface los requerimientos especificados.*

Otras definiciones menos formales se pueden encontrar de la siguiente forma:

Verificación (3): *¿se está construyendo correctamente el producto? (am I building the product right?)*

Validación (3): *¿se está construyendo el producto correcto? (am I building the right product?)*

En este trabajo se tratan los términos atribuyéndoles significados distintos, especialmente en la Ingeniería de Requerimientos que no tienen que ir necesariamente unidos.

Por verificación se entenderá el conjunto de actividades relacionadas con la calidad de las especificaciones de requerimientos respecto a un conjunto de propiedades deseables.

Se entenderá por validación de requerimientos el conjunto de actividades encaminadas a llegar a un acuerdo entre todos los participantes, en el que se ratifique que los requerimientos elicitados y analizados representan realmente las necesidades de clientes y usuarios.

Administración de requerimientos

La administración de requerimientos es un conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requerimientos y los cambios en cualquier momento. (29)

La administración de los requerimientos incluye todas las actividades que mantienen la integridad y exactitud de los requerimientos a medida que el proyecto progresa. En esta etapa se enfatizan (34):

- Control de los cambios de los requerimientos que están sobre la línea base definida.
- Control de versiones tanto de requerimientos individuales como del documento de requerimientos.

Dificultades fundamentales evidenciadas en la etapa de Elicitación de requerimientos

La elicitación de requerimientos es el primer y más crítico paso dentro de la Ingeniería de Requerimientos. Realizado de manera errónea llevará a la creación de productos con calidad pobre y fechas de entregas tardías.

Los desarrolladores de software tradicionalmente han creado productos para otros usuarios y no para sí mismos. La experiencia ha demostrado que las personas son una fuente imperfecta de información. La mayoría de los sistemas poseen una gran cantidad de usuarios y roles y cada uno solo conoce lo que hacen individualmente pero en raras ocasiones tienen una visión global. Los interesados no saben cómo puede hacerse más eficiente la operación en su conjunto. Además no tienen noción que parte de su trabajo puede transformarse en software.

Son innumerables las situaciones y dificultades que se evidencian ya sean limitaciones en el conocimiento, problemas de comportamiento humano, problemas técnicos, entre otros tantos que se puedan poner de manifiesto. (26)

1.7.2 Técnicas utilizadas en la Ingeniería de Requerimientos

Con el objetivo de minimizar las dificultades que se presentan en la IR, se aplican un conjunto de técnicas para la obtención de la información. A continuación se detallarán algunas de estas técnicas.

Entrevistas y cuestionarios

Durante la entrevista, el analista conversa con el encuestado; el cuestionario consiste en una serie de preguntas relacionadas con varios aspectos de un sistema. Por lo común, los encuestados son usuarios de los sistemas existentes o usuarios en potencia del sistema propuesto. En algunos casos, son gerentes o empleados que proporcionan datos para el sistema propuesto o que serán afectados por él.

Con frecuencia, se utilizan preguntas abiertas para descubrir sentimientos, opiniones y experiencias generales, o para explorar un proceso o problema.
(28)

Las preguntas pueden ser enfocadas a un elemento del sistema, tales como usuarios, procesos, etc. El siguiente ejemplo muestra algunos tipos de preguntas abiertas.

- ¿Quién es el cliente?
- ¿Quién es el usuario?
- ¿Son sus necesidades diferentes?
- ¿Cuáles son sus habilidades, capacidades, ambiente?
- ¿Cuál es la razón por la que se quiere resolver este problema?
- ¿Cuál es el valor de una solución exitosa?
- ¿Cómo usted resuelve el problema actualmente?
- ¿Qué retrasos ocurren o pueden ocurrir?
- ¿Qué problemas podría causar este producto en el negocio?
- ¿En qué ambiente se usará el producto?
- ¿Cuáles son sus expectativas para los conceptos fácil de usar, confiable, rendimiento?

Tormenta de ideas

La tormenta de ideas es una técnica simple para la generación de ideas. Se reúne un conjunto de personas relacionadas al producto y éstas sugieren y exploran sus ideas libremente. Generalmente las reuniones se realizan con 4 a 10 personas; una de ellas deberá actuar como líder para comenzar pero no para restringir la expresión de los restantes participantes. Se genera una amplia variedad de puntos de vistas, se estimula el pensamiento creativo, se logra construir una imagen más completa del problema, se provee un ambiente social mucho más confortable y de fácil aprendizaje. (33)

Sistemas existentes

Esta técnica consiste en analizar distintos sistemas ya desarrollados que estén relacionados con el sistema a ser construido. Por un lado, es posible analizar las interfaces de usuario, observando el tipo de información que se maneja y cómo es manejada, por otro lado también es útil analizar las distintas salidas que los sistemas producen (listados, consultas, etc.), porque siempre pueden surgir nuevas ideas sobre la base de estas. (26)

Prototipos

Durante la actividad de extracción de requerimientos, puede ocurrir que algunos requerimientos no estén demasiado claros o que no se esté muy seguro de haber entendido correctamente los requerimientos obtenidos hasta el momento, todo lo cual puede llevar a un desarrollo no eficaz del sistema final.

Para validar los requerimientos hallados, se construyen prototipos. Los prototipos son simulaciones del posible producto, que luego son utilizados por el usuario final, permitiéndonos conseguir una importante retroalimentación en cuanto a si el sistema diseñado con base a los requerimientos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva.

Un prototipo de requerimientos de software es una implementación parcial del sistema, se construye para ayudar a los desarrolladores, usuarios y clientes en

la obtención de un mejor entendimiento del sistema, en especial de los requerimientos que están menos claros. Existe un fuerte solapamiento de su uso para la elicitación de requerimientos respecto a la validación de requerimientos. El prototipado es un medio común para la validación de la interpretación de los ingenieros de software de los requerimientos de software, así como también para la elicitación de los mismos. (35)

Casos de uso

Los casos de uso son una técnica para especificar el comportamiento de un sistema. Los casos de uso permiten entonces describir la posible secuencia de interacciones entre el sistema y uno o más actores, en respuesta a un estímulo inicial proveniente de un actor, es una descripción de un conjunto de escenarios, cada uno de ellos comenzado con un evento inicial desde un actor hacia el sistema. La mayoría de los requerimientos funcionales, sino todos, se pueden expresar con casos de uso. Según el autor Sommerville, los casos de uso son una técnica que se basa en escenarios para la obtención de requerimientos (29) .

Desarrollo Conjunto de Aplicaciones

La técnica denominada *JAD (Joint Application Development, Desarrollo Conjunto de Aplicaciones)*, desarrollada por IBM en 1977, es una alternativa a las entrevistas individuales que se desarrolla a lo largo de un conjunto de reuniones en grupo durante un periodo de 2 a 4 días. En estas reuniones se ayuda a los clientes y usuarios a formular problemas y explorar posibles soluciones, involucrándolos y haciéndolos sentirse partícipes del desarrollo.

Esta técnica se base en cuatro principios (35): dinámica de grupo, el uso de ayudas visuales para mejorar la comunicación (diagramas, transparencias, multimedia, herramientas CASE, etc.), mantener un proceso organizado y racional y una filosofía de documentación *WYSIWYG (What You See Is What You Get, lo que se ve es lo que se obtiene)*, por lo que durante las reuniones se trabaja directamente sobre los documentos a generar.

Debido a las necesidades de organización que requiere y a que no suele adaptarse bien a los horarios de trabajo de los clientes y usuarios, esta técnica no suele emplearse con frecuencia, aunque cuando se aplica suele tener buenos resultados, especialmente para elicitar requerimientos en el campo de los sistemas de información (36).

1.8 Métricas de software

Para una mayor comprensión de la utilidad de las métricas de software en la IR, es necesario primeramente dar respuesta a algunas interrogantes como ¿qué son las métricas?, ¿para qué sirven?, ¿cómo se puede evidenciar su utilidad en la IR? En esta sección se dará respuesta a las interrogantes formuladas anteriormente.

Según aparece definido en el Glosario Estándar de Términos de la Ingeniería de Software de la IEEE, la métrica es una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado.

La medida es un factor clave en el desarrollo del software ya que: ayuda a entender qué está pasando durante el desarrollo y el mantenimiento, permite controlar el desarrollo del proyecto, y estimula la mejora de procesos y productos. (35)

Hay varias razones que justifican el uso de las métricas en el proceso de desarrollo de software. Por un lado se dice que cuando se puede medir aquello de lo cual se está hablando y se puede expresar en números, se sabe realmente acerca de ello; pero cuando no puede medirse, y no puede expresarse en números, el conocimiento que se tiene de ello es escaso e insatisfactorio. Además, una aplicación correcta de las métricas de software permite realizar mejores estimaciones de tiempo y costo, cuantificar la productividad de los desarrolladores y del uso de las herramientas de desarrollo.

Pocos antecedentes de métricas existen que traten directamente la cuantificación de atributos del proceso de elicitación de requerimientos. La herramienta automática conocida como “Automated Requirement Measurement (ARM)”, desarrollada por el Centro Tecnológico de Aseguramiento de Software (Software Assurance Technology Center (SATC)) de la NASA Goddard Space Flight Center para la administración de requerimientos, ha sido utilizada como apoyo en la especificación de requerimientos.

El objetivo de la ARM es proveer métricas que puedan ser utilizadas por el líder de proyecto para evaluar la calidad del documento de requerimientos. La herramienta no tiene la intención de evaluar la exactitud de los requerimientos especificados, sino que es una ayuda para “escribir los requerimientos correctamente” y no para “escribir los requerimientos correctos”. (37)

1.9 Métodos para la medición de la satisfacción del cliente

En esta sección se describen algunos de los métodos más utilizados para medir la satisfacción del cliente.

SERVQUAL

SERVQUAL (Service Quality) es un método para la medición de la satisfacción del cliente, permitiendo hacer de las mediciones, indicadores lo más cercanos a la realidad. Fue desarrollado por Zeithaml, Parasuraman y Berry en 1988 en sus estudios realizados para su trabajo “Delivering Quality Service”.

Define un conjunto de cinco dimensiones básicas que caracterizan a un servicio (38):

1. Elementos tangibles: representan las características físicas y apariencia del proveedor.
2. Fiabilidad: implica la habilidad que tiene la organización para ejecutar el servicio prometido de forma adecuada y constante.
3. Capacidad de respuesta: representa la disposición de ayudar a los clientes y proveerlos de un servicio rápido.

4. Seguridad (Garantía): son los conocimientos y atención mostrados por los empleados respecto al servicio que están brindando, además de la habilidad de los mismos para inspirar confianza y credibilidad.
5. Empatía: es el grado de atención personalizada que ofrecen las empresas a sus clientes.

Como una de las actividades que se propone se encuentra la aplicación de un cuestionario, en el cual se evalúan una serie de indicadores que tributan a las dimensiones antes mencionadas. Los indicadores son medidos en dos momentos diferentes:

- El primero, dedicado a recoger afirmaciones que tratan de reflejar las expectativas generales de los clientes.
- El segundo, dedicado a recoger afirmaciones que reflejen las percepciones de la empresa que presta el servicio.

El método además propone cinco brechas que determinan el grado de ineficiencia en la gestión de los servicios de satisfacción del cliente. La brecha cinco por ejemplo, representa la diferencia entre las expectativas que se generan de los clientes antes de recibir el servicio y la percepción que obtienen del mismo una vez recibido, determinando el nivel de satisfacción de los clientes.

Si el valor de la brecha es positivo se puede concluir que las expectativas de los clientes fueron superadas, mientras más positiva sea la brecha más sorpresa se generó en los mismos, ya que la experiencia que vivieron fue superada muy por encima de lo que esperaban antes de vivirla. De ocurrir lo contrario, en que la brecha tome un valor negativo, entonces el cliente esperaba más acerca de su experiencia que lo que en realidad percibió.

Modelo ACSI de Satisfacción del cliente

The American Customer Satisfaction Index (ACSI) es un indicador que establece el nivel de satisfacción de los ciudadanos de los EEUU con los

productos y servicios recibidos desde 1994. La representación gráfica del modelo se puede apreciar a continuación:



Figura 4 Diagrama del modelo ACS

El valor del indicador se obtiene del tratamiento de las respuestas de los estadounidenses a un cuestionario telefónico, y se presentan los resultados en 4 niveles (39):

- Valor del indicador a nivel nacional.
- Valor del indicador en 10 sectores económicos.
- Valor del indicador en 43 industrias diferentes.
- Valor del indicador en más de 200 empresas y agencias del gobierno.

Expectativas del cliente: las expectativas del cliente son una medida anticipada de la calidad que el cliente espera recibir por los productos y servicios que la organización ofrece.

Calidad percibida: tomando como entrada las expectativas del cliente, la Calidad percibida se considera asociada principalmente a 2 factores: la personalización y la fiabilidad.

Valor percibido: este parámetro expresa la relación entre la calidad obtenida y el precio pagado.

Quejas del cliente: las quejas son la expresión más palpable de la insatisfacción. Cuanto más satisfecho está un cliente, menos ganas tiene de expresar una queja.

Fidelidad del cliente: la fidelidad del cliente es el componente crítico del modelo. Se observa que, si bien la satisfacción del cliente ocupa un lugar central en el diagrama, las flechas relacionales desembocan en este parámetro.

El método de Kano

El académico japonés de la universidad de Tokio Noriaki Kano definió un modelo de evaluación de la calidad de los productos que permite medir la relación entre la funcionalidad de un producto y la satisfacción que esta funcionalidad les brinda a los clientes. Además permite discriminar y clasificar los requerimientos en tres tipos ideales de atributos en función de la relación entre funcionalidad y satisfacción. Ver la figura que aparece a continuación, tomada de (40).

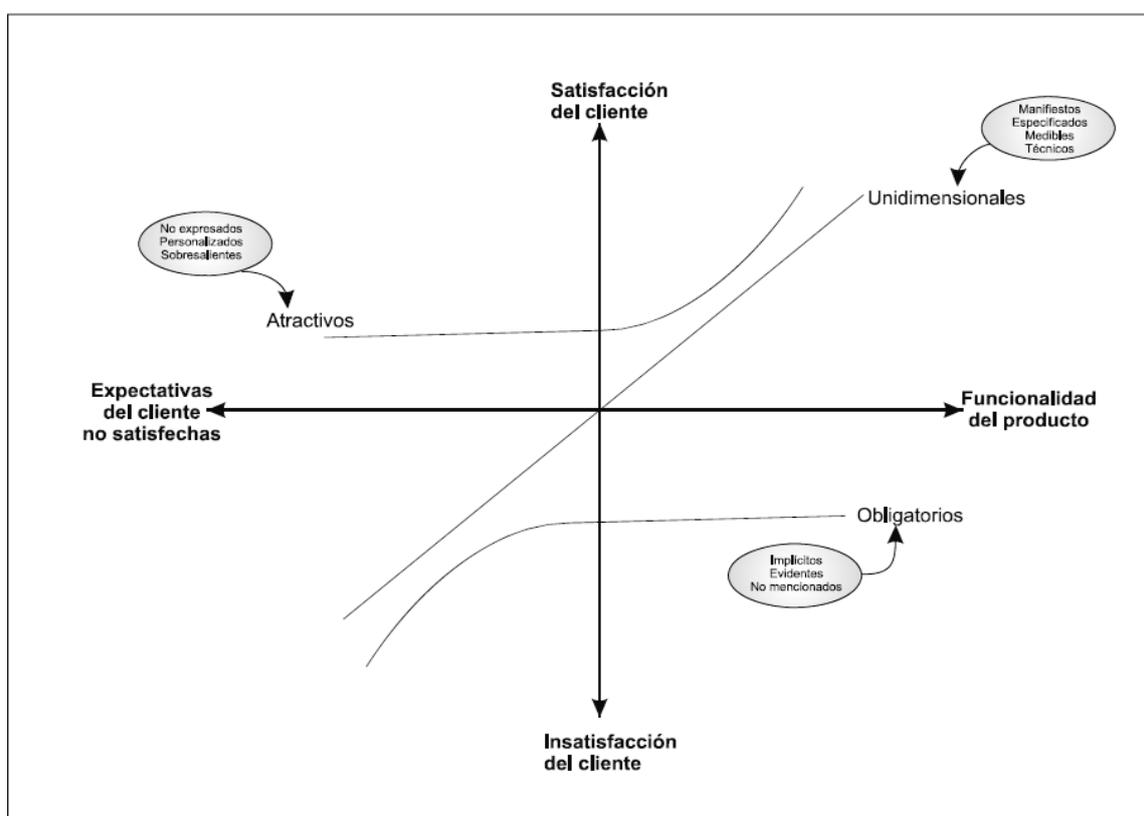


Figura 5 Clasificación de los requerimientos con el método de Kano.

Atractivos: son aquéllos que, por debajo de cierto umbral de funcionalidad, mantienen un nivel de satisfacción relativamente bajo y constante, pero que, una vez superado ese umbral, producen un aumento significativo de la satisfacción. Los requerimientos atractivos suelen denominarse deleitosos (del inglés *delighter*).

Unidimensionales: se caracterizan porque la satisfacción que producen aumenta de modo aproximadamente proporcional al nivel de funcionalidad. Responden a la percepción tradicional de la relación entre funcionalidad y satisfacción: a mayor funcionalidad, se observa una mayor satisfacción.

Obligatorios: son aquéllos que, hacia las gamas bajas de funcionalidad, aumentan la satisfacción en relación directa con la funcionalidad pero que, superado cierto umbral, dejan de producir un incremento importante en la satisfacción.

Para obtener la clasificación antes mencionada Kano propone un cuestionario, en el que se formulan dos preguntas por cada requerimiento analizado:

- ¿Cómo se siente si la característica X está presente en el producto?
(Requerimientos funcionales)
- ¿Cómo se siente si la característica X NO está presente en el producto?
(Requerimientos disfuncionales)

Para cada pregunta el cliente responde entre cinco posibles opciones:

1. Me agrada
2. Es de esperarse
3. Neutral
4. Lo acepto
5. Me desagrada

Una vez respondidas las dos preguntas se busca su combinación en la Tabla 1, y es de esta forma que los requerimientos de un producto se pueden clasificar en una de las seis categorías que se muestran a continuación:

A: Atractivo

B: Obligatorio

U: Unidimensional

I: Indiferencia

Inv: Respuesta inversa

D: Respuesta dudosa

	Requerimientos disfuncionales					
Requerimientos funcionales		1	2	3	4	5
	1	D	A	A	A	U
	2	Inv	I	I	I	O
	3	Inv	I	I	I	O
	4	Inv	I	I	I	O
	5	Inv	Inv	Inv	Inv	D

Tabla 1 Clasificación de los requerimientos

Cuando un cliente se comporta Indiferente indica que una mayor o menor funcionalidad respecto a este requerimiento no se refleja en un aumento o disminución de la satisfacción del cliente y se representaría como una recta paralela al eje horizontal de la Figura 1. Una respuesta Inversa indica que la interpretación de criterios funcionales y disfuncionales del diseñador es la inversa a la percepción del cliente, es decir, lo que la pregunta supone como funcional es percibido como no funcional por quien responde. Por último, cuando existe una contradicción en las repuestas a las preguntas, se clasifica como una respuesta Dudosa. Esto quiere decir que no es razonable contestar lo mismo ante un par de preguntas complementarias.

