

Universidad de las Ciencias Informáticas
Facultad 3



**Título: Análisis del módulo Tesorería en el Sistema de
Administración Financiera de SAREN.**

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

Autor: Ana Cecilia Labrador Valdés

Tutor(es): Ing. Dayana La Rosa Montes
Ing. Yaumarys Pino Cueto

La Habana, Cuba
Junio 2008

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Ana Cecilia Labrador Valdés

Firma del Autor

Ing. Dayana La Rosa Montes

Firma del Tutor

Ing. Yaumarys Pino Cueto

Firma del Tutor

AGRADECIMIENTOS

A Dayana y Yaumarys tutoras y amigas por ayudarme y guiarme en la realización de este trabajo sin las cuales este no hubiera sido posible, gracias por todo el tiempo que me han dedicado.

A mis compañeros del proyecto RN especialmente a Yinett y Dailen por ser mis compañeras de tesis prácticamente y ayudarme y aclararme dudas en el momento que las necesitase.

A la Revolución Cubana por haberme dado la posibilidad de estudiar una carrera como esta y especialmente a la UCI por enseñarme todo lo que se necesita para ser un mejor profesional.

A mis amigos (as) de mi antiguo grupo 4507 conocerlos fue lo mejor que me ha pasado en especial a Annelis, Dailys, Sharito, Eivys y Hermes porque junto a ustedes he pasado los mejores años de mi vida.

A Yaniris por ser mi mejor amiga y segunda hermana.

A toda mi familia especialmente a mi mamá, mi papá, mis abuelos y mi hermana por todo el apoyo que me han dado durante estos años.

A todos aquellos que se preocuparon por mi tesis en algún momento.

DEDICATORIA

A mis padres por su amor incondicional, por apoyarme y aconsejarme en todo momento, gracias por estar siempre presente.

A mis abuelos maternos por ser mis segundos padres, gracias por todo el amor y la educación que me han dado.

A mis abuelos paternos porque sé que les gustaría estar presentes en este momento.

A mi hermana por ser mi mejor amiga.

A toda mi familia porque de una forma u otra han contribuido a mi formación.

RESUMEN

El Ministerio del Poder Popular para las Relaciones Interiores y Justicia siguiendo la línea trazada por el gobierno bolivariano decidió continuar y profundizar un conjunto de transformaciones ya iniciadas, creando proyectos como Servicio Autónomo de Registros y Notarías (SAREN), el cual tiene como base automatizar todo el flujo de procesos que se llevan a cabo en los Registros y Notarías (RN). El mismo está compuesto por el sistema de Administración Financiera, conformado por 11 módulos entre los que se encuentra Tesorería.

Este módulo debe ser capaz de gestionar una Tesorería única y centralizada, que entre sus funciones principales permita llevar el control y registro de las operaciones bancarias que se realizan en los RN sobre las cuentas bancarias que se crean en los mismos, además darle respuesta a las órdenes de pago solicitadas por estos debido a la necesidad de adquirir bienes o servicios.

Para el desarrollo de este módulo fue necesario realizar un estudio de los procesos que se llevan cabo en la Tesorería que se desarrolla en los RN, además de la metodología de desarrollo, lenguaje de modelado y herramientas utilizadas para su elaboración. A partir de esto se logró determinar los requisitos funcionales necesarios para la informatización de este módulo, logrando con esta solución un mejor aprovechamiento y control de los recursos públicos, erradicando de esta forma el mal manejo que había de los mismos.