1.10 Patrones de Casos de Uso

Un patrón de caso de uso generalmente es un diseño probado en un modelo de casos de uso, junto a una descripción del contexto en el cual debería ser aplicado y las consecuencias que se derivan de su aplicación en el modelado.

Los patrones de casos de uso son usados como plantillas para estructurar el modelado.

Concordancia: Reuso (Commonality: Reuse)

El patrón Concordancia: Re-uso es un patrón de estructura que consiste en tres casos de uso. El primero llamado “Sub-secuencia Común”, modela la secuencia de acciones que aparecen en múltiples casos de uso del modelo. Los otros dos casos de uso comparten de esta sub-secuencia común de acciones (dos es la menor cantidad que puede existir).

La sub-secuencia tiene que estar en un fragmento, es decir, todo lo que requiere estar incluido tiene que estar en un único fragmento completo. Además no se puede hacer referencia desde la sub-secuencia a donde esta es utilizada, porque el caso de uso incluido tiene que ser independiente del caso de uso base

Concordancia: Especialización (Commonality: Specialization)

Otro patrón de concordancia que contiene casos de uso del mismo tipo. En este caso, ellos son modelados como especializaciones de un caso de uso de uso común. Todas las acciones del caso de uso común son heredadas por sus casos de uso hijos, en donde otras acciones pueden ser agregadas o las acciones heredadas pueden ser especializadas.

Este patrón es aplicable cuando las acciones modeladas por los casos de uso son del mismo tipo, y este tipo debería ser hecho visible en el modelo.

Extensión o Inclusión Concreta: Extensión (Concrete Extension or Inclusion: Extension)

El patrón Extensión o Inclusión Concreta: Extensión consiste en dos casos de uso y una relación de extensión entre ellos. El caso de uso extendido es concreto; es decir, puede ser instanciado por el mismo, así como también puede extender del caso de uso base.

El patrón es aplicable cuando un flujo puede extender el flujo de otro caso de uso, así como también puede ser realizado por el mismo.

Extensión o Inclusión Concreta: Inclusión (Concrete Extension or Inclusion: Inclusion)

En este patrón, existe una relación de inclusión desde el caso de uso base al caso de uso incluido. El último puede ser instanciado por el mismo. El caso de uso base podría ser cualquiera, concreto o abstracto.

El patrón es usado cuando un flujo podría ser incluido en el flujo de otro caso de uso y además puede ser realizado por el mismo.

CRUD: Completo

El patrón CRUD: Completo consiste en un caso de uso, llamado Gestionar Información, modelando todas las diferentes operaciones que pueden ser realizadas en una pieza de información de cierto tipo, como crearla, leerla, actualizarla y eliminarla.

Este patrón debería ser usado cuando todos los flujos contribuyen al mismo valor de negocio y son cortos y sencillos.

Actores Múltiples: Roles distintos (Multiple Actors: Distinct Roles)

Consiste en un caso de uso y al menos dos actores. Es usado cuando dichos actores juegan diferentes roles en relación con el caso de uso, es decir, ellos interactúan diferentemente con el caso de uso.

Actores Múltiples: Roles comunes (Multiple Actors: Common Roles)

Es un patrón alternativo, los dos actores juegan el mismo rol en relación con el caso de uso. Este rol es representado por otro actor, del cual heredan los dos actores anteriores que comparten el mismo rol.

El patrón es aplicable cuando, desde el punto de vista del caso de uso, solo existe una entidad externa interactuando con cada instancia del caso de uso.

1.11 Conclusiones

Se puede concluir que:

- Una vez analizados los principales lenguajes de modelado, se decidió utilizar BPMN para el modelamiento de los procesos del negocio por su facilidad de comprensión para todos los participantes en el negocio y en el desarrollo del software.
- Para modelado del sistema, resaltó por sus características el estándar UML para la realización de los casos de uso y SysML para la realización de los diagramas de requerimientos.
- La herramienta seleccionada para el modelado de procesos fue Business Process Visual Architech 2.0, dado que soporta el lenguaje de modelado de procesos que será utilizado, entre otras características mencionadas en el capítulo.
- La metodología a utilizar será RUP por tener características perfectamente ajustables al proyecto y el conocimiento por parte del equipo de desarrollo es mucho mayor que con otras descritas.
- Para el modelado de sistema en general se optó por la herramienta Visual Paradigm Suite 3.0, dado que contiene la herramienta de modelado de procesos antes mencionada y la herramienta para el modelado de sistema Visual Paradigm for UML 6.0. Además, permite la modelación con la especificación SysML para la administración de los requerimientos. Los prototipos de interfaz de usuario serán modelados en Office Visio 2007, pues permite crear modelos gráficos de aspecto profesional.
- El método de satisfacción del cliente seleccionado fue el de Kano por ser este mucho más específico a la hora de determinar no solo el grado de satisfacción del cliente, sino también permite medir la calidad de las características de los productos ofrecidos.

CAPÍTULO 2 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

2.1 Introducción

En el presente capítulo se realiza la propuesta de solución del problema planteado. Teniendo como guía la metodología de desarrollo de software RUP, con el fin de lograr un entendimiento del negocio se realiza un modelado de procesos utilizando BPMN, que constituye el estándar finalmente adoptado por la OMG para este fin.

Se aplican las etapas de la Ingeniería de Requerimientos para obtener los requerimientos del software a construir y lograr que los mismos sean consistentes, completos, sin ambigüedad y con la calidad requerida, además de posibilitar la gestión de cambio de los mismos.

Se presenta además el diagrama de casos de uso del sistema (DCUS) y las descripciones de tres de ellos.

2.2 Modelado de procesos

El modelado de procesos de negocio es la base para comprender mejor la operación de una organización, documentar y publicar los procesos buscando una estandarización en la organización, buscar eficiencias en la operación e integrar soluciones en arquitecturas orientadas a servicios.

Los procesos de negocio son la base para comprender mejor la forma en que opera un negocio en sus diferentes áreas y son una herramienta fundamental para acceder a modelos de calidad y eficiencia.

Proceso de Registro y Aprobación de Proyectos

El proceso de Registro y Aprobación de Proyectos comienza a partir de un análisis de las necesidades y oportunidades de desarrollar un proyecto, por parte de una persona desempeñando un rol de planificador. La entidad a la que

pertenece el planificador mencionado anteriormente, es la responsable de llevar a cabo todos los proyectos que de dicha entidad formulen.

Una vez analizadas las necesidades y oportunidades, se comienza a identificar lo que podría posteriormente convertirse en una propuesta. En la mayoría de los casos, en este proceso de identificación de la propuesta, el planificador recurre a un banco de proyectos y propuestas con el fin de identificar elementos afines.

Posteriormente comienza el proceso de formulación de la propuesta. En este punto es donde se le define al futuro proyecto el tiempo estimado, los objetivos que se traza, entre otros elementos. Una vez conformada la propuesta, el planificador la envía a revisión. Los revisores de la entidad son los encargados de revisar dicha propuesta. Internamente en la entidad, la propuesta pasa por toda la jerarquía que exista definida de niveles de revisión. Una vez que esta propuesta llega al último nivel dentro de la entidad y es aprobada, regresa al planificador que la creó para comenzar el proceso de planificación.

En el proceso de planificación es donde se definen todas las acciones y montos necesarios para darle cumplimiento a los objetivos trazados. A partir de este momento deja de ser propuesta y se convierte en proyecto. Una vez planificado completamente el proyecto, se envía a revisión. En este caso vuelve a pasar por los mismos niveles de aprobación por los que pasó siendo propuesta dentro de la entidad.

Una vez que el proyecto sale de la entidad origen, comienza a transitar por todos los niveles de aprobación existentes hasta llegar al ente rector, en la mayoría de los casos es el Ministerio de Planificación y Desarrollo (MPD). El paquete de proyectos aprobados por el MPD es revisado por el Presidente, los que este último aprueba, se le asigna financiamiento y son los que conforman el Proyecto de Ley que es presentado a la Asamblea Nacional para ser aprobado. Desde que la Asamblea Nacional aprueba el Proyecto de Ley, los

proyectos que estén contemplados podrán comenzar su ejecución según lo planificado.

La siguiente figura representa un diagrama del proceso de formulación de la propuesta de proyecto:

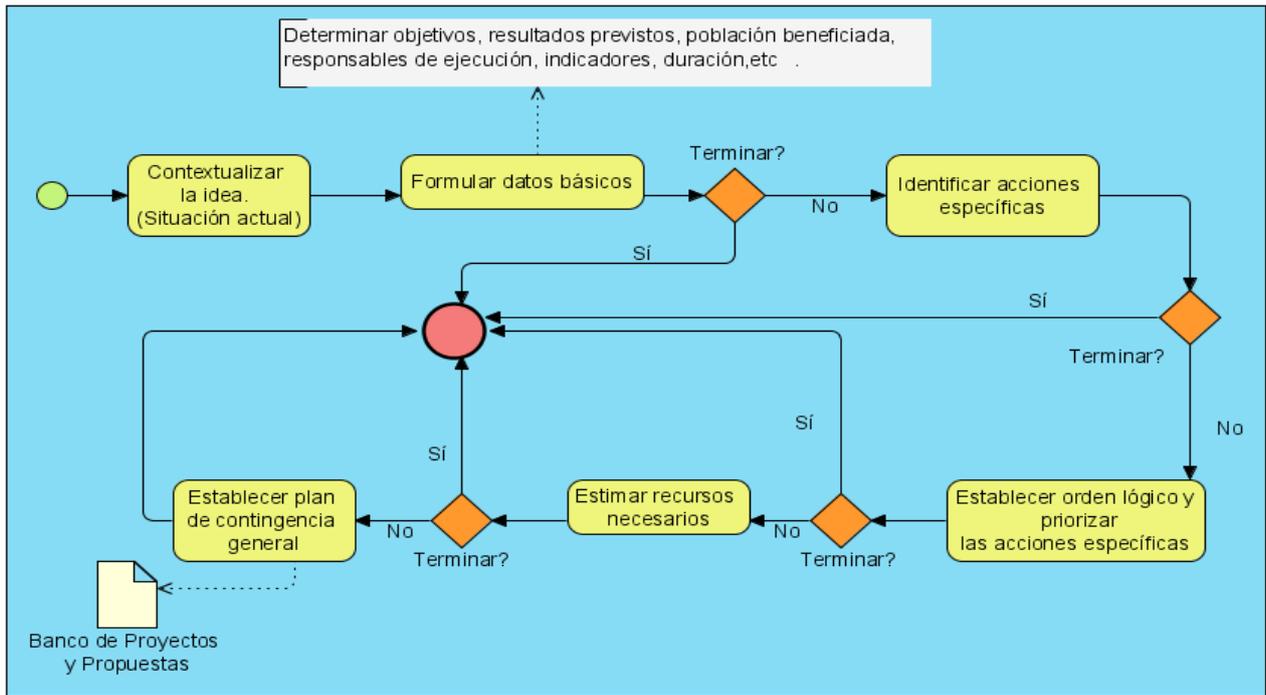


Figura 6 Diagrama del proceso de negocio Formulario propuesta

El modelado completo de los procesos de registro y aprobación de proyectos se encuentran en el anexo A

2.3 Reglas del negocio

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.

Para el modulo Registro y Aprobación de Proyectos del sistema SINAPSIS, se identificaron y clasificaron 19 reglas, a continuación se muestran algunas de ellas:

Nombre	Para solicitar la aprobación de una propuesta/proyecto es necesario que dicha propuesta/proyecto tenga todos los campos requeridos
Identificador	RN. 1
Tipo	Restricciones de operaciones

Tabla 2 Regla del negocio 1

Nombre	Los proyectos no pueden ser eliminados
Identificador	RN. 7
Tipo	Modelo de datos

Tabla 3 Regla del negocio 7

Nombre	Si se solicita modificar una propuesta esta pasa a ser nuevamente formulada por el planificador que la creó
Identificador	RN. 11
Tipo	Flujo

Tabla 4 Regla del negocio 11

Nombre	La suma de los recursos financieros asignados a las entidades adscritas debe ser igual a los recursos asignados a la entidad de adscripción
Identificador	RN. 19
Tipo	Calculo

Tabla 5 Regla del negocio 19

Las reglas del negocio restantes se encuentran en el anexo B

2.4 Requerimientos

Para realizar la Ingeniería de Requerimientos del módulo Registro y Aprobación de Proyectos se utilizaron las siguientes técnicas de obtención de requerimientos: La entrevista en combinación con la técnica JAD permitió que se fueran concretando las ideas sobre cada una de las funcionalidades que el sistema debía cumplir, sirvió además el estudio de la documentación del sistema Nueva Etapa.

Los requerimientos previamente obtenidos fueron analizados para agruparlos en funcionales y no funcionales. Siendo los primeros, capacidades o condiciones que el sistema debe cumplir, y los segundos, propiedades o

calidades que el producto debe tener.

Los funcionales fueron clasificados además según su prioridad en Alta (Esencial), Media (Deseado) o Baja (Opcional), en función de las necesidades de los clientes, y según su costo de desarrollo en Alto, Medio o Bajo.

Los no funcionales a su vez, se clasificaron según las categorías de Usabilidad, Fiabilidad, Eficiencia, Seguridad, Portabilidad, Reusabilidad y Capacidad (Consumo de recursos).

2.2.1 Requerimientos funcionales

Nombre	Registrar datos necesarios para formular una propuesta
Especificación	<p>El sistema permitirá a todos los planificadores de los organismos responsables registrar toda la información necesaria para formular una propuesta de proyecto. Los datos básicos necesarios para formular la propuesta son los siguientes:</p> <ul style="list-style-type: none"> • Enunciado y descripción del problema o necesidad que da origen al proyecto. (Formato: alfanumérico; Obligatorio: No) • Denominación de la propuesta (Formato: alfanumérico; Obligatorio: Sí) • Gerente (Formato: alfabético; Obligatorio: Sí) • Descripción de la propuesta. (Formato: alfanumérico; Obligatorio: No) • Producto <ul style="list-style-type: none"> ○ Cantidad (Formato: numérico; Obligatorio: Sí) ○ Unidad de medida (Formato: alfabético; Obligatorio: Sí) ○ Masculino y Femenino (para el caso que la unidad de medida sea sensible al género) (Formato: numérico; Obligatorio: Sí) • Descripción del producto (Formato: alfanumérico; Obligatorio: No) • Nivel de visibilidad (Formato: alfabético ; Obligatorio: No) • Obedece a: (Lineamientos) (Formato: alfabético; Obligatorio: Sí) • Clasificación <ul style="list-style-type: none"> • Planificación <ul style="list-style-type: none"> ○ Tipo de proyecto (Formato: alfabético; Obligatorio: Sí) ○ Tipo de inversión (para el caso de que el tipo de proyecto sea de Inversión) (Formato: alfabético; Obligatorio: Sí)

	<ul style="list-style-type: none">● Presupuestaria<ul style="list-style-type: none">○ Tipo de proyecto (Formato: alfabético; Obligatorio: Sí)○ Sector (Formato: alfabético; Obligatorio: Sí)○ Subsector (Formato: alfabético; Obligatorio: Sí)● Costo total (Formato: numérico; Obligatorio: Sí)● Equivalente en: (Formato: alfabético; Obligatorio: Sí)● Necesidad de atención (Formato: alfabético; Obligatorio: Sí)<ul style="list-style-type: none">○ A corto plazo○ A mediano plazo○ A largo plazo● Estrategia de la nación. (Formato: alfabético; Obligatorio: Sí)● Estrategia institucional. (Formato: alfabético; Obligatorio: Sí)● Objetivo específico del proyecto (Formato: alfanumérico; Obligatorio: Sí)● Datos territoriales<ul style="list-style-type: none">● Internacional (Formato: alfabético; Obligatorio: No)<ul style="list-style-type: none">○ País● Nacional (Formato: alfabético; Obligatorio: No)<ul style="list-style-type: none">○ Estado○ Municipio○ Parroquia○ Centro poblado● Regional (Formato: alfabético; Obligatorio: No)<ul style="list-style-type: none">○ Región● Coordenadas geográficas (Formato: numérico; Obligatorio: Sí)<ul style="list-style-type: none">○ Latitud grados○ Latitud minutos○ Latitud segundos○ Longitud grados○ Longitud minutos○ Longitud segundos● Resultados esperados<ul style="list-style-type: none">● Indicadores de la situación actual<ul style="list-style-type: none">○ Nombre del indicador (Formato: alfabético; Obligatorio: Sí)○ Valor (Formato: alfanumérico; Obligatorio: Sí)○ Fórmula (Formato: alfabético; Obligatorio: Sí)○ Fecha de la última información (Formato: dd/mm/aaaa; Obligatorio: Sí)○ Fuente (Formato: alfabético; Obligatorio: Sí)● Indicadores esperados<ul style="list-style-type: none">○ Nombre del indicador (Formato: alfabético; Obligatorio: Sí)
--	--

	<ul style="list-style-type: none"> ○ Valor (Formato: alfabético; Obligatorio: Sí) ○ Fórmula (Formato: alfanumérico; Obligatorio: Sí) ○ Tiempo de impacto (Formato: numérico; Obligatorio: Sí) ● Empleos directos femeninos en fase de ejecución (Formato: numérico; Obligatorio: No) ● Empleos directos masculinos en fase de ejecución (Formato: numérico; Obligatorio: No) ● Empleos indirectos en fase de ejecución (Formato: numérico; Obligatorio: No) ● Empleos directos femeninos en fase de operación (Formato: numérico; Obligatorio: No) ● Empleos directos masculinos en fase de operación (Formato: numérico; Obligatorio: No) ● Empleos indirectos en fase de operación (Formato: numérico; Obligatorio: No)
Identificador	RN_P.3
Prioridad	Alta
Coste de Desarrollo	Alto

Tabla 6 Requerimiento funcional 3

Nombre	Modificar datos de propuesta de proyecto formulada
Especificación	El sistema permitirá al planificador de cada organismo responsable de llevar a cabo el proyecto modificar los datos de la propuesta formulada una vez que se haga la solicitud previamente. Los datos modificables de la propuesta son los mismos especificados en RN_P.3
Identificador	RN_P.8
Prioridad	Alta
Coste de Desarrollo	Alto

Tabla 7 Requerimiento funcional 8

Nombre	Solicitar aprobación de propuesta de proyecto formulada
Especificación	El sistema permitirá al planificador encargado de formular la propuesta, una vez que esta ha sido conformada con todos los campos requeridos, solicitar la aprobación al revisor de nivel inmediato superior. Una vez que este último lo apruebe se solicita la aprobación a su nivel inmediato superior. Así sucede tantas veces como niveles de aprobación existan
Identificador	RN_P.18

Prioridad	Alta
Coste de Desarrollo	Medio

Tabla 8 Requerimiento funcional 18

La especificación de requerimientos funcionales se puede encontrar en el anexo C

2.2.2 Requerimientos no funcionales

Nombre	Contar con una ayuda en línea para guiar en el uso de la interfaz
Especificación	El sistema deberá proporcionar ayudas en línea, según la página donde el usuario se encuentre. La ayuda tendrá un nivel de detalle suficiente para que el usuario que no conoce la funcionalidad del sistema, entienda el contexto y pueda entender las operaciones del mismo, de tal forma que pueda, de manera intuitiva, determinar el paso que debe seguir para ejecutar lo que el sistema le permite. La ayuda debe contar con una descripción de cada una de las funciones posibles a realizar según el contexto y con imágenes que complementen esta información
Identificador	RNF.5
Tipo	Usabilidad

Tabla 9 Requerimiento no funcional 5

Nombre	Responder en tiempos aceptables las peticiones que se realicen en el sistema
Especificación	El sistema debe ser capaz de dar respuestas a las peticiones con un nivel aceptable de desempeño. Teniendo en cuenta el nivel de concurrencia que pueda existir, debe ser capaz de prestar servicio sin que se deterioren los tiempos de respuestas
Identificador	RNF.9
Tipo	Eficiencia

Tabla 10 Requerimiento no funcional 9

Nombre	El sistema debe ser una aplicación Web
Especificación	El sistema se debe ejecutar sobre un entorno web de manera que se permita su acceso desde cualquier punto del país utilizando la red
Identificador	RNF.12

RF_P.4	X	X									
RF_P.5	X	X									
RF_P.6	X	X									
RF_P.7	X	X									
RF_P.8	X	X									
RF_P.9									X		
RF_P.10											X
RF_P.11									X		
RF_P.12											X
RF_P.13									X		
RF_P.14											X
RF_P.15									X		
RF_P.16											X
RF_P.17	X										
RF_P.18	X										
RF_P.19	X										
RF_P.20	X										
RF_P.21	X										
RF_P.22	X										

Tabla 12 Porción de la matriz de trazabilidad de requerimientos

La matriz de trazabilidad completa y la de seguimiento de dependencia se puede encontrar en el anexo E.