PALABRAS CLAVES

Proceso de Tesorería, Ingeniería de Requisitos, Metodologías de Desarrollo, Lenguajes de Modelado, Herramientas CASE, Requerimientos, Prototipo de Interfaz de Usuario.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1. PROCESO DE TESORERÍA	5
1.1.1. <i>Importancia de la Tesorería</i>	6
1.1.2. <i>Necesidad de desarrollar software para informatizar el proceso de Tesorería.</i>	6
1.1.3. <i>Algunas soluciones existentes relacionadas con el Proceso de Tesorería</i>	7
1.2. INGENIERÍA DE REQUISITOS	9
1.2.1. <i>¿Qué es la Ingeniería de Requisitos?</i>	9
1.2.2. <i>Importancia de la IR.</i>	9
1.2.3. <i>Requisitos de Software</i>	10
1.2.4. <i>Actividades de la Ingeniería de Requisitos</i>	11
1.2.5. <i>Técnicas para llevar a cabo la Ingeniería de Requisitos</i>	14
1.3. TENDENCIAS ACTUALES SOBRE METODOLOGÍAS, HERRAMIENTAS CASE Y LENGUAJE DE MODELADO PARA EL DESARROLLO DE SOFTWARE. 18	
1.3.1. <i>Procesos de Desarrollo de Software</i>	18
1.3.2. <i>Lenguaje de Modelado</i>	24
1.3.3. <i>Herramientas de Modelado</i>	26
1.3.4. <i>Herramientas CASE</i>	26
CAPÍTULO 2: MODELACIÓN DEL NEGOCIO	33
2.1. ¿EN QUE CONSISTE LA MODELACIÓN DEL NEGOCIO?	33
2.1.1. <i>Objetivos del Modelado del Negocio</i>	34
2.2. MODELADO DEL NEGOCIO DE TESORERÍA	35
2.2.1. <i>Alcance</i>	35
2.2.2. <i>Descripción del Negocio</i>	35
2.2.3. <i>Actores y Trabajadores del Negocio</i>	36
2.2.4. <i>Casos de Uso del Negocio</i>	38
2.2.5. <i>Modelo de Casos de Uso del Negocio</i>	40
2.2.6. <i>Realización de los Casos de Uso del Negocio</i>	42
2.2.7. <i>Modelo de Clases de Análisis del Negocio</i>	55
2.2.8. <i>Reglas del Negocio</i>	60
CAPÍTULO 3: REQUERIMIENTOS.....	61
3.1. FLUJO DE TRABAJO DE REQUERIMIENTOS	61
3.1.1. <i>Objetivos del Flujo de Trabajo Requerimientos</i>	62
3.1.2. <i>Requerimientos Funcionales y No Funcionales</i>	63
3.1.3. <i>Actores del Sistema</i>	70
3.1.4. <i>Patrones de Casos de Uso</i>	70
3.1.5. <i>Casos de Uso del Sistema</i>	72
3.1.6. <i>Modelo de Casos de Uso del Sistema</i>	75
3.1.7. <i>Realización de los Casos de Uso del Sistema</i>	79
3.2. VALIDACIÓN DE LOS RESULTADOS OBTENIDOS.....	105
3.2.1. <i>Validación Mediante Prototipo</i>	105
3.2.2. <i>Aplicación de Métricas</i>	105
CONCLUSIONES	112
RECOMENDACIONES.....	113
REFERENCIAS.....	114
BIBLIOGRAFÍA.....	116
GLOSARIO.....	117

INTRODUCCIÓN

El Ministerio del Poder Popular para las Relaciones Interiores y Justicia (MPPRIJ), siguiendo la línea trazada por la Revolución Bolivariana de provocar un cambio en la gestión de los ministerios públicos, ha realizado un conjunto de transformaciones en la legislación como es la promulgación de la Ley de Registro Público, favoreciendo la legalidad y transparencia de todos los trámites que se realizan en los mismos.

El MPPRIJ no dispone de herramientas tecnológicas que garanticen la eficiencia y calidad en la ejecución de las funciones que se llevan a cabo en los Registros y Notarías. Debido a esto la información que se obtiene en SAREN¹, dirección general de los mismos no es la más confiable, trayendo como consecuencia que no se pueda obtener un control sobre las transacciones realizadas y los ingresos prestados. En el 90% de las oficinas de Registros y Notarías, la infraestructura es inapropiada y la plataforma tecnológica no garantiza el control efectivo y eficiente de la información. (Pérez, 2005)

El conjunto de servicios que se prestan en las oficinas registrales tienen asociados diferentes precios directamente relacionados a los impuestos por los recaudos que se deben presentar según estipula la ley. Es de ese caudal de donde también se nutre el estado para desarrollar los numerosos planes sociales que favorecen a las grandes masas de personas necesitadas. Cuanto más control y transparencia se logre aplicar al proceso mejor administrado estará siendo el dinero del pueblo. Con la informatización del módulo de Tesorería se evidenciarán grandes cambios en las funciones que se llevan a cabo en los Registros y Notarías y los servicios que en ellos se prestan, esta solución contribuirá a erradicar la corrupción que existe actualmente con el manejo de los recursos públicos del MPPRIJ.