Los diagramas de requerimientos fueron modelados en Visual Paradigm, a partir de la especificación SysML descrita brevemente en el epígrafe 1.5. Los diagramas de requerimientos permitieron representar gráficamente las relaciones entre requerimientos y casos de uso, siendo una forma más, para controlar el cambio en los requerimientos y su impacto en las funcionalidades del sistema a desarrollar.

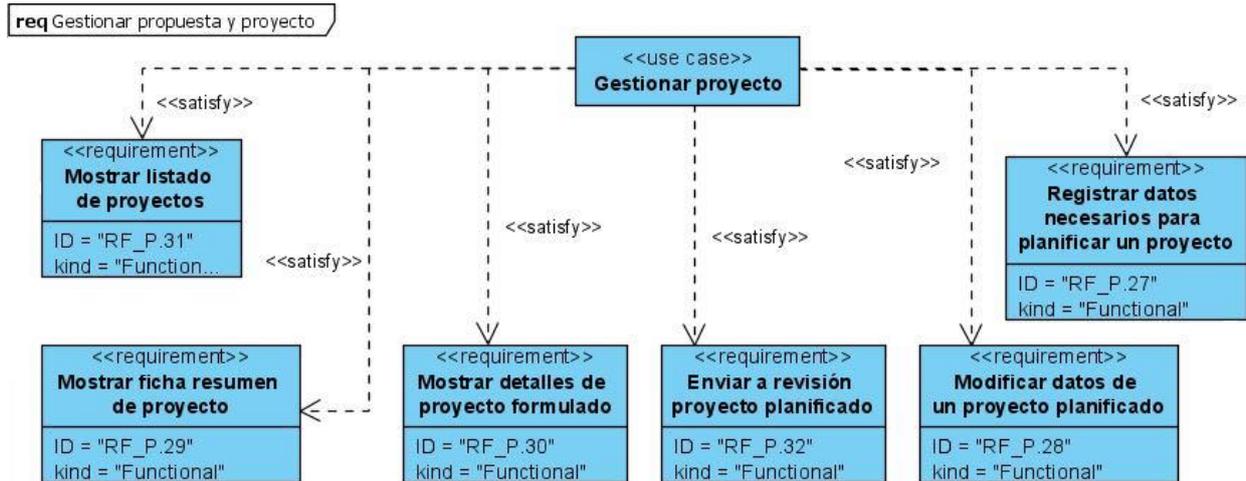


Figura 7 Porción del Diagrama de requerimientos del caso de uso *Gestionar proyecto*
 Los diagramas de requerimientos restantes se pueden apreciar en el anexo F

2.5 Actores del sistema

Los actores del sistema representan cada trabajador del negocio (inclusive si fuera un sistema ya existente) que tiene actividades a automatizar, además de los actores del negocio que interactúan de alguna forma con el sistema.

En este tópico se definirán los actores del sistema involucrados en el proceso de Registro y Aprobación de proyectos.

Actor	Descripción
SISPRE	El actor "SISPRE" es un sistema externo que interactúa con el sistema a través de una interfaz de comunicación. Este actor puede actualizar en el sistema los datos presupuestarios que fueron modificados.
Usuario	Posee los permisos necesarios para acceder a la funcionalidad de mostrar propuestas y proyectos enviados a revisión y la funcionalidad de mostrar proyectos aprobados. Es una generalización de los usuarios que son descritos a continuación.
Planificador	Gestiona la información de las propuestas y proyectos. Así como también de los indicadores iniciales y

	esperados, acciones específicas y actividades asociados al proyecto. Realiza la distribución territorial. Puede buscar proyectos y usuarios existentes en el sistema.
Revisor	Puede solicitar modificación de una propuesta o proyecto determinado, así como la aprobación de ambos. Constituye una generalización de los usuarios que son descritos a continuación
Responsable de Asignación de Recursos Financieros a Entidades	Asigna recursos financieros a las entidades adscritas de la entidad a la que él pertenece.
Responsable de Asignación de Recursos Financieros a Proyectos	Asigna recursos financieros a los proyectos pertenecientes a la entidad a la que él pertenece.
ONAPRE	Actualiza los datos presupuestarios de los proyectos aprobados para que se envíen al sistema SISPRE.

Tabla 13 Actores del sistema

2.6 Diagramas de Casos de Uso del Sistema

Luego de identificados los requerimientos y especificados en una plantilla formal, se estructura el diagrama de casos de uso del sistema, aplicando algunos de los patrones de casos de uso descritos en el epígrafe 1.10. Los casos de uso fueron organizados por paquetes, siendo estos los siguientes:

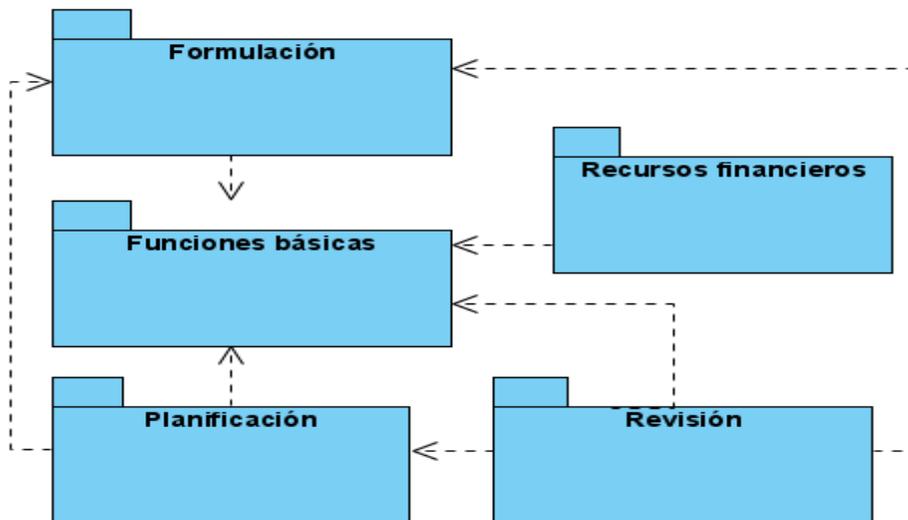


Figura 8 Paquetes del módulo Registro y Aprobación de Proyectos

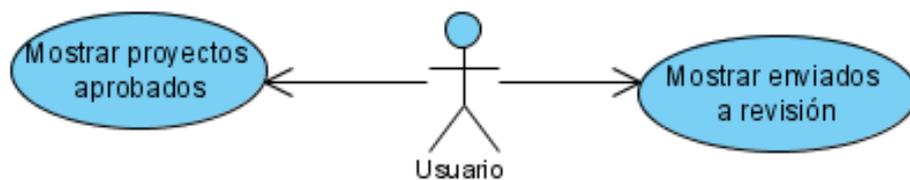


Figura 9 Diagrama de casos de uso del paquete Funciones básicas

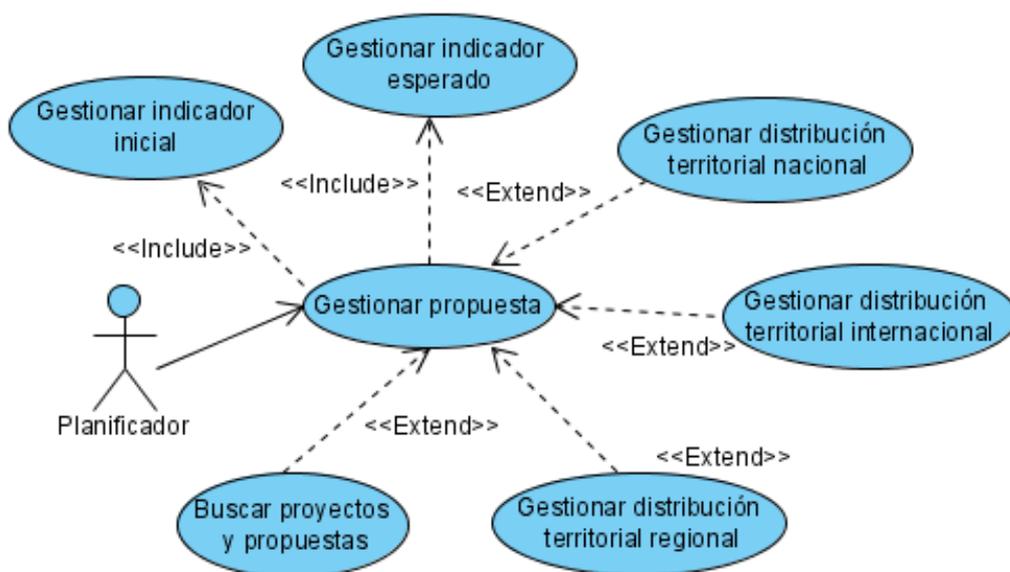


Figura 10 Diagrama de casos de uso del paquete Formulación

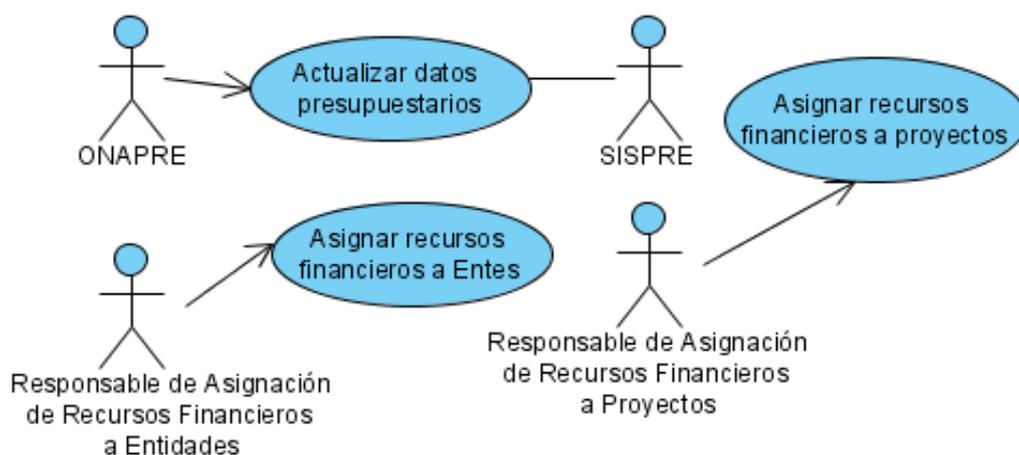


Figura 11 Diagrama de casos de uso del paquete Recursos financieros

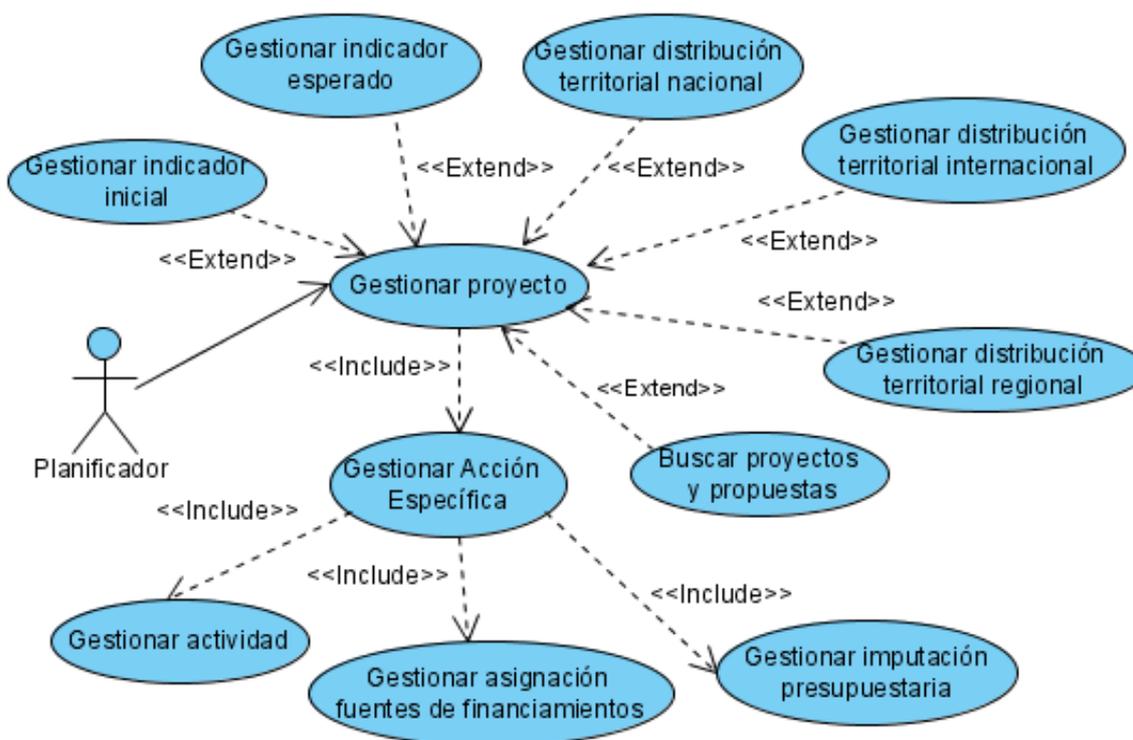


Figura 12 Diagrama de casos de uso del paquete Planificación

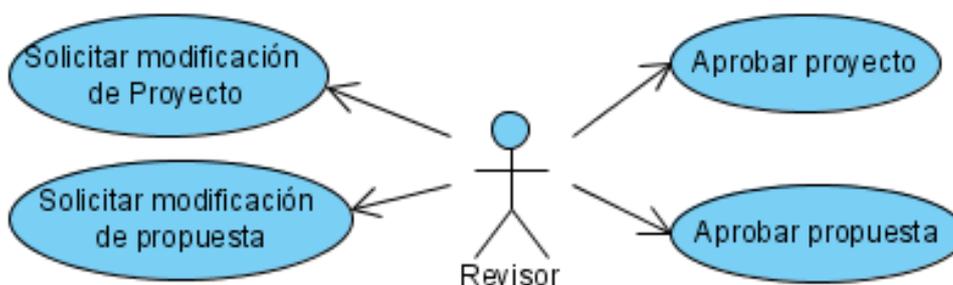


Figura 13 Diagrama de casos de uso del paquete Revisión

2.7 Especificación de Casos de Uso del Sistema

Partiendo de la premisa que ya se identificaron los actores y casos de uso apropiados del sistema, lo que corresponde es detallar los pasos necesarios para cumplir con dicho casos de uso, indicando cada uno de los eventos que ocurren durante el mismo. A continuación se especifican tres casos de uso del modulo Registro y Aprobación.

Caso de Uso:	Gestionar proyecto
Actores:	Planificador
Resumen:	El caso de uso se inicia cuando el planificador de una entidad

	necesita gestionar los datos de un proyecto. Consiste en que el planificador selecciona la opción correspondiente a planificar y llena los campos requeridos para la creación, también puede que seleccione un proyecto existente ya sea para eliminarlo, modificarlo, ver sus detalles, ver su ficha resumen o enviarlo a revisión. El caso de uso termina con la creación, modificación o eliminación de un proyecto.
Precondiciones:	<ul style="list-style-type: none"> • El sistema debe estar instalado y ejecutándose correctamente. • El actor debe estar autenticado con los permisos necesarios.
Referencias	RF_P. 5, RF_P.6, RF_P. 7, RF_P.27, RF_P. 28, RF_P.29, RF_P.30, RF_P.31, RF_P. 32, CU Buscar propuestas y proyectos (extiende), CU Gestionar Indicador inicial (extiende), CU Gestionar Indicador esperado (extiende), CU Buscar Usuario (extiende), CU Gestionar distribución territorial internacional (extiende), CU Gestionar distribución territorial nacional (extiende), CU Gestionar distribución territorial regional (extiende), CU Gestionar Acción Específica (incluido)
Reglas negocio del	<ul style="list-style-type: none"> • El planificador solo puede modificar un proyecto si no ha sido enviado a revisión. • Los proyectos no pueden ser eliminados. • Si el proyecto es modificado debe pasar todo el flujo de aprobación desde el organismo responsable.
Prioridad	Crítico
Complejidad	Complejo
Nivel del caso de uso	Usuario

Tabla 14 Resumen de la descripción textual del caso de uso del sistema Gestionar

Caso de Uso:	Gestionar acción específica
Actores:	Planificador
Resumen:	El caso de uso se inicia cuando es invocado desde el CU Gestionar proyecto. Consiste en que el planificador de una entidad necesita gestionar los datos de una acción específica. El planificador selecciona la opción correspondiente a nueva acción específica y llena los campos requeridos para la creación, también puede que seleccione una propuesta existente ya sea para eliminarla, modificarla o ver sus detalles. El caso de uso termina con la creación, modificación o eliminación de una acción específica.
Precondiciones:	<ul style="list-style-type: none"> • El sistema debe estar instalado y ejecutándose correctamente. • El actor debe estar autenticado con los permisos necesarios.
Referencias	RF_A.33, RF_P. 34, RF_P.37, RF_P. 38, RF_P.39, RF_P.40, RF_P.41, RF_P. 42, CU Buscar usuario (incluido), CU Gestionar

	imputación presupuestaria (incluido), CU Gestionar asignación de fuente de financiamiento (incluido), CU Gestionar actividad (incluido), CU Generar diagrama de Gantt (incluido)
Reglas del negocio	<ul style="list-style-type: none"> • La suma de los montos de todas las acciones específicas de un proyecto es el costo total de dicho proyecto. • La fecha de inicio de un proyecto está determinada por la fecha de inicio de su primera actividad. • La fecha de fin de un proyecto está determinada por la fecha de fin de su última actividad.
Prioridad	Crítico
Complejidad	Complejo
Nivel del caso de uso	Sub-función

Tabla 15 Resumen de la descripción textual del caso de uso del sistema Gestionar acción específica

Caso de Uso:	Buscar proyectos y propuestas
Actores:	Usuario
Resumen:	El caso de uso se inicia cuando se necesita buscar proyectos o propuestas, especificando algunos criterios de búsqueda. Lo cual dará como resultado un listado de propuestas o proyectos, según los criterios especificados.
Precondiciones:	<ul style="list-style-type: none"> • El sistema debe estar instalado y ejecutándose correctamente. • El actor debe estar autenticado con los permisos necesarios.
Referencias	RF_P.1, RF_P.2, CU Gestionar propuesta (extendido), CU Gestionar proyecto (extendido)
Prioridad	Crítico
Complejidad	Simple
Nivel del caso de uso	Usuario

Tabla 16 Resumen de descripción textual del caso de uso del sistema Buscar propuestas y proyectos

Las especificaciones completas de los casos de uso se encuentran en el anexo G

2.8 Conclusiones

- Utilizando RUP y aprovechando sus principales características, se generaron los artefactos necesarios con un grado de detalle aceptable.