Este proceso de automatización ha sido contratado a la empresa cubana ALBET, Ingeniería y Sistemas S.A., que constituye la representación de la Universidad de las Ciencias Informáticas (UCI), dentro del marco de cooperación del proyecto de integración latinoamericana ALBA. En consecuencia se trabaja en el Proyecto de Modernización de los Registros y Notarías de la República Bolivariana de Venezuela creado con el objetivo de automatizar todos los procesos que se llevan a cabo en los Registros y Notarías, en el mismo se encuentra el Sistema de Administración Financiera encargado de

¹ SAREN: Servicio Autónomo de Registros y Notarías.

informatizar toda el área financiera, este a su vez cuenta con 11 módulos, entre ellos está el de Tesorería siendo este el objetivo de análisis del presente trabajo.

El MPPRIJ y en especial SAREN presentan la necesidad de gestionar una Tesorería única y centralizada, que permita llevar el control y registro de las operaciones bancarias, que se realizan en las Unidades Ejecutoras Locales (UEL) sobre la cuenta que se crea en la Unidad Administradora Desconcentrada (UAD) a las que estas pertenecen y darle respuesta a las órdenes de pago solicitadas por las UAD a petición de las UEL que esta administra, por lo que es necesario desarrollar un software que permita automatizar dichos procesos, el cual debe cumplir con las reglas del negocio y las necesidades especificadas por los clientes y usuarios finales. Para obtener esto se debe lograr un entendimiento común entre clientes y el equipo de desarrollo sobre los requisitos que debe cumplir el sistema a desarrollar.

Con el presente trabajo se pretende dar solución a la situación problemática expuesta anteriormente, donde se extrae como **problema científico** ¿Cuáles serían las necesidades automatizables dentro del módulo de Tesorería a partir de los procesos definidos por SAREN? Siendo el **objeto de estudio** la Ingeniería de Requisitos y teniendo como **campo de acción** el análisis, especificación y validación de los requisitos del módulo de Tesorería para el registro y control de las operaciones bancarias en los Registros y Notarías de la República Bolivariana de Venezuela.

Para darle solución a la problemática planteada anteriormente, se define el siguiente **objetivo general**: Realizar la modelación del negocio y levantamiento de requisitos del módulo de Tesorería del sistema de Administración Financiera de SAREN.

Derivándose los siguientes **objetivos específicos**:

- Construir el Modelo de Casos de Uso del Negocio.
- Identificar las actividades automatizables en cada uno de los subprocesos descritos.
- Identificar los requerimientos funcionales y no funcionales del sistema de Tesorería.
- Construir el Modelo de Casos de Uso del Sistema.
- Realizar un prototipo de interfaz de usuario para validar los requisitos identificados.
- Aplicar un conjunto de métricas para validar los requisitos del sistema identificados.

A continuación se exponen un conjunto de **tareas de la investigación** para dar cumplimiento al objetivo trazado:

- Realización de talleres con los especialistas funcionales de la organización con el objetivo de comprender el negocio.
- Estudio de la documentación de los procesos de la organización.
- Identificación de los procesos fundamentales que se llevan a cabo en la Tesorería que se desarrolla en los Registros y Notarías de la República Bolivariana de Venezuela.
- Investigación de soluciones informáticas que automaticen el proceso de Tesorería para determinar si existe alguno que sirva de apoyo al sistema a desarrollar.
- Estudio de metodologías de desarrollo, herramientas CASE y lenguajes de modelado para el desarrollo de software.

A partir de lo expuesto anteriormente se plantea la siguiente **hipótesis**:

Si se realiza la modelación del negocio y el levantamiento de requisitos del módulo de Tesorería, entonces se logrará identificar las necesidades automatizables a partir de los procesos definidos por SAREN.

Para dar cumplimiento a las tareas propuestas anteriormente se van a utilizar los métodos científicos de la investigación: **teórico** y el **empírico**.

Dentro de los métodos de investigación teóricos se emplearán:

- Método Analítico-Sintético: Permite analizar, estudiar en partes e interpretar la teoría con el fin de extraer los elementos más importantes que se relacionan con el objeto.
- Método Inductivo-Deductivo: Permite a través de un razonamiento llegar a un grupo de conocimientos particulares y generales.
- Método Histórico: Analiza la trayectoria completa del fenómeno y revela las etapas principales de su desenvolvimiento y las conexiones históricas fundamentales.
- Método de la Modelación: Es el método mediante el cual se crean abstracciones con el objetivo de explicar la realidad.

Dentro de los Métodos de investigación empírico se emplearán:

- Método de la Observación: Posibilita percibir el desarrollo de la investigación en distintos momentos de la misma, conociendo el problema y el objeto de investigación. (Coello González, y otros, 2002).