- Se identificaron las políticas que deben cumplirse, así como los procesos a automatizar, mediante el modelado de procesos del negocio utilizando BPMN.
- Se identificaron las funcionalidades y características del producto, reflejadas en el modelado del sistema, donde se especifican los actores y las acciones que estos realizan.
- La aplicación de patrones de casos de uso facilitó la organización estructural del diagrama de casos de uso del sistema, lo cual facilita su comprensión.
- Las matrices de trazabilidad y seguimiento de dependencia, así como el diagrama de requerimientos, sirvieron de apoyo para identificar, controlar y seguir los requerimientos y sus cambios.

CAPÍTULO 3 ANÁLISIS DE LOS RESULTADOS

3.1 Introducción

En este capítulo se aplican métricas y técnicas para medir la calidad de los requerimientos identificados. Para ello se llevan a cabo los procesos de verificación y validación de requerimientos. Además se aplica el método de Kano para medir la satisfacción del cliente en la fase de levantamiento de requerimientos del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS.

3.2 Verificación de requerimientos

Para la verificación de los requerimientos se llevaron a cabo tres actividades fundamentales: obtención de las métricas para los requerimientos, revisión de los requerimientos a partir de las métricas obtenidas y análisis del progreso de los requerimientos (45).

Las métricas identificadas se aplicaron para verificar la existencia de un grupo de propiedades que debe cumplir la especificación de requerimientos para garantizar su calidad, propuesta por Davis en el modelo de calidad que él define(42). Para la verificación de la especificación de casos de uso se aplicaron preguntas para comprobar la presencia de un conjunto de propiedades a partir de las propuestas de Davis(42), Cockburn(43) y las listas de chequeo de Lilly(44).

La primera de las propuestas se consideró por recoger varias de las propiedades que se utilizaron para la especificación de requerimientos y que son completamente válidas para la especificación de los casos de uso. La segunda y tercera se utilizaron por ser específicas para la especificación de casos de uso; como es el caso de las listas de chequeo de Lilly, que verifican el uso correcto de las relaciones entre casos de uso y la propuesta de Cockburn por constituir una de las guías más exhaustiva en el buen uso de las técnicas de casos de uso.

Modelos de calidad para requerimientos

A la hora de presentar un modelo de calidad para ingeniería de requerimientos las propuestas que se han estudiado, optan por una de las siguientes opciones: aportar una lista de características de calidad de los requerimientos o, en contraposición, proporcionar una taxonomía de defectos, entendiendo que un defecto en los requerimientos es la ausencia de alguna de las características de calidad.

Para la verificación de los requerimientos se utilizó el tipo de modelo de calidad que propone una lista de características. Uno de los trabajos más citados en lo que respecta a las clasificaciones de las características de calidad en los requerimientos es la propuesta de Davis. Su propuesta presenta un conjunto de veinticuatro propiedades deseables de los requerimientos, además de ciertas métricas para verificar hasta qué punto los requerimientos de una especificación cumplen esas propiedades. En el presente trabajo se realizó una selección de las que se consideraron de mayor importancia y se aplicaron las métricas correspondientes para verificar la presencia de dichas propiedades.

Métricas para la verificación de la especificación de requerimientos

En la siguiente tabla se muestran las métricas auxiliares obtenidas y aplicadas como resultado de la primera y segunda actividad (obtención de las métricas y revisión de los requerimientos) de la metodología aplicada.

Métrica	Descripción	Valor
RT	Total de requerimientos	93
NUI	Número de requerimientos para los que todos los revisores tuvieron interpretaciones idénticas.	91
RC	Cantidad de requerimientos cambiados (suma de los requerimientos insertados, modificados y eliminados)	10
CN	Cantidad de necesidades de los clientes	40
CNR	Cantidad de necesidades sin requerimientos	0
RRT	Cantidad de requerimientos reutilizados	36
RCI	Cantidad de requerimientos en conflictos internamente en la especificación.	0
RCE	Cantidad de requerimientos en conflictos con otra	0

documentación del sistema.

CRR	Cantidad de requerimientos repetidos	0
------------	--------------------------------------	---

Tabla 17 Métricas auxiliares aplicadas a la especificación de requerimientos

A continuación se muestran las métricas principales aplicadas a la especificación de requerimientos con las respectivas propiedades que verifican según la propuesta de Davis.

No.	Métrica	Fórmula	Propiedad	Valor
1.	Especificidad	$Q = NUI / RT * 100$	No ambigüedad	98 %
2.	Estabilidad	$ETR = (RT-RC)/RT*100$	Estabilidad	89 %
3.	Promedio de requerimientos por necesidades	$CNR = RT / CN$	Corrección	2.33
4.	Reutilización	$RR = RRT / RT * 100$	Reutilizabilidad	38 %
5.	Requerimientos en conflictos	$PRC= (RCI+RCE)/RT* 100$	Consistencia interna y externa	0 %
6.	Por ciento de redundancia	$RR = CRR / RT * 100$	No redundancia	0 %

Tabla 18 Métricas principales aplicadas a la especificación de requerimientos

Si se observa detenidamente los resultados de las tablas 17 y 18 se puede apreciar que el grado de reutilización de la especificación de requerimientos de proyectos anteriores es medio si se tiene en cuenta que en proyectos anteriores se ha trabajado la misma línea de gestión de proyectos.

Un alto porcentaje de reutilizabilidad se logra en gran medida si la especificación de proyectos anteriores se realiza con vista a una futura reutilización, utilizando para ello patrones de reutilización. Como se evidencia en este caso, no se dio la situación propicia para lograr altos niveles de reutilización, que es tan importante en esta etapa, de manera que se logre una estandarización cada vez más sólida de la especificación de requerimientos.

Otros de los resultados válidos a destacar es el caso del promedio de requerimientos por necesidades que se muestra en la Tabla 18. Si se analizan los valores se evidencia un promedio de 2.33 requerimientos por necesidad.

Este valor por sí solo no es totalmente determinante si se tiene en cuenta que se puede dar el caso que, varios requerimientos pueden tributar a una necesidad y existir una necesidad que no esté cubierta por ningún requerimiento, y sin embargo el promedio de requerimiento por necesidad da elevado.

Para complementar el resultado mencionado anteriormente es necesario recurrir a la métrica de Cantidad de necesidades sin requerimientos que se encuentra en la Tabla 17. Estas dos métricas permiten verificar el nivel de corrección que tiene la especificación de requerimientos. Como se puede apreciar este nivel supera la unidad, por lo que se puede afirmar que es adecuado.

Verificación de la especificación de casos de uso

A continuación se muestran la lista de comprobación aplicada que verifica la calidad de la especificación de los casos de uso. Las preguntas se han agrupado bajo una categoría relativa a una característica de calidad. En caso de alguna respuesta negativa, implica la presencia de un defecto en dicha especificación. En el proceso de revisión, la lista de comprobación se le aplica a las especificaciones de cada uno de los casos de uso.

Propiedad	Pregunta	Valor
Compleitud	¿Hay respuestas a todas las peticiones que el actor del caso de uso hace al sistema y viceversa?	Sí
	¿Se contemplan todos los posibles escenarios para poder alcanzar el objetivo del caso de uso?	Sí
	¿Se especifican todas las secuencias alternativas a la secuencia normal?	Sí
	¿Se contemplan todas las posibles excepciones a la secuencia normal?	Sí
Comprensibilidad	¿Es posible leer el caso de uso sin volver atrás en repetidas ocasiones?	Sí
	¿Es posible seguir la secuencia normal del caso de uso con la presencia de las relaciones <i>include</i> o <i>extend</i> ?	Sí

	¿Es posible seguir la secuencia de pasos con la existencia de pasos alternativos?	Sí
	¿Se han desglosados los pasos de los actores y del sistema de tal manera que permita al caso de uso avanzar a un ritmo adecuado?	Sí
Concisión	¿Se ha expresado el caso de uso con las palabras necesarias?	Sí
	¿Se han obviado los elementos o anotaciones innecesarias que podrían dificultar la lectura del caso de uso?	Sí
	¿Se han obviado las interacciones innecesarias entre el actor del caso de uso y los elementos del entorno?	Sí
Independencia del diseño	¿Describe el caso de uso sólo lo que debe hacer el sistema sin especificar cómo debe hacerlo?	Sí

Tabla 19 Lista de comprobación aplicada a la especificación de casos de uso

Los resultados obtenidos al aplicar la lista de chequeo que se muestra en la tabla anterior evidencian que la especificación de casos de uso fue satisfactoria. El resto de la lista de comprobación aplicada se puede consultar en el anexo H. Algunos de los defectos detectados en las primeras revisiones fueron de semántica, como es el caso de que la no coincidencia del número de páginas del documento con el total de páginas reflejadas en las reglas de confidencialidad y omisión de algunas de las secciones obligatorias definidas en la plantilla correspondiente.

El proceso de revisión de la especificación de requerimientos y casos de uso fue llevado a cabo por revisores de calidad del proyecto SINAPSIS y revisores de calidad UCI. El certificado de aceptación de la documentación revisada se encuentra en el anexo I.

3.3 Validación de requerimientos

Para la validación de los requerimientos se llevaron a cabo tres actividades fundamentales: validación de los requerimientos funcionales, validación de los requerimientos no funcionales y cierre de la versión de los requerimientos (1).

En la primera actividad, mediante *walkthroughs*, asistidos con prototipos de interfaz de usuario no funcionales, permitió una validación conjunta de forma sencilla guiada por los requerimientos funcionales descritos como casos de usos. Como resultado de esta primera actividad se obtuvo una aceptación y validación de los requerimientos funcionales por parte de los clientes y usuarios.

De forma paralela a la primera actividad se realizó la segunda, con el objetivo de validar los requerimientos no funcionales. Como resultado se obtuvo la aceptación y validación de los requerimientos no funcionales, de forma similar a la actividad anterior. A diferencia de la primera actividad, la única técnica que se utilizó fue la revisión por parte de los clientes y usuarios.

Al igual que en las actividades de análisis, en las que los requerimientos no funcionales no suelen contemplarse en las técnicas de modelado, en la validación tampoco parece existir una forma de integrarlos en un proceso de *walkthrough* o de prototipado.

3.4 Medición del grado de satisfacción del cliente

En esta sección se presenta los resultados obtenidos, para ello se muestran primeramente los requerimientos analizados y el formato de la encuesta aplicada para obtener información de cada uno de estos requerimientos. Luego se muestra el mapa de respuestas obtenidas en cada uno de los 15 requerimientos evaluados. Por último, se lleva a cabo la clasificación de los requerimientos según los resultados obtenidos.

Selección de los requerimientos y encuestas

Los requerimientos seleccionados fueron obtenidos a partir del catálogo de requerimientos que aparecen en el documento de especificación de requerimientos. Para la selección de la muestra se tuvo en cuenta que fueran requerimientos nuevos, es decir, requerimientos que no estuvieran presentes en el sistema anterior (Nueva Etapa), además de que no fueran demasiados para no agobiar al cliente en el llenado de la encuesta. Para cada requerimiento funcional y disfuncional, el encuestado debía seleccionar entre una de 5 opciones posibles, así mismo cada requerimiento se evaluó en una escala de importancia con 10 valores. En la siguiente tabla se muestra el listado de requerimientos.

No.	Requerimiento
1.	Buscar Proyectos y propuestas
2.	Buscar instituciones
3.	Calcular equivalentes de moneda
4.	Adicionar nuevo indicador
5.	Solicitar aprobación de proyecto
6.	Usar propuesta como plantilla
7.	Solicitar modificación de proyecto
8.	Generar diagrama de Gantt asociado a una acción específica
9.	Registrar datos de imputaciones presupuestaria
10.	Enviar datos presupuestarios a SISPRE
11.	Recibir datos presupuestarios de SISPRE
12.	Mostrar listado de entidades adscritas a un determinado órgano de adscripción
13.	Mostrar listado de proyectos asociados a una determinada entidad.
14.	Asignar recursos financieros a entidades adscritas.
15.	Asignar recursos financieros a proyectos

Tabla 20 Requerimientos a evaluar utilizando el método de Kano

La encuesta fue realizada vía correo electrónico. Para cada requerimiento se mostraban dos preguntas. Una medía la funcionalidad del requerimiento y la otra la disfuncionalidad. Para cada pregunta el encuestado tenía la posibilidad de seleccionar una de 5 respuestas. Luego de estas preguntas debía seleccionar el grado de importancia que le atribuía al requerimiento mostrado. La siguiente figura muestra el formato de encuesta utilizado.

Requerimiento 1: Buscar Proyectos y propuestas					Para esta característica seleccione una respuesta entre las siguientes:				
¿Cómo se sentiría si pudiera buscar proyectos y propuestas de manera fácil?					Respuestas: 1. Me agrada <input type="button" value="v"/>				
¿Cómo se sentiría si NO pudiera buscar proyectos y propuestas de manera fácil?					Respuestas: 2. Es de esperarse <input type="button" value="v"/>				
Por favor, haga una evaluación de este requerimiento									
	Para nada importante	Algo importante	Importante	Muy importante	Extremo importante				
Requerimiento 1	<input type="radio"/> 1 <input type="radio"/> 2	<input type="radio"/> 3 <input type="radio"/> 4	<input type="radio"/> 5 <input type="radio"/> 6	<input checked="" type="radio"/> 7 <input type="radio"/> 8	<input type="radio"/> 9 <input type="radio"/> 10				

Figura 14 Formato de encuesta. Captura de pantalla

Representación de las respuestas

En esta sección se muestra un análisis detallado del mapa de las respuestas, las cuales son agrupadas en una tabla de concentración, que corresponde a cada una de las preguntas del cuestionario. El objetivo fundamental de esta tabla es observar la dispersión de las respuestas. En el anexo J se muestran los resultados obtenidos en las encuestas para cada uno de los 15 requerimientos evaluados.

Para la realización de la tabla que agrupa los datos de clasificación de cada uno de los requerimientos según las respuestas de los encuestados, se utilizó la propuesta hecha por León Duarte en (40) como representación alterna de clasificación de los requerimientos. Esta interpretación de la clasificación (ver Tabla 21) de los requerimientos se basa en el incremento de la satisfacción (columna “Mejor”), o bien, en el decremento de ella (columna “Peor”), debido a una inclusión o no de una característica o requerimiento del producto. Para la obtención de los valores Mejor y Peor se utilizó la fórmula propuesta en (40).

$$Mejor = \frac{A + U}{A + U + O + I} \quad Peor = \frac{U + O}{A + U + O + I}$$

Las formulas se obtienen de la percepción de ser Mejor que la competencia al satisfacer requerimientos tipo atractivos y unidimensionales, o bien, la de ser Peor que la competencia al no satisfacer requerimientos del tipo unidimensional y obligatorios. En el denominador de ambas fórmulas aparece la suma de todos los atributos con excepción de las percepciones Inv y D, por su carácter confuso.

Otro de los valores reflejados en la Tabla 21 es la importancia promedio que le atribuyeron los encuestados a cada uno de los requerimientos (columna “Imp”). Las columnas C1 y C2 se obtienen de la multiplicación de los valores Mejor y Peor por la importancia promedio. Estos valores se grafican en una escala bidimensional de Mejor y Peor en donde se puede identificar la clasificación del tipo de requerimiento (ver Figura 14).

Req.	A	O	U	I	Inv	D	Imp.	Mejor	Peor	C1	C2
1	1	5	1	1	1	1	0,37	0,25	0,75	0,09	0,65
2	2	1	4	1	1	1	0,85	0,75	0,63	0,64	0,67
3	2	1	3	1	1	2	0,98	0,71	0,57	0,7	0,81
4	4	1	2	1	1	1	0,9	0,75	0,38	0,75	0,15
5	1	0	4	0	2	3	0,81	1	0,8	0,81	0,74
6	4	0	1	2	1	2	0,87	0,71	0,14	0,62	0,34
7	4	1	1	1	2	1	0,79	0,71	0,29	0,56	0,22
8	5	1	2	0	1	1	0,72	0,88	0,38	0,63	0,25
9	1	1	4	1	2	1	0,95	0,71	0,71	0,68	0,64
10	4	0	2	1	3	0	0,89	0,86	0,29	0,76	0,12
11	3	2	2	0	2	1	0,85	0,71	0,57	0,6	0,14
12	1	3	2	1	2	1	0,47	0,43	0,71	0,2	0,62
13	3	4	2	1	0	0	0,31	0,5	0,6	0,15	0,57
14	2	5	2	1	0	0	0,32	0,4	0,7	0,13	0,67
15	3	2	1	4	0	0	0,87	0,4	0,3	0,35	0,29

Tabla 21 Clasificación de los requerimientos

Clasificación de los requerimientos

Una vez analizados los datos recogidos en las encuestas se clasificaron los requerimientos de la siguiente manera. De los quince requerimientos analizados, resultaron atractivos 6, unidimensionales 4, obligatorios 4 y 1 como indiferente. La distribución de los requerimientos se puede ver en la Figura 14, al igual que la Tabla 22 donde se listan los requerimientos con su respectiva clasificación.

Los resultados evidencian claramente que la satisfacción del cliente es elevada. Es buena señal que la mayor cantidad de los requerimientos clasificados están en el rango de los atractivos. Estos son los requerimientos que más satisfacción causan en los clientes, además de los unidimensionales y obligatorios.