A continuación se estructura el trabajo investigativo realizado, el cual se dividió en tres capítulos y varios anexos afines de representar los diagramas y descripciones generados durante las diferentes fases.

Capítulo 1: Fundamentación Teórica, en él se abordará sobre el proceso de Tesorería, su importancia y la necesidad de informatizar procesos de este tipo, además se hará un análisis de algunos software que han automatizado este proceso, de los cuales se extraerán las funcionalidades más importantes, las que complementarán el sistema a desarrollar. También se indagará sobre la Ingeniería de Requisitos, su importancia en el desarrollo de software y las técnicas que se utilizan para su aplicación. Además se realizará un estudio sobre metodologías de desarrollo, herramientas CASE y lenguajes de modelado para utilizar en la realización de este trabajo.

Capítulo 2: Modelación del Negocio, este estará enfocado a la solución del problema en cuestión, en el mismo se realizará la modelación del negocio del módulo de Tesorería donde se hará un estudio de dicho proceso, es decir, como este se lleva a cabo en los Registros y Notarías, a partir de esta se obtendrán artefactos como el modelo de casos de uso del negocio, modelo de objetos, reglas del negocio, glosario de términos del negocio además de la realización de casos de usos en la cual a partir de los diagramas de actividades se determinará las actividades que serán automatizadas.

Capítulo 3: Requerimientos, a partir de los resultados obtenidos en el capítulo anterior y la aplicación de la Ingeniería de Requisitos, se identificarán los requisitos de software, los cuales serán analizados, especificados para conformar posteriormente los casos de uso del sistema como resultado final, estos serán validados a través de la técnica de prototipos y la aplicación de métricas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se abordan diferentes temas entre los que se encuentran el proceso de Tesorería, su importancia y la necesidad de informatizar dicho proceso. Se realiza un estudio de algunas soluciones que informatizan este proceso, de las cuales se tomaron las funcionalidades más importantes para complementar el software a desarrollar. Además de realizar un estudio sobre metodologías de desarrollo, herramientas CASE y lenguajes de modelado a utilizar en la realización de este trabajo.

1.1. Proceso de Tesorería

Los flujos de efectivos actualmente constituyen una herramienta indispensable para la toma de decisiones en la empresa moderna. Rara es la decisión empresarial que no requiera conocer cómo anda la situación financiera de la empresa, es decir si la misma cuenta con dinero líquido para hacerle frente a determinada obligación. Cuando se habla de flujos de efectivo, se refiere a las corrientes de efectivo que se producen, motivado por los movimientos de entradas y salidas de dinero, es decir las variaciones que han ocurrido en la Tesorería en un determinado período de tiempo.

Al referirse a la Tesorería, se están incluyendo todas las cuentas de efectivo con que cuenta la empresa, incluso las inversiones financieras a corto plazo, que pueden ser convertibles en dinero de forma inmediata. (Menéndez, y otros)

A partir de los nuevos cambios y reformas realizadas en el MPPRIJ, se crea una nueva estructura financiera en SAREN, órgano rector de los mismos donde los Registros y Notarías pasan a ser Unidades Ejecutoras Locales (UEL), aquellos que por sus condiciones físicas tengan la capacidad de administrar a otros Registros y Notarías serán Unidades Administradoras Desconcentradas (UAD), las cuales a su vez serán administradas por la Unidad Administradora Central (UAC), que no es más que la dirección de SAREN. A partir de esta estructura, el proceso de Tesorería se va a desarrollar en la UAC, el cual va a controlar las cuentas bancarias que se crean en cada una de las Unidades Administradoras (UA) y en dichas cuentas se registrarán las operaciones bancarias que se llevan a cabo en las UEL adjuntas a las UA, quedando este flujo de entrada y salida de efectivo registrado en el libro auxiliar de la unidad, permitiendo este ser comparado posteriormente con los estados de cuentas proporcionados por el banco, realizando de esta forma el proceso de conciliación bancaria, lo que da la posibilidad de detectar si hubo errores al realizar las transacciones.

1.1.1. Importancia de la Tesorería

Benjamin Tripier en su artículo “La importancia de la Tesorería” plantea que esta es una de las funciones críticas dentro del área financiera, pues debe administrar el flujo de caja con una cierta tasa de rentabilidad y oportunidad para ese mercado. Ese flujo es muy propio de cada industria, de cada tipo de negocio, y de cada empresa en particular, con sus productos y servicios.