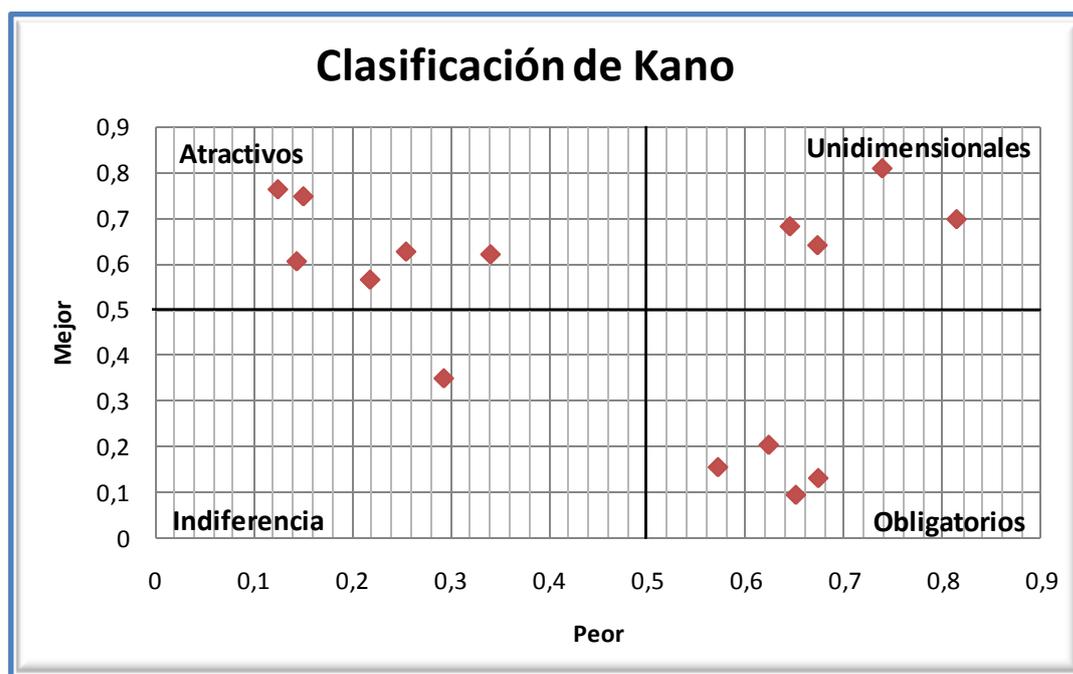


Figura 15 Clasificación de los requerimientos en función de los resultados obtenidos

No.	Requerimiento	Clasificación
1.	Buscar Proyectos y propuestas	Obligatorio
2.	Buscar instituciones	Unidimensional
3.	Calcular equivalentes de moneda	Unidimensional
4.	Adicionar nuevo indicador	Atractivo
5.	Solicitar aprobación de proyecto	Unidimensional
6.	Usar propuesta como plantilla	Atractivo
7.	Solicitar modificación de proyecto	Atractivo
8.	Generar diagrama de Gantt asociado a una acción específica	Atractivo
9.	Registrar datos de imputaciones presupuestaria	Unidimensional
10.	Enviar datos presupuestarios a SISPRE	Atractivo
11.	Recibir datos presupuestarios de SISPRE	Atractivo
12.	Mostrar listado de entidades adscritas a un	Obligatorio

determinado órgano de adscripción

13.	Mostrar listado de proyectos asociados a una determinada entidad.	Obligatorio
14.	Asignar recursos financieros a entidades adscritas.	Obligatorio
15.	Asignar recursos financieros a proyectos	Indiferencia

Tabla 22 Clasificación de los requerimientos en función de los resultados obtenidos

3.5 Conclusiones

En este capítulo se ha hecho un análisis de los resultados obtenidos en el capítulo anterior.

- Se realizó la etapa de verificación y validación de los requerimientos. Posteriormente se hizo una medición del grado de satisfacción del cliente utilizando el método de Kano.
- Para la verificación se utilizaron métricas que dieron una medida del grado de calidad de la especificación de los requerimientos. Además para la especificación de los casos de uso se utilizaron listas de comprobación que garantizaban la presencia de determinadas características deseables en dichas especificaciones. Tanto las métricas como las listas de comprobación verificaban un conjunto de características deseables que propone el modelo de calidad de Davis. La aplicación de estas técnicas permitió verificar la calidad de los requerimientos.
- Para la validación se utilizó la técnica del *walkthrough* combinado con los prototipos de interfaz de usuario, por ser estas las que más se adapta a las necesidades de validación (36).
- Por último, se aplicó el método de Kano para medir la satisfacción del cliente. Se realizó una encuesta a un total de 10 personas. Luego se analizaron los resultados de las encuestas y se obtuvo una clasificación de los requerimientos, en función de la satisfacción del cliente. Los resultados arrojados fueron satisfactorios, con un nivel de satisfacción de los clientes bueno en la mayoría de los requerimientos analizados.

CONCLUSIONES GENERALES

- Con la realización del estudio de las tendencias actuales en la disciplina de la Ingeniería de Requerimientos, se pudo identificar los lenguajes de modelado adecuados para los procesos de negocio, casos de uso y requerimientos. Así como las herramientas y metodologías adecuadas. Los resultados del estudio permitieron obtener la siguiente selección:
 - RUP como metodología de desarrollo de software
 - BPMN, UML y SysML como lenguajes de modelado
 - Visual Paradigm Suite 3.0 como herramienta de modelado general, dado que contiene Business Process Visual Architech 2.0 para el modelado de procesos y Visual Paradigm for UML 6.0 para el modelado de sistema y diagramas de requerimientos.
 - Microsoft Office Visio 2007 como herramienta para la realización de los prototipos de interfaz de usuario.
- Los documentos de especificación de requerimientos y casos de uso recogieron la especificación de los requerimientos obtenidas a partir de las necesidades identificadas de los clientes. Las especificaciones obtenidas fueron evaluadas con la aplicación de un conjunto de métricas y revisiones por parte de un equipo técnico de calidad que certificaron su aceptación.
- Desde el comienzo de las actividades inherentes a la Ingeniería de Requerimientos aplicada al proceso de Registro y Aprobación de Proyectos, se realizaron un conjunto de actividades de forma paralelas que garantizaron mantener un seguimiento y control de los cambios en los requerimientos obtenidos. Para ello fue necesario aplicar algunas técnicas como la trazabilidad y modelado.
- Se logró demostrar la hipótesis planteada al comienzo de este trabajo, dado que los clientes quedaron satisfechos, y muestra de ello está en los resultados obtenidos luego de la aplicación del método de Kano para medir la satisfacción del cliente.

RECOMENDACIONES

- Continuar perfeccionando el modelado de sistema mediante la actualización de los cambios que sean necesarios durante las etapas de diseño, implementación y pruebas.
- Realizar el diseño e implementación del sistema a partir de modelado de sistema presentado.
- Utilizar los modelos de requerimientos realizados a partir de la especificación SysML, como guía en la etapa de administración de requerimientos para otros proyectos de la UCI.
- Profundizar en el estudio de las patrones lingüísticos y de reutilización de requerimientos para lograr especificaciones cada vez más estándares y correctas.
- Profundizar en el estudio de la aplicación de la Ingeniería de Requerimientos a los procesos de Gestión de Proyectos en general.
- Utilizar la propuesta de medición de la satisfacción del cliente presentada en este trabajo para otros proyectos de la UCI.

BIBLIOGRAFÍA

1. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000. págs. 5 - 8.
2. —. *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000. pág. XVI.
3. **Corporation, Rational Software.** *Rational Unified Process Extended Help*. 2003.
4. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000. págs. 11 - 16.
5. **Ambler, Scott W.** Scott W. Ambler's Home Page. [En línea] 3 de febrero de 2008. [Citado el: 17 de marzo de 2009.] <http://www.ambysoft.com/scottAmbler.html>.
6. **Ambler, Scott.** The Agil Unified Process (AUP). [En línea] 16 de Agosto de 2008. <http://www.ambysoft.com/unifiedprocess/agileUP.html>.
7. —. *The Agile Unified Process v1.1*. 2006.
8. **Wells, J. Donovan.** Extreme Programming: A gentle introduction. [En línea] 15 de Abril de 2001. <http://www.extremeprogramming.org/>.
9. **Beck, Kent.** *Extreme Programming Explained: Embrace Change*. s.l. : Addison-Wesley, 1999. 0201616416.
10. **Fernández Escribano, Gerardo.** *Introducción a Extreming Programming*. 2002.
11. Visual Paradigm. *Visual Paradigm for UML*. [En línea] <http://www.visual-paradigm.com/product/vpuml/>.
12. **Systems, Sparx.** *Guía de Usuario de Enterprise Architect*.
13. Apexnet Software Factory. [En línea] [Citado el: 2 de marzo de 2009.] <http://www.apexnet.com.ar/index.php/product/viewProducts/24/sl=0>.
14. **Zhao, Jie y Thomas, Dunstan.** Comparación de Herramientas de modelado UML: Enterprise Architect y Rational Rose. [En línea] 15 de junio de 2005. [Citado el: 2 de marzo de 2009.] <http://www.apexnet.com.ar/index.php/news/main/38/event=view>.
15. **White, Stephen A.** Object Management Group/Business Process Management Initiative. *Introduction a BPMN*. [En línea] Mayo de 2004. <http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf>.

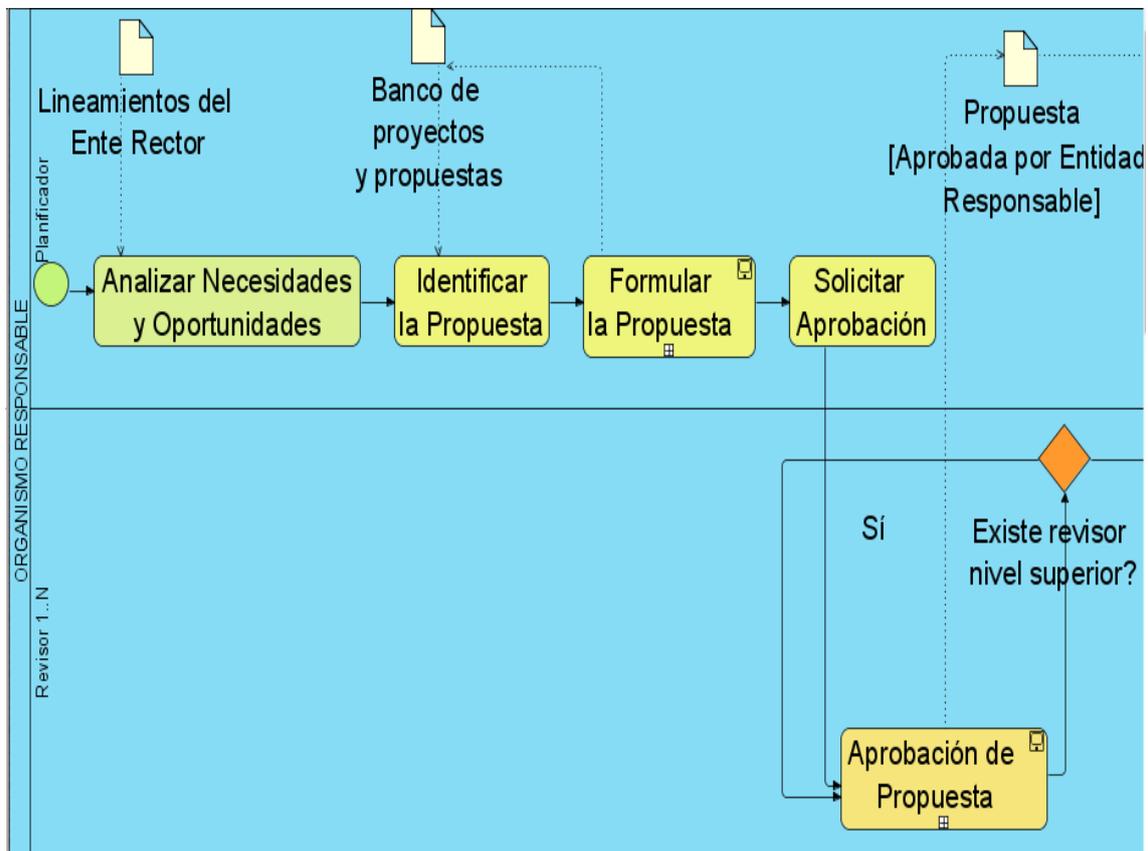
16. Object Management Group/Business Process Management Initiative. *Business Process Modeling Notation (BPMN) Information*. [En línea] OMG, 2 de Febrero de 2009. <http://www.bpmn.org/Documents/FAQ.htm>.
17. **Knowledge Based Systems, Inc.** Integrated Definition Method. [En línea] [Citado el: 2 de marzo de 2009.] <http://www.idef.com/IDEF0.html>.
18. OMG System Modelling Language. *The Official OMG SysML site*. [En línea] 19 de Septiembre de 2007. <http://www.omg.org/cgi-bin/apps/doc?formal/07-09-01.pdf>.
19. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Addison Wesley, 2000.
20. **Larman, Craig.** *UML y Patrones*. s.l. : Prentice Hall, 1999.
21. Visual Paradigm. *Business Process VISUAL ARCHITECT*. [En línea] 2 de Enero de 2007. http://content.europe.visual-paradigm.com/media/documents_download/bpva20ug/bpva_user_guide.pdf.
22. Visual Paradigm. *Business Process Modeling*. [En línea] <http://www.visual-paradigm.com/product/vpuml/demos/bpmodeling/>.
23. **Guevara Injoke, Marco Antonio y Flores Nazario, Cesar.** My Group Net. *La comunidad de programación*. [En línea] 2002. [Citado el: 10 de marzo de 2009.] http://www.mygnet.net/manuales/uml/manual_de_bpwin.1205.
24. **Corp, Microsoft.** Información general del producto Microsoft Office Visio 2007. [En línea] 23 de mayo de 2006. <http://www.microsoft.com/latam/office/preview/programs/visio/overview.mspx>.
25. —. Microsoft Office Visio Homepage. [En línea] [Citado el: 5 de abril de 2009.] <http://office.microsoft.com/es-es/visio/FX100487863082.aspx>.
26. **Young, Ralph R.** *The Requirements Engineering Handbook*. Londres : s.n., 2004.
27. **IEEE.** *IEEE Computer Dictionary - Compilation of IEEE Standard Computer Glossaries*. 1990.
28. **Kontoya, G y Sommerville, I.** *Requirements Engineering: Processes and Techniques*. 2000.
29. **Pressman, Rogen S.** *Ingeniería de Software. Un enfoque práctico*. s.l. : Addison Wesley, 2005.

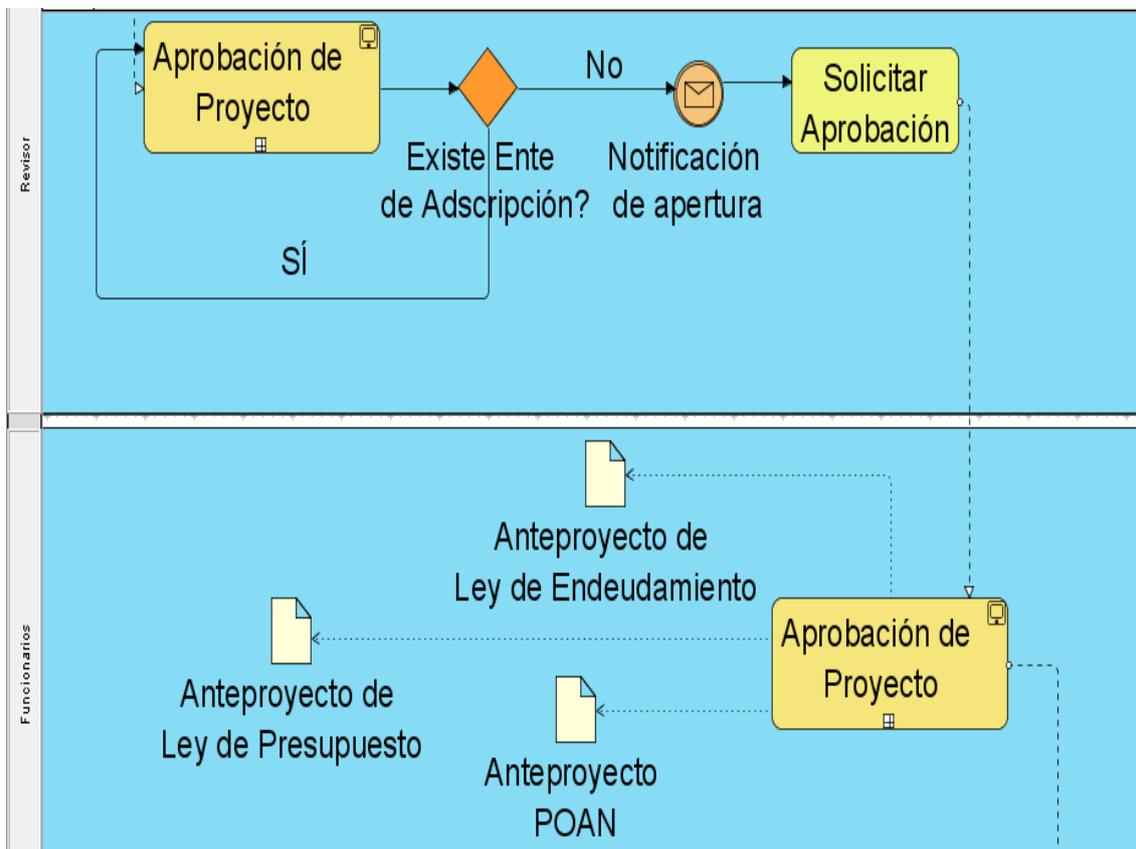
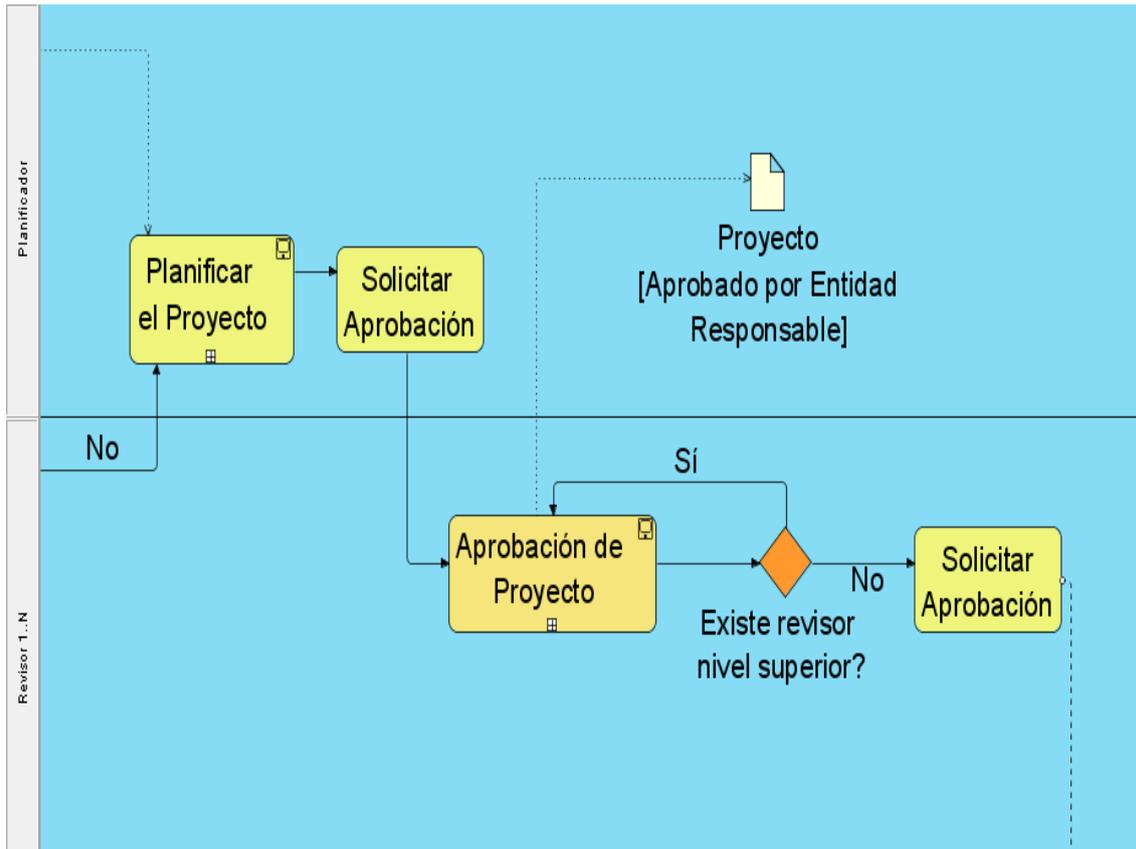
30. **IEEE, SWEBOK.** *Guide to the Software Engineering Body of Knowledge.* 2004.
31. **Escalona, Jose María y Koch, Nora.** *Ingeniería de Requisitos en Aplicaciones para la Web - Un estudio comparativo.* 2002.
32. **Antonelli, Leandro y Oliveros, Alejandro.** *Revisión de las fuentes usadas para elicitar requerimientos.* 2003.
33. **Sumano, María de los Angeles.** *Análisis de Requerimientos de Software. Estado del Arte.* 1999.
34. **Wieggers, Karl E.** *More About Software Requirements: Thorny Issues and Practical Advice.* 2006. 0735622671.
35. **Adriano, Natalia.** *Comparación del Proceso de Elicitación de Requerimientos en el desarrollo de Software a Medida y Empaquetado. Propuesta de métricas para la elicitación.* 2006.
36. **Durán, Amador.** *Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información.* Universidad de Sevilla : s.n., 2000.
37. **NASA.** Software Assurance Technology Center (SATC). [En línea] [Citado el: 2 de marzo de 2009.] <http://satc.gsfc.nasa.gov/tools/arm/>.
38. **Zeithaml, Parasuraman & Berry.** *Delivering Quality Service; Balancing Customer Perceptions and Expectations.* s.l. : Free Press, 1990.
39. **Center, National Quality Research.** American Customer Satisfaction Index Homepage. [En línea] [Citado el: 2 de abril de 2009.] <http://www.theacsi.org/>.
40. **León Duarte, Jaime Alfonso.** *Metodología para la detección de requerimientos subjetivos en el diseño de producto.* 2005.
41. **Gunnar Övergaard, Karin Palmkvist.** *Use Cases Patterns and Blueprints.* s.l. : Addison Wesley Professional, 2004.
42. *Identifying and measuring quality in software requirements specification.* **A. Davis, S. Overmyer, K. Jordan, J. Caruso, F. Dandashi, A. Dinh, G. Kincaid, G. Ledebor, P. Reynolds, P. Sitaran, A. Ta, M. Theofanos.** Los Alamitos, California : s.n., 1993.
43. **Cockburn, A. A. Cockburn.** *Writing Effective Use Cases.* s.l. : Addison-Wesley, 2001.
44. **Lilly, S.** *Use Case-Based Requirements: Review Checklist.* 1999.