La gestión de la tesorería está asociada con ritmos y velocidades de cobros y pagos, la mayoría de los cuales pasarán por los bancos. Entre los actores internos y externos que se mueven alrededor de la tesorería están clientes, proveedores y bancos, estos últimos tienen la característica de ser la plataforma sobre la cual se efectúan las transacciones. Gran parte del éxito que pueda tener una tesorería, estará dado por la capacidad de ir un paso adelante de los bancos, llevando un control proactivo de las operaciones que permita anticipar con información precisa y consistente, los movimientos que lleva el banco. La estrategia de manejo del banco debe ser capaz de mantener una conciliación de saldos permanentemente apoyada en tecnología de información, que explique las discrepancias y sea parte del ciclo de negocio. (Tripier, 2005)

1.1.2. Necesidad de desarrollar software para informatizar el proceso de Tesorería.

Al ser un tipo de operación basada en información, la tecnología debe acompañar a los procesos y reflejarse en la organización. Una operación de tesorería, es principalmente una operación sin papeles, la cual podría virtualizarse totalmente bajo un diseño agresivo e innovador con altos niveles de delegación, y con fortaleza de control en los sistemas de información. Debe apoyarse en información actualizada y confiable, la cual se soporte en una estructura de datos que identifique a la empresa desde adentro, y a la industria y al mercado desde afuera. Pues solo de esa manera se podrá confiar en que la estructura de la cartera de inversiones y de endeudamiento es lo que el sistema dice que son. De esa misma forma debe poder confiar en las cifras de cuentas por pagar y cobrar, así como en las de los bancos.

Se podría establecer que si una tesorería, en Venezuela, no ha sido revisada, reestructurada, refrescada en los últimos dos años, seguramente necesitará de cambios y actualizaciones para poder cumplir con el rol relevante que estos tiempos le demandan. Tanto en la situación presente de Venezuela, como en la nueva economía hacia la que (inevitablemente) se dirige hoy la sociedad, será importante, como instrumento estratégico, contar con una Tesorería capaz de acompañar y habilitar procesos de transformación en los cuales el componente de oportunidad financiera podrá significar la diferencia entre el éxito y el fracaso. (Tripier, 2005)

1.1.3. Algunas soluciones existentes relacionadas con el Proceso de Tesorería

Se mencionan a continuación algunas de las soluciones informáticas existentes en relación al proceso de Tesorería, debido a las funcionalidades que éstas brindan como son: administración de cuentas bancarias, registro y control de operaciones bancarias, reportes sobre estados financieros, generación automática de comprobantes de las operaciones entre otras, las que se tomaron en cuenta para el desarrollo de este sistema y permitieron enfocar el software a desarrollar directamente al proceso de Tesorería específico de Venezuela.

Sistemas Integrados de Gestión

Al hablar sobre software de gestión no cabe dudas el importante papel que juegan los Sistemas Integrados de Gestión como la expresión más acabada de sistemas de información orientados al registro de las operaciones económicas en su sentido más amplio. Son el producto lógico del desarrollo alcanzado en el uso de nuevas tecnologías y la experiencia acumulada en la explotación de los sistemas automatizados de contabilidad de mayor o menor envergadura, así como de otras aplicaciones informáticas asociadas a la gestión económica y financiera.

Dentro de estos sistemas se pueden mencionar los ERP (Enterprise Resourcing Planning) traducido al español Planificación de Recursos de la Empresa, estos sistemas se caracterizan por gozar de un alto nivel de integración entre sus módulos y además porque abarcan la mayoría de los procesos de negocios de las diferentes organizaciones. Entre los módulos que implementan se pueden encontrar: el de Gestión Contable y Financiera capaz de recoger y contabilizar de forma automática todas las transacciones como resultado de las operaciones que se generan en los diferentes módulos entre ellos el de Tesorería con el cual se vincula estrechamente, así como aquellas que no provengan de ninguno de ellos y se introduzcan de manera directa, ofreciendo los Estados Financieros y otros reportes con los indicadores preseleccionados. (Terry, y otros, 2006)

Sistema Informativo Bancario (SIB)

El SIB se creó debido a la necesidad de las unidades organizativas del Banco Central de Cuba de contar con distintas informaciones para el cumplimiento de sus funciones y que, a su vez, partiera de un sistema informativo centralizado, eficiente y racional de captación, almacenamiento, procesamiento y distribución de la información.