45. Award Winning UML Tool. *Visual Paradigm for UML User's Guide*. [En línea] 1 de Agosto de 2007. http://content.europe.visual-paradigm.com/media/documents_download/vpuml61ug1/vpuml_user_guide.pdf.
46. **Ambler, Scott**. Dr. Dobb's Portal. *The Agile Edge: Unified and Agile*. [En línea] 1 de Enero de 2006. <http://www.ddj.com/architect/184415460> The Agile Edge: Unified and Agile.
47. —. Test Driven Development (TDD). [En línea] 2006. <http://www.agilemodeling.com/essays/tdd.htm>.
48. —. Agile Model Driven Development (AMDD): The Key to Scaling Agile Software Development. [En línea] 2006. <http://www.agilemodeling.com/essays/amdd.htm>.
49. **Lutowski, Rick**. *Software Requirements. Encapsulation, Quality and Reuse*. 2005.
50. **Robertson, Suzanne y Robertson, James**. *Mastering the Requirements Process Second Edition*. s.l.: Addison Wesley Professional, 2006. 0-321-41949-9.
51. **Armas, Rolando**. *Desde ISO 9001 hacia CMMI, Pasos para la mejora de los procesos y métricas*. 2007.
52. **Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla**. Sitio Web de la Herramienta REM. [En línea] [Citado el: 20 de Marzo de 2009.] http://www.lsi.us.es/descargas/descarga_programas.php?id=3.
53. *Una propuesta para la verificación de requisitos basada en métricas*. **Bernárdez, B., Durán, Amador y Toro, M.** 2, 2004, Revista de Procesos y Métricas de las Tecnologías de la Información, Vol. 1. ISBN: 1698-2029.
54. **Bernárdez, Beatriz**. *Una Aproximación Empírica a la Verificación de Especificaciones de Requisitos para Sistemas de Información*. Universidad de Sevilla : s.n., 2003.

ANEXOS

Anexo A:







Nota: El diagrama del modelado de procesos ha sido particionado en las imágenes anteriores para incluirlos en este documento.

Anexo B:

Documento de Reglas del Negocio v1.0 del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS.

Anexo C:

Documento Especificación de Requerimientos del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS.

Anexo D:

Nombre	Cumplir con las pautas de diseño de las interfaces
Especificación	El sistema deberá tener una interfaz gráfica uniforme a través del mismo incluyendo pantallas, menús y opciones. Las pautas de diseño serán definidas por el equipo de diseño gráfico y se realizarán siguiendo los lineamientos de la arquitectura de información.
Identificador	RNF.1
Tipo	Usabilidad

Nombre	Agrupar vínculos y botones por grupos funcionales
Especificación	La consistencia de la interacción entre usuario y sistema estará determinada por el diseño de la interfaz de usuario que mantendrá los elementos como menús, banners y zona de trabajo, en posiciones fijas, además de la mayor uniformidad posible entre cuadros de texto y botones. El sistema deberá ser de uso intuitivo, de tal forma que se reduzca los tiempos de entrenamiento, soporte y prueba por parte del usuario. La agrupación de los botones por funcionalidad determinará además la capacidad de componer la interfaz de acuerdo a las funciones requeridas por un rol determinado
Identificador	RNF.2
Tipo	Usabilidad

Nombre	Mostrar los mensajes, títulos y demás textos que aparezcan en la interfaz del sistema en idioma español
Especificación	Tanto los títulos de los componentes de la interfaz, como los mensajes para interactuar con los usuarios, así como los mensajes de error, deberán ser en idioma español y tener una apariencia uniforme en todo el sistema. Los mensajes de error deberán ser lo suficientemente informativos para dar a

	conocer la severidad del error
Identificador	RNF.3
Tipo	Usabilidad

Nombre	Utilizar campos de selección en la interfaz en los casos que sea posible
Especificación	<p>El sistema deberá facilitar la entrada de datos a los usuarios, presentando campos de selección que permitan escoger los valores. Estos campos contendrán los valores posibles con los que se podrá llenar un determinado elemento en la interfaz, haciendo que el proceso de llenado de datos sea lo más intuitivo posible de tal forma que los usuarios se puedan adaptar con facilidad al sistema.</p> <p>Los elementos mostrados en estos campos de valores deben ser configurables en el sistema de manera que se puedan cambiar, eliminar o agregar nuevos</p>
Identificador	RNF.4
Tipo	Usabilidad

Nombre	Contar con una ayuda en línea para guiar en el uso de la interfaz
Especificación	<p>El sistema deberá proporcionar ayudas en línea, según la página donde el usuario se encuentre. La ayuda tendrá un nivel de detalle suficiente para que el usuario que no conoce la funcionalidad del sistema, entienda el contexto y pueda entender las operaciones del mismo, de tal forma que pueda, de manera intuitiva, determinar el paso que debe seguir para ejecutar lo que el sistema le permite. La ayuda debe contar con una descripción de cada una de las funciones posibles a realizar según el contexto y con imágenes que complementen esta información</p>
Identificador	RNF.5
Tipo	Usabilidad

Nombre	Permitir uso del teclado para realizar operaciones sobre el sistema
Especificación	<p>El sistema debe permitir su uso desde el teclado con funciones básicas como Tab para moverse por los campos de una página determinada, Enter para accionar un botón seleccionado. Se utilizarán tres funciones, dentro de las que se encuentran Ctrl+C para copiar una información seleccionada en un campo específico, Ctrl+V para pegar una información específica y Ctrl+X para cortar la información</p>

	seleccionada
Identificador	RNF.6
Tipo	Usabilidad

Nombre	Mostrar los valores de los campos numéricos utilizando separadores, según estándares
Especificación	El sistema debe ser preciso en cada una de las salidas debido al manejo de dígitos e información referente a diferentes montos asociados a los proyectos. Para esto debe seguir un estándar único en la codificación de las cifras y la separación de los dígitos, para la separación de miles se utilizará la coma y para las cifras decimales el punto (ej. 100. 000.000,00)
Identificador	RNF.7
Tipo	Usabilidad

Nombre	Prever contingencias para eventos de caída del sistema
Especificación	<p>El sistema deberá prever contingencias que pueden afectar la prestación estable y permanente del servicio. La siguiente es la lista de las contingencias que se deben tener en cuenta y se pueden considerar críticas:</p> <ul style="list-style-type: none"> • Sobrecarga del sistema por volumen de usuarios. • Caída del sistema por sobrecarga de procesos. • Caída del sistema por sobrecarga de transacciones. • Caída del sistema por volumen de datos excedido en la base o bodega de datos. <p>Estas consideraciones implicarán que la infraestructura técnica sobre la que se implantará el sistema garantice una alta disponibilidad del mismo</p>
Identificador	RNF.8
Tipo	Fiabilidad

Nombre	Responder en tiempos aceptables las peticiones que se realicen en el sistema
Especificación	El sistema debe ser capaz de dar respuestas a las peticiones con un nivel aceptable de desempeño. Teniendo en cuenta el nivel de concurrencia que pueda existir, debe ser capaz de prestar servicio sin que se deterioren los tiempos de respuestas
Identificador	RNF.9
Tipo	Eficiencia

Nombre	Permitir el intercambio de datos entre el cliente y el servidor por canales cifrados
Especificación	El sistema deberá permitir la transmisión por canales cifrados cuando se trate de información confidencial, de manera que no viaje en texto plano por la red
Identificador	RNF.10
Tipo	Seguridad

Nombre	Almacenar de manera cifrada las claves de los usuarios en la base de datos
Especificación	Las contraseñas de los usuarios no deben ser almacenadas en texto plano, esta información debe ser privada y específica de cada uno, de manera que nadie pueda reemplazar la identidad de un usuario en el sistema
Identificador	RNF.11
Tipo	Seguridad

Nombre	El sistema debe ser una aplicación Web
Especificación	El sistema se debe ejecutar sobre un entorno web de manera que se permita su acceso desde cualquier punto del país utilizando la red
Identificador	RNF.12
Tipo	Portabilidad

Nombre	Garantizar compatibilidad con navegadores de uso común
Especificación	El sistema deberá ser compatible con los navegadores: <ul style="list-style-type: none"> • Microsoft Internet Explorer 6.0 o superior • Mozilla Firefox 2.0 o superior
Identificador	RNF.13
Tipo	Portabilidad

Nombre	Utilizar estándares de codificación
Especificación	El código fuente del sistema deberá cumplir con un estándar de codificación. El estándar especificado debe considerar puntos como: Estándares de nombres utilizados en todos sus objetos: programas, formas, tablas, campos, índices, procedimientos, paquetes. Codificación de los comentarios para la generación automática de la documentación. Empleo

	de las características del IDE (Interface Development Enviroment) para el formato del código
Identificador	RNF.14
Tipo	Portabilidad

Nombre	Diseñar un sistema compuesto por módulos, que agrupen funcionalidad
Especificación	El sistema deberá garantizar que cada subsistema, aplicación y componente tienen fronteras claramente definidas y funciones relacionadas. Cada módulo será independiente del resto
Identificador	RNF.15
Tipo	Portabilidad

Nombre	Garantizar que las actualizaciones del sistema sean a nivel central (Servidor)
Especificación	El sistema deberá estar orientado a que las actualizaciones sólo se hagan en el sitio del servidor, de tal manera que no sea necesario actualizar todos y cada uno de los clientes que acceden a la información del sistema
Identificador	RNF.16
Tipo	Portabilidad

Nombre	Garantizar que los formatos de los archivos de salida del sistema sean compatibles con los programas más comunes
Especificación	Los ficheros que genere el sistema deben utilizar formatos estándares como (rtf, pdf, xsl) de manera que sean compatibles con las siguientes herramientas: <ul style="list-style-type: none"> • Microsoft Office 2003 o superior • Acrobat Reader 6.0 o superior • Open Office 2.3 o superior
Identificador	RNF.17
Tipo	Reusabilidad

Nombre	Definir un modelo tres capas para el sistema
Especificación	El sistema deberá considerar en su arquitectura un modelo tres capas, donde se definen tres componentes lógicos de manera independiente: capa de presentación o interfaz de usuario, capa de lógica de negocio y capa de datos. Esta

	arquitectura determina una separación entre la lógica del negocio y la presentación de la aplicación, lo que permite el uso de servicios desde la capa de lógica del negocio ya sea por la capa de presentación del sistema o por otros sistemas que requieran el mismo servicio
Identificador	RNF.18
Tipo	Reusabilidad

Nombre	Considerar características técnicas mínimas para la ejecución en clientes
Especificación	<p>Para que un cliente de la aplicación pueda ejecutar procesos, en línea, considerados en el sistema el punto de acceso deberá cumplir con los siguientes requisitos mínimos:</p> <ul style="list-style-type: none"> • Procesador 2.0 GHz • Memoria 512 MB. • Disco duro 20 GB. • Sistema Operativo Windows 98, 2000, XP o para Servidor o Linux. • Navegador internet Explorer 6.0 o posterior, Mozilla Firefox 2.X • Conexión a Internet. mínimo 56Kbps <p>Para que un cliente de la aplicación pueda observar y analizar resultados de los procesos consignados en documentos electrónicos, el punto de acceso deberá cumplir con los siguientes requisitos mínimos:</p> <ul style="list-style-type: none"> • Herramienta Microsoft Office 2003 o superior • Herramienta Acrobat Reader 6.0 o superior • Herramienta Open Office 2.3 o superior
Identificador	RNF.19
Tipo	Capacidad

RF_P.51					X																
RF_P.52					X																
RF_P.53						X															
RF_P.54						X															
RF_P.55												X									
RF_P.56														X							
RF_P.57														X							
RF_P.58														X							
RF_P.59							X														
RF_P.60							X														
RF_P.61								X													
RF_P.62								X													
RF_P.63								X													
RF_P.64																X					
RF_P.65																X					
RF_P.66																	X		X		
RF_P.67																			X		
RF_P.68									X												

Tabla 23 Matriz de trazabilidad de requerimientos (hasta 19/12/2008)

REQ.	CASOS DE USOS																				
	CU 4.1	CU 4.2	CU 4.3	CU 4.4	CU 4.5	CU 4.6	CU 4.7	CU 4.8	CU 4.9	CU 4.10	CU 4.11	CU 4.12	CU 4.13	CU 4.14	CU 4.15	CU 4.16	CU 4.17	CU 4.18	CU 4.19	CU 4.20	CU 4.21
RF_P.1																X					
RF_P.2																X					
RF_P.3	X																				
RF_P.4	X	X																			
RF_P.5	X	X																			
RF_P.6	X	X																			
RF_P.7	X	X																			
RF_P.8	X	X																			
RF_P.9										X											
RF_P.10											X										
RF_P.11										X											
RF_P.12											X										
RF_P.13										X											
RF_P.14											X										
RF_P.15										X											
RF_P.16											X										
RF_P.17	X																				
RF_P.18	X																				
RF_P.19	X																			X	
RF_P.20	X																				
RF_P.21	X																				
RF_P.22	X																				
RF_P.23												X									
RF_P.24														X							
RF_P.25															X						

RF_P.53						X														
RF_P.54						X														
RF_P.55												X								
RF_P.56													X							
RF_P.57													X							
RF_P.58													X							
RF_P.59							X													
RF_P.60							X													
RF_P.61								X												
RF_P.62								X												
RF_P.63								X												
RF_P.64															X					
RF_P.65															X					
RF_P.66																X	X			
RF_P.67																	X			
RF_P.68																X				
RF_P.69																X				
RF_P.70																		X		
RF_P.71																		X		
RF_P.72									X											
RF_P.73						X														
RF_P.74																				

Tabla 24 Matriz de trazabilidad de requerimientos (hasta 15/02/2009)

Nota: Las X resaltadas en rojo representan requerimientos cambiantes. RF_P.68, RF_P.69, RF_P.70, RF_P.71, RF_P.73 y RF_P.74 fueron agregados posterior a la fecha 19/12/2008.

Anexo F:

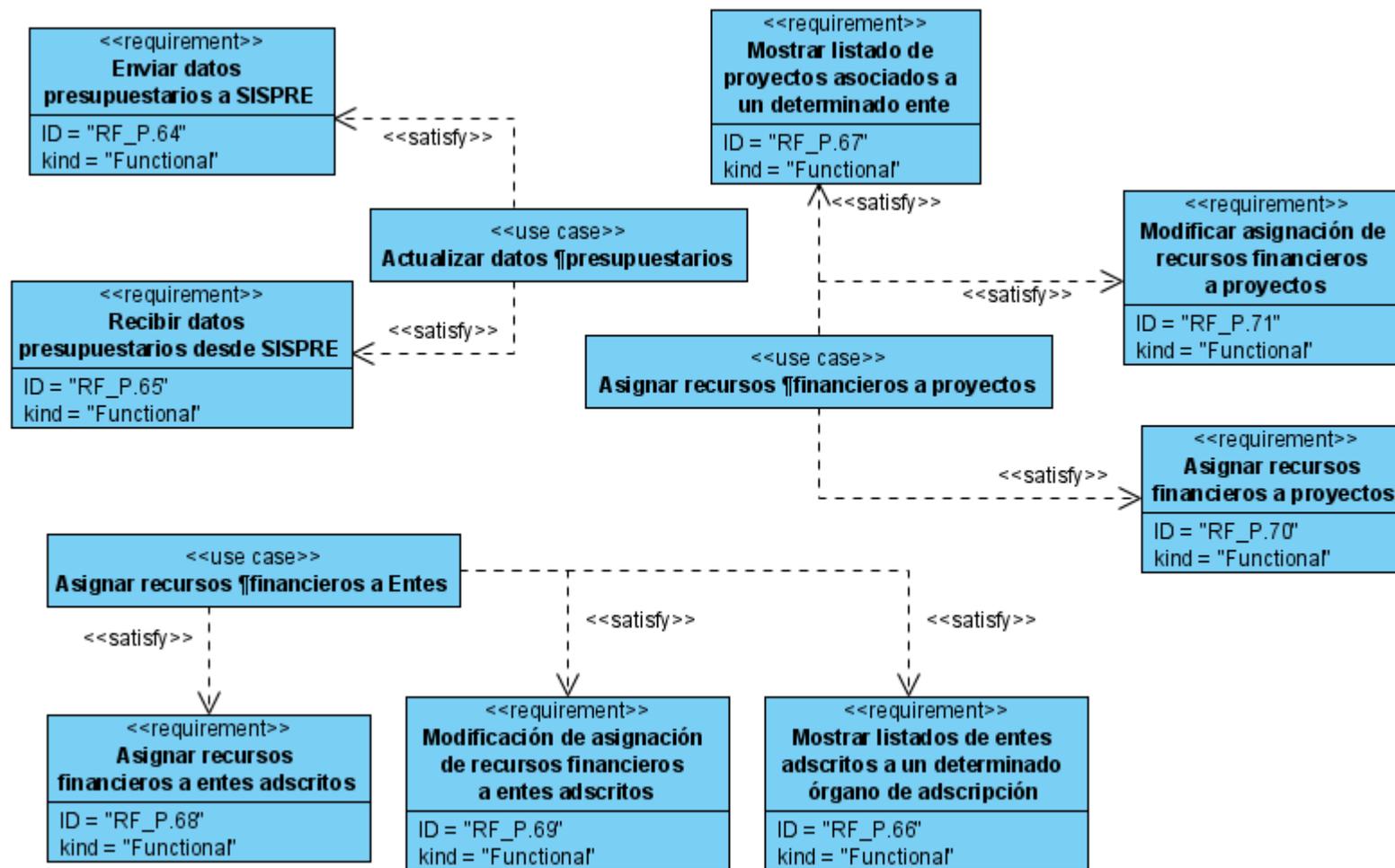


Figura 16 Diagrama de requerimientos de los casos de uso Actualizar datos presupuestarios, Asignar recursos financieros a Entes y Asignar recursos financieros a Proyectos

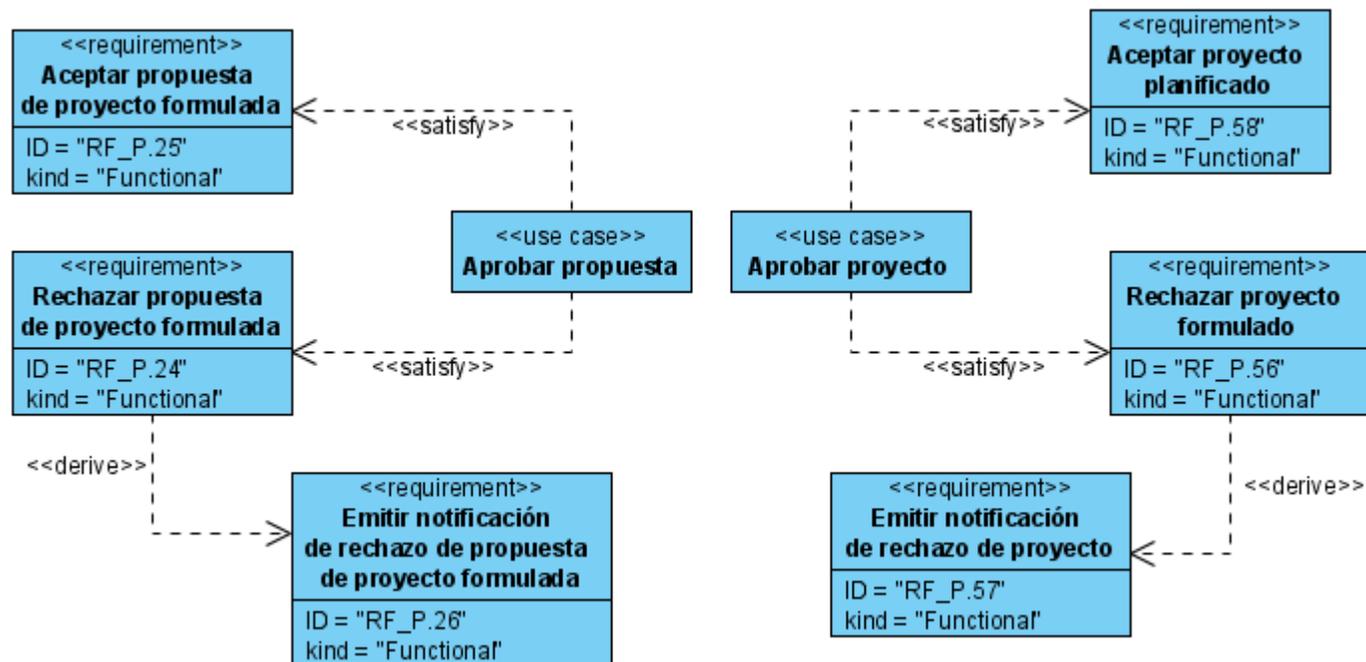


Figura 17 Diagrama de requerimientos de los casos de uso Aprobar propuesta y Aprobar proyecto

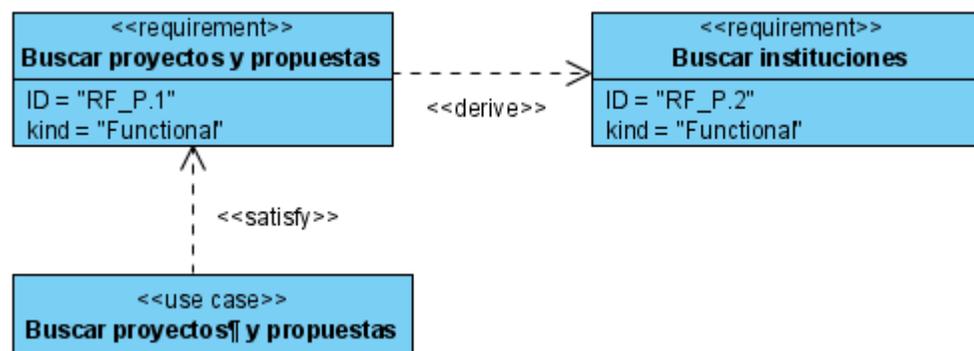


Figura 18 Diagrama de requerimientos del caso de uso Buscar proyectos y propuestas

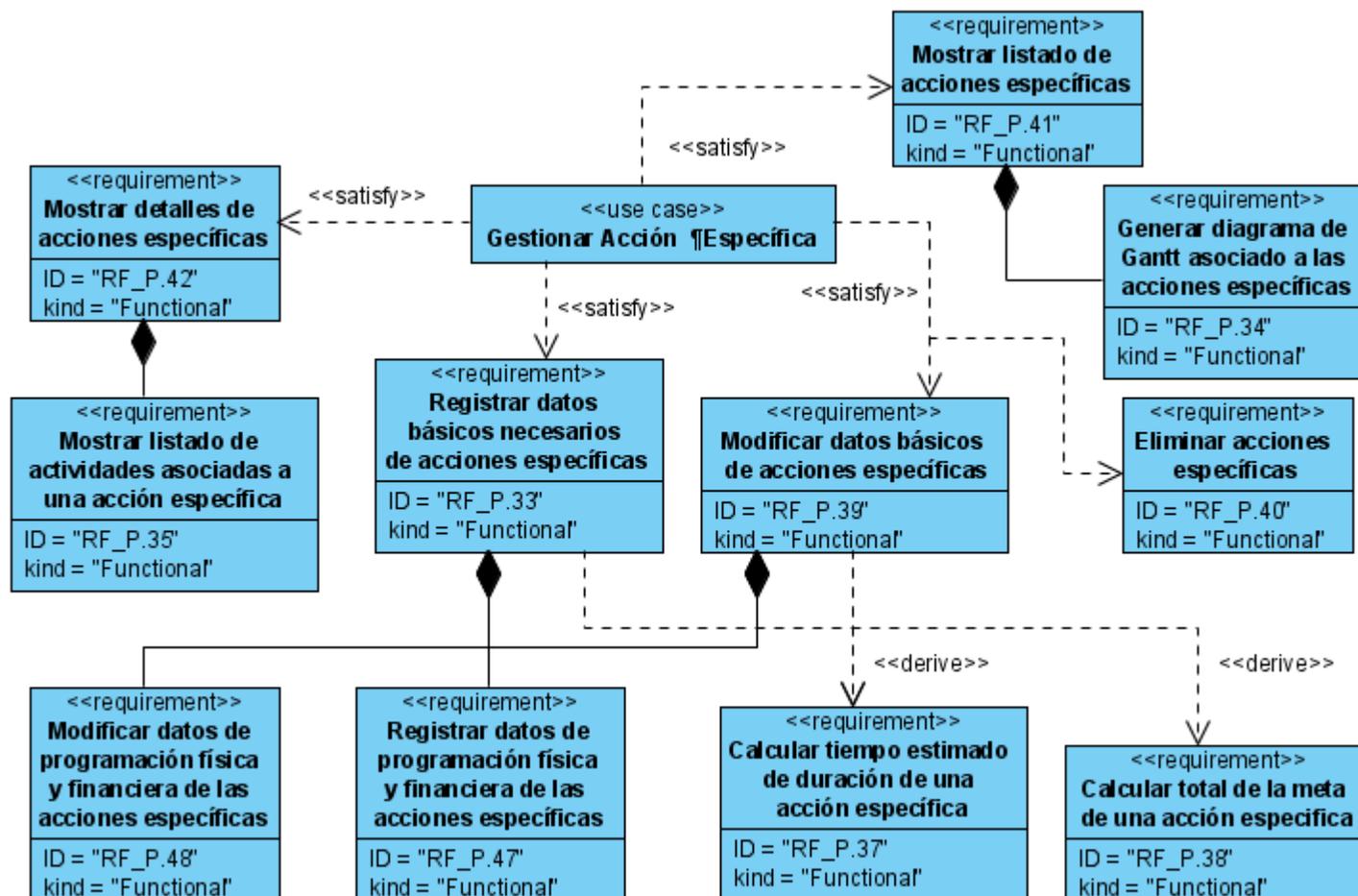


Figura 19 Diagrama de requerimientos del caso de uso Gestionar acción específica

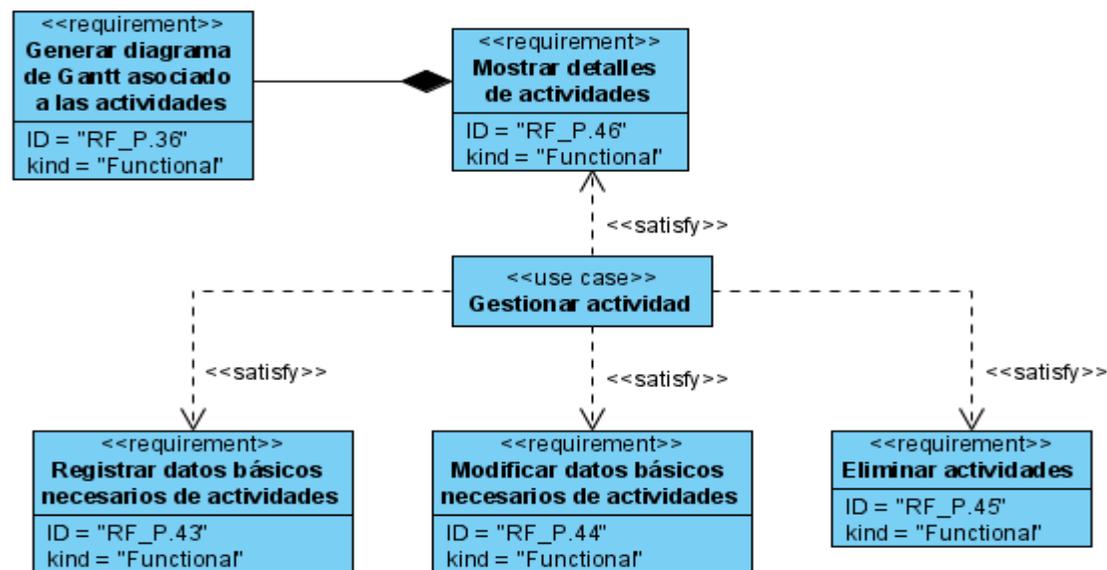


Figura 20 Diagrama de requerimientos del caso de uso Gestionar actividad

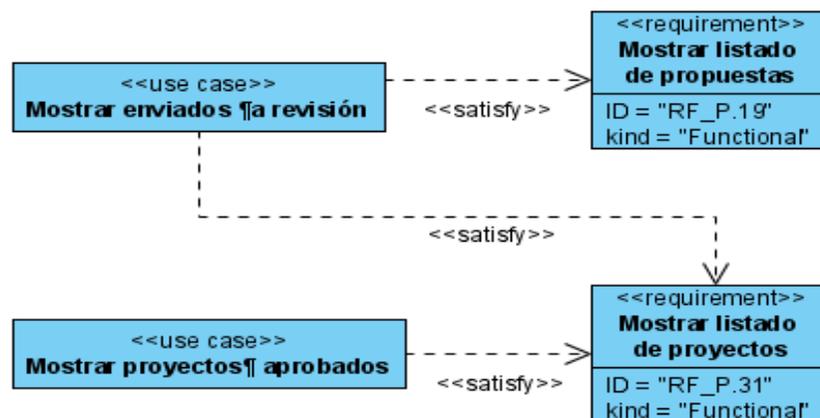


Figura 21 Diagrama de requerimientos de los casos de uso Mostrar enviados a revisión y Mostrar proyectos aprobados

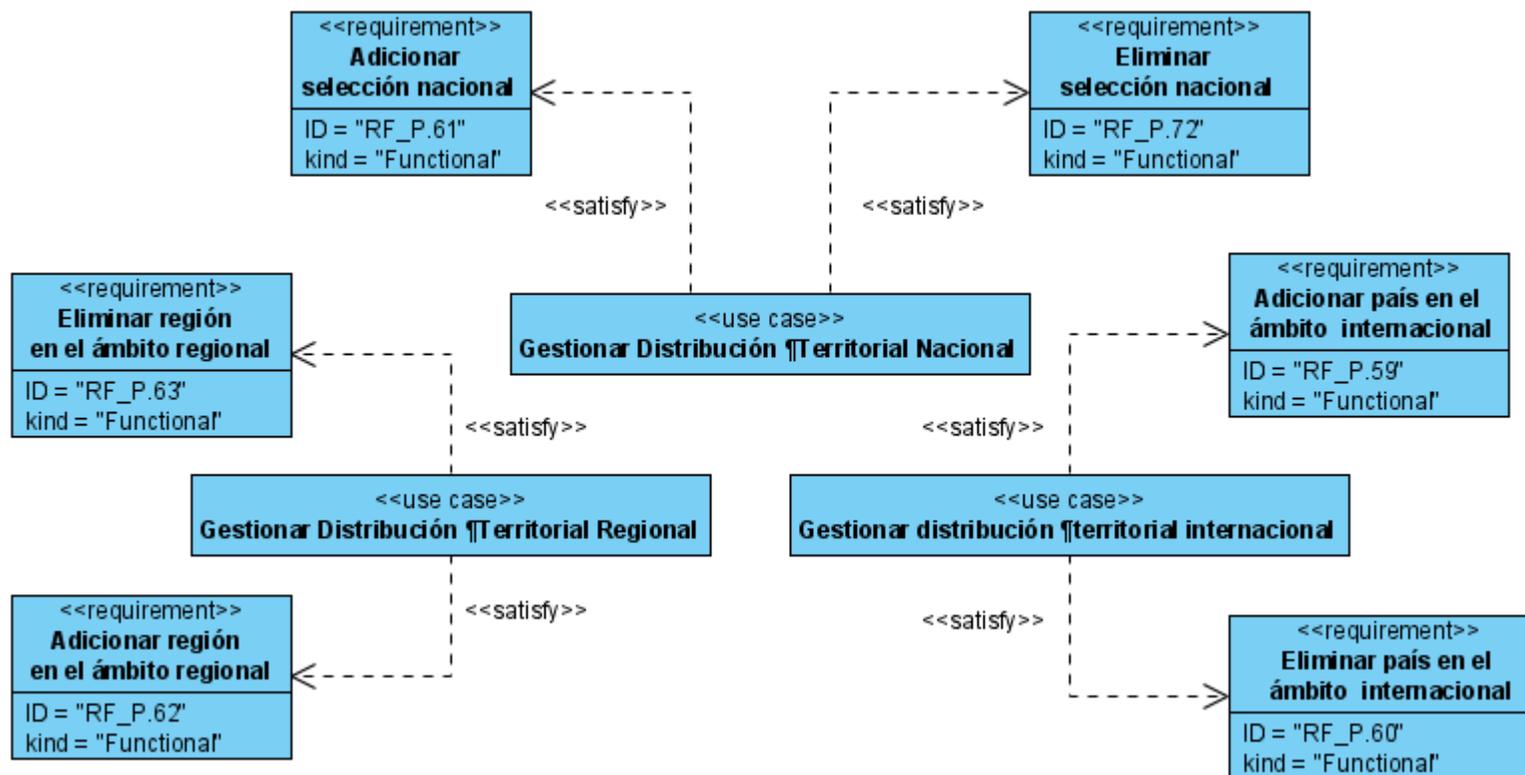


Figura 22 Diagrama de requerimientos de los casos de uso Gestionar distribución territorial regional, Gestionar distribución territorial internacional y Gestionar distribución territorial nacional

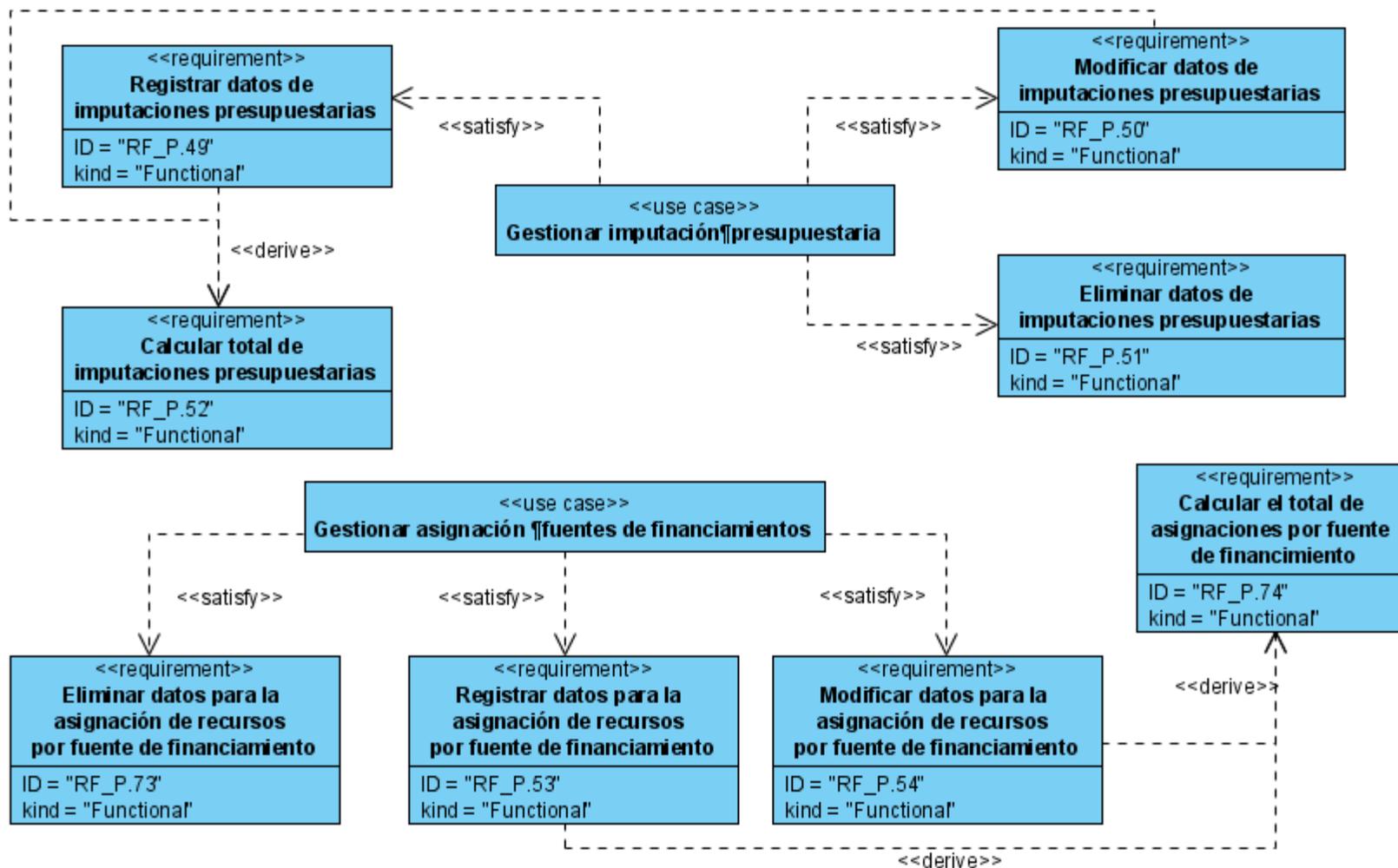


Figura 23 Diagrama de requerimientos de los casos de uso Gestionar imputación presupuestaria y Gestionar asignación de fuente de financiamiento

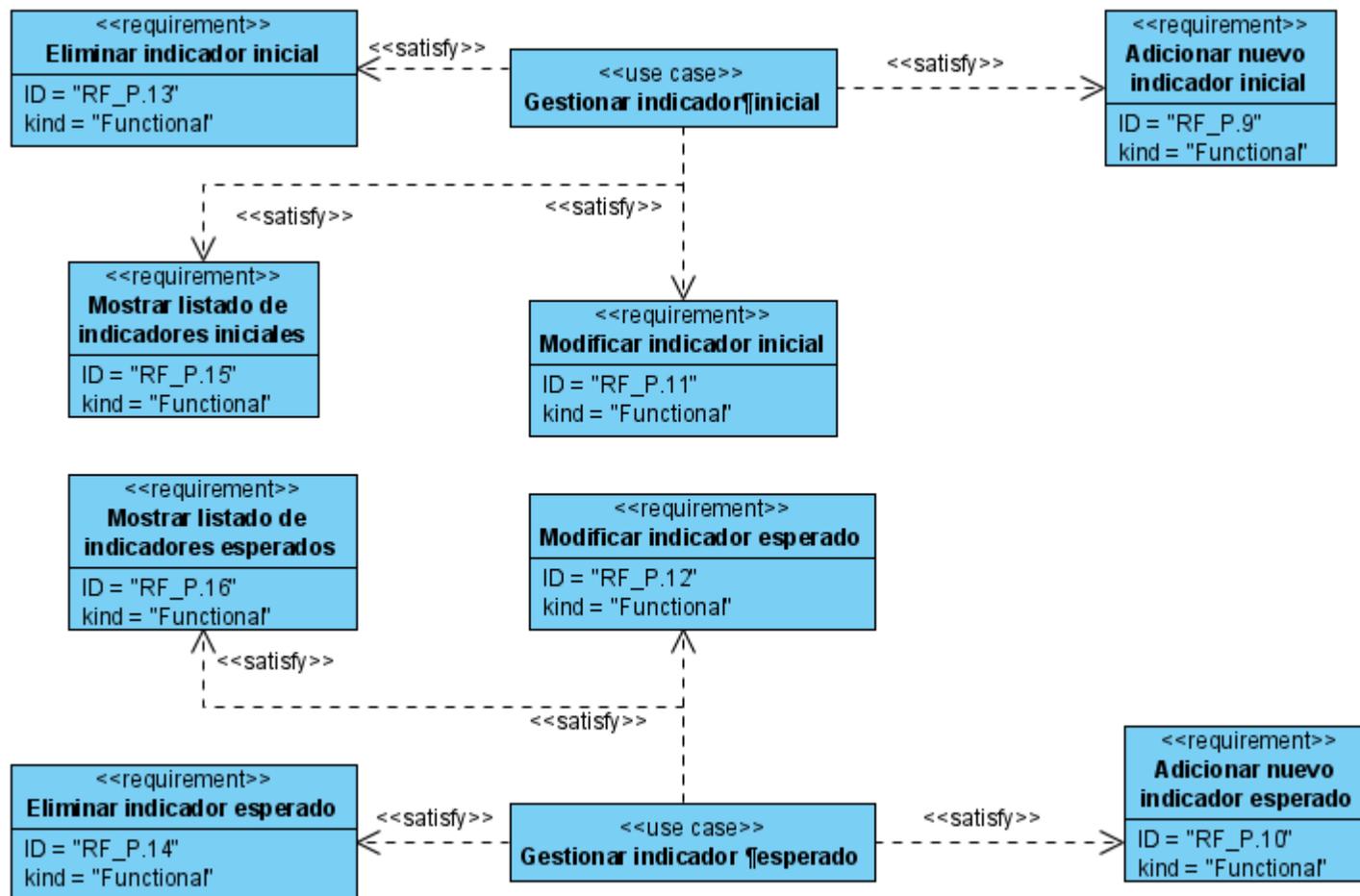


Figura 24 Diagrama de requerimientos de los casos de uso Gestionar indicador inicial y Gestionar indicador esperado

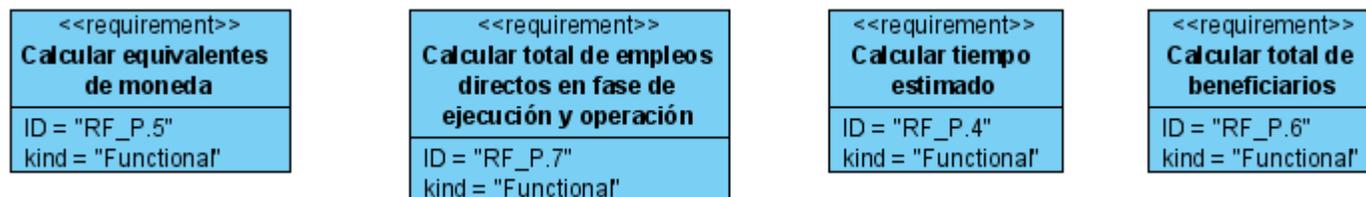


Figura 26 Paquete Requerimientos auxiliares

Anexo G:

Documento Modelo de Sistema v1.0 del módulo Registro y Aprobación de Proyectos del sistema SINAPSIS

Anexo H:

Estructura del documento
1. ¿Está el documento acorde con a la plantilla estándar del proyecto o del expediente de proyecto?
2. ¿Contiene las secciones obligatorias definidas en el expediente? (Ver Expediente de Proyecto)
Aspectos Generales
1. ¿Cada caso de uso registra claramente lo que el sistema debe hacer?
2. ¿Están clasificado los casos de uso que definen la arquitectura básica del sistema?
3. ¿Se ha identificado los casos de uso que darán soporte y mantenimiento al sistema?
4. ¿Se ha descrito con precisión todas las alternativas o excepciones?
5. ¿Están clasificados los casos de uso que sirven de apoyo a los caso de uso que cubren las principales funciones que el sistema debe realizar?
6. ¿Están clasificados los casos de uso que no son claves para la arquitectura?
Nombre del Caso de Uso
1. ¿Está en infinitivo y refleja de manera clara el objetivo del usuario sobre el sistema?
2. ¿El nombre del caso de uso es único?
3. ¿El nombre del caso de uso es intuitivo?
Descripción
1. ¿El resumen dice como se inicia, como termina y las operaciones principales que realiza el caso de uso?
Precondición
1. ¿Se escribe una precondición si y solo si a partir de la ocurrencia de un suceso determinado comienza el caso de uso?
2. ¿La precondición es válida tanto para flujos básicos como flujos alternativos?
Poscondición
1. ¿La poscondición plasma cambios que suceden en el sistema al terminarse de ejecutar el caso de uso?
Complejidad del CU
1. ¿Se especifica la complejidad del caso de uso?
Forma de presentar la información
1. ¿Está descrito el caso de uso en presente?
2. ¿Se describe de manera comprensible y detallada las acciones del actor frente al sistema? ¿Está lo más parecido a un manual de ayuda?

Actores del CU
1. ¿El Caso de Uso está relacionado con al menos un actor?
2. ¿Si hay dos actores interactuando con el caso de uso está generalizado en uno solo?
3. ¿Si el caso de uso es abstracto (incluye, extiende, generalización-especialización), no lo inicializa ningún actor?
Flujo Básico
1. ¿Comienza diciendo “El caso de uso se inicia cuando el actor...”?
2. ¿Termina diciendo en un evento independiente “El caso de uso termina”?
3. ¿No existen abreviaturas?
4. ¿Las partes del flujo de eventos que se repiten en otro caso de uso se especifican como un Caso de Uso incluido?
5. ¿Si las alternativas que se describen casi nunca ocurren o son alternativas comunes a otros casos de uso se especifican como un Caso de Uso extendido?
6. ¿Si existe un proceso general y a partir de él se especializan otros se especifican como una generalización/especialización?
Flujo Alterno
1. ¿Las alternativas o excepciones se reflejan como flujos alternos?
2. ¿En todos los CU que se introducen datos tienen un flujo alternativo donde el sistema valida la integridad de los datos que se introducen y muestra un mensaje en caso de que los datos estén incompletos?
3. ¿Los flujos alternativos se nombran con el número del paso que lo generó en el flujo básico, una letra, ordenados alfabéticamente que lo produjo?
4. ¿En la sección flujos alternativos se describen todas las excepciones que existan por muy evidentes que parezcan?
Casos de uso incluidos y extendidos
1. ¿Al describir el caso de uso base se mencionan todos los casos de Uso que Extienden, se incluyen o se generalizan del Caso de Uso?
2. ¿La descripción de los Casos de Uso incluidos, extendidos y especializados se realiza aparte?
Navegabilidad
1. ¿La navegabilidad en los caso de uso de inclusión se inicia desde el caso uso base hasta el caso de uso incluido?
2. ¿La navegabilidad en los caso de uso de extensión se inicia desde el caso uso extendido hasta el caso de uso base?
3. ¿La navegabilidad en la generalización/especialización se inicia desde el caso de uso especializado a al generalizado y se representa con una relación de herencia?
Relaciones
1. ¿Las relaciones de inclusión y extensión entre los caso de uso se han representado con línea discontinua?
Información General
1. ¿El diagrama de casos de uso expresa en detalles y claramente lo que debe hacer el

sistema?
2. Si la modelación de las interacciones con el sistema es muy extensa ¿ha empleado los paquetes de caso de uso?
Semántica del documento
1. ¿Ha identificado errores ortográficos?
2. ¿Se entiende claramente lo que se ha especificado en el documento?
3. ¿El número de página que aparece en el índice coincide con el contenido que se refleja realmente en dicha página?
4. ¿El total de páginas que aparecen en las reglas de confidencialidad coincide con el total de páginas que tiene el documento?

Anexo I:

Nota: Este anexo corresponde a la certificación de calidad UCI

Anexo J:

Req. 1	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1		1		1
	2	1			1
	3				2
	4				3
	5				1
Req. 4	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1		2	2
	2				1
	3				
	4				1
	5			1	
Req. 7	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1	1	2	1
	2				1
	3	1			
	4		1		
	5	1			
Req. 10	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1		1	1	2
	2				
	3				
	4		1		
	5	1		2	
Req. 13	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1	2		1
	2				3
	3			1	1
	4				
	5				

Req. 2	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1		2		4
	2				
	3			1	
	4				1
	5	1			
Req. 5	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	2	1		4
	2				
	3				
	4				
	5		1	1	
Req. 8	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1	1	2	2
	2				
	3				
	4	1			1
	5				
Req. 11	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1	2	1	2
	2				1
	3				1
	4				
	5	1			1
Req. 14	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1		2		2
	2				3
	3			1	1
	4				1
	5				

Req. 3	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	2	1	1	3
	2			1	
	3				1
	4				
	5		1		
Req. 6	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	2	1	2	1
	2				
	3				
	4				2
	5	1			
Req. 9	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1		1	4
	2				
	3			1	1
	4				
	5		2		
Req. 12	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1	1		1	2
	2				2
	3				
	4		1		1
	5	1			1
Req. 15	Req. Disfuncionales				
	1	2	3	4	5
Req. Funcionales	1			3	1
	2			1	
	3			1	2
	4		1		1
	5				

GLOSARIO

AMDD: Acrónimo del inglés *Ágile Model Driven Development* (Desarrollo Guiado por Modelado Ágil). Es una técnica que consiste en la creación de modelos ágiles para guiar los esfuerzos del equipo de desarrollo antes de codificar, pudiéndose alternar entre modelado y codificación cuando sea necesario.

CASE: Acrónimo de *Computer Aided Software Engineering* (Ingeniería de Software Asistida por Ordenador), son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

Clipboard: Portapapeles. Simple herramienta del Sistema Operativo que permite almacenar temporalmente información de cualquier tipo.

ERD: Acrónimo de *Entity Relationship Diagram* (Diagrama de Entidad Relación), es una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades.

ERWin: Herramienta para el diseño de base de datos, que brinda productividad en su diseño, generación, y mantenimiento de aplicaciones. Desde un modelo lógico de los requerimientos de información, hasta el modelo físico perfeccionado para las características específicas de la base de datos diseñada. Genera automáticamente las tablas.

IBM: *International Business Machines* (conocida coloquialmente como el Gigante Azul) es una empresa que fabrica y comercializa herramientas, programas y servicios relacionados con la informática.

IEEE: Corresponde a las siglas de *The Institute of Electrical and Electronics Engineers* (el Instituto de Ingenieros Eléctricos y Electrónicos), es la asociación técnica y profesional, sin fines de lucro, más grande del mundo formada por profesionales de todas las disciplinas de la ingeniería. Su trabajo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales.

NASA: Es la sigla, en inglés, para la Administración Nacional de Aeronáutica y del Espacio (inglés: National Aeronautics and Space Administration) de los Estados Unidos, que es la agencia gubernamental responsable de los programas espaciales.

OMG: Object Management Group es una asociación sin fines de lucro formada por grandes corporaciones, muchas de ellas de la industria del software, como IBM, Apple, Sun Microsystems y HP. Se encarga de la definición y el mantenimiento de estándares para aplicaciones de la industria de la computación, como UML, CORBA y otros.

PDF: acrónimo del inglés Portable Document Format (formato de documento portátil), es un formato de almacenamiento de documentos, de tipo compuesto (imagen vectorial, mapa de bits y texto). Está especialmente ideado para documentos susceptibles de ser impresos, ya que especifica toda la información necesaria para la presentación final del documento, determinando todos los detalles de cómo va a quedar, no requiriéndose procesos anteriores de ajuste ni de maquetación.

Plug-in: Un complemento (o plug-in en inglés) es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de la API.

Programación en pares (pair-programming): Es una técnica de desarrollo de software, en la cual dos programadores trabajar juntos en la misma máquina. Uno de ellos escribe el código mientras el otro lo revisa, mejorando la productividad y el diseño. Pasado un tiempo intercambian los roles.

Programación Orientada a Objetos (POO): Es un paradigma de programación que define los programas en términos de “clases de objetos”, objetos que son entidades que combinan estado (propiedades o datos), comportamiento (procedimientos o métodos) e identidad (propiedad del objeto que lo diferencia del resto). Expresa un programa como un conjunto de estos objetos, que colaboran entre ellos para realizar tareas. Esto permite hacer los programas y módulos más fáciles de escribir, mantener y reutilizar.

Pruebas de aceptación: Pruebas realizadas por el cliente para validar y verificar que se han cumplido todos los requerimientos. Estas pruebas las realiza el usuario final una vez liberado el producto software.

Pruebas de unidad: Pruebas realizadas para determinar el correcto funcionamiento de un módulo de código. Son usadas para asegurar que cada uno de los módulos funcione correctamente por separado.

Refactorización (Refactoring): Tipo de reestructuración de código que se aplica en desarrollos orientados a objetos. Tiene como objetivo limpiar el código para lograr una mejora en el diseño del software y ayudar a encontrar errores ocultos.

Release: Producto final preparado para lanzarse como versión definitiva a menos que aparezcan errores que lo impidan. Implementa todas las funciones del diseño y se encuentra libre de cualquier error que suponga un punto muerto en el desarrollo.

Stakeholders: Son los interesados, todas aquellas personas u organizaciones que afectan o son afectadas por el proyecto, ya sea de forma positiva o negativa.

TDD: Acrónimo de Desarrollo guiado por pruebas, (en inglés Test-driven development). Es una práctica de programación que involucra otras dos prácticas: Escribir las pruebas primero y Refactorización. Para escribir las pruebas generalmente se utilizan las pruebas unitarias.

UML: Acrónimo del inglés Unified Modeling Language (Lenguaje Unificado de Modelado). Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

Walkthrough: Técnica de análisis estático en la que un programador o diseñador dirige a miembros del equipo de desarrollo u otras personas interesadas a través de un segmento de documentación o código y los participantes realizan comentarios sobre posibles errores, violaciones de estándares de desarrollo y otros problemas.

XMI: XML Metadata Interchange (XML de Intercambio de Metadatos) es una especificación para el Intercambio de Diagramas, escrita para proveer una manera de compartir modelos UML entre diferentes herramientas de modelado.

XML: Acrónimo de Extensible Markup Language («lenguaje de marcas ampliable»), es un metalenguaje extensible de etiquetas, aunque no es

realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

XPS: Acrónimo de XML Paper Specification. Es un formato de archivo electrónico que conserva el formato del documento y permite compartir archivos. El formato XPS garantiza que, cuando el archivo se vea en línea o impreso, se conservará exactamente el formato original y que los datos del archivo no se puedan copiar o cambiar con facilidad.