El SIB consta de aproximadamente 30 modelos, en cuya creación han estado involucradas las distintas áreas del banco y la Dirección de Informática y Computación. El sistema abarca informaciones correspondientes a las áreas de Política Monetaria, Operaciones, Tesorería, Supervisión Bancaria, Balanza de Pagos, Deuda Externa y se nutre de informaciones procedentes del sistema financiero, tanto de instituciones bancarias como no bancarias, de los Organismos de la Administración Central del Estado y de empresas que reportan de forma independiente.

La implantación del SIB ha recorrido distintas etapas, en las que se ha interactuado con las entidades tributantes de forma sistemática, a fin de erradicar los constantes errores en las informaciones entregadas, entre otras cosas, debido al desconocimiento que sobre el uso de bases de datos existía en numerosas entidades. (Banco Central de Cuba, 2007)

Sistema Automatizado para el Control de la Tesorería

El Ministerio de Salud Pública es priorizado por el Estado cubano, lo cual implica que sea altamente complejo manejar los recursos financieros que le son asignados, tanto en moneda nacional como en divisas; de ahí que la automatización de este proceso sea vital para el organismo. A propuesta del Vice-ministerio de Economía del Ministerio de Salud Pública se solicitó al Centro de Cálculo Provincial de Salud Pública de Santiago de Cuba, a través de la Vice-dirección de Economía de la Dirección Provincial de Salud, el desarrollo e implementación de una nueva versión del sistema automatizado que controla el Departamento de Tesorería de dicho Vice-ministerio y de las demás instituciones que así lo ameriten. Este permite el control de los aspectos relacionados con el Departamento de Planificación, además de ganar en la calidad de las informaciones que se brindan, así como facilitar la labor, permitiendo trabajar con varias cuentas bancarias e incluir nuevos reportes generados por el usuario. Otro aspecto que mejora es la seguridad de la aplicación, así como la generación automática de los comprobantes de operaciones. (Rojas, y otros, 2003)

A continuación se explicarán algunos aspectos de la Ingeniería de Requisitos, la cual ayuda a identificar de manera más clara y concisa los requisitos de software.

1.2. Ingeniería de Requisitos

El desafío de la ingeniería de sistemas y de los ingenieros del software, se plantea de forma general a través de la siguiente interrogante: ¿Cómo asegurar que se ha especificado un sistema que recoge las necesidades del cliente y satisface sus expectativas? No hay una respuesta segura a esta difícil pregunta, pero un sólido proceso de Ingeniería de Requisitos (IR) es la mejor solución de que se dispone actualmente. (Pressman, 2002)

1.2.1. ¿Qué es la Ingeniería de Requisitos?

La ingeniería de sistemas demanda una intensa comunicación entre el cliente y el ingeniero del sistema. Esto se realiza a través de un conjunto de actividades bajo la denominación de IR.

La IR del software es un proceso de descubrimiento, refinamiento, modelado y especificación.

Donald Reifer describe el proceso de ingeniería de requisitos del software de la siguiente forma:

La IR es el uso sistemático de procedimientos, técnicas, lenguajes y herramientas para obtener con un coste reducido el análisis, documentación, evolución continua de las necesidades del usuario y la especificación del comportamiento externo de un sistema que satisfaga las necesidades del usuario. (Pressman, 2002)

1.2.2. Importancia de la IR

La IR es un tema complejo y crucial para el éxito de todo el proyecto de Ingeniería del Software. Esta facilita el mecanismo apropiado para comprender lo que quiere el cliente, analizando necesidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedad, validando la especificación y gestionando los requisitos para que se transformen en un sistema operacional. (Pressman, 2002)

Esta tiene como objetivo la determinación de las necesidades del sistema, la adquisición por parte del equipo de desarrollo de la información necesaria para desarrollar un producto de calidad, permitiendo de esta forma la comunicación inicial entre el grupo de expertos y los clientes y usuarios. (Escalona, y otros)

1.2.3. Requisitos de Software

¿Qué es un requisito?

Lo más difícil en la construcción de un sistema de software es decidir exactamente qué construir. Ninguna otra parte del trabajo conceptual es tan difícil como establecer los requerimientos técnicos detallados, incluyendo todas las interfaces con las personas, las máquinas y otros sistemas de software. Ninguna otra parte deshabilita tanto al sistema resultante si se hace mal. Ninguna otra parte es más difícil de corregir después. (Brooks, 1987) Esto es precisamente a lo que se denomina requisito de software, aquella parte del sistema que se desea construir, a continuación se muestran algunas definiciones del mismo.

La IEEE 1233-1998 define requisito como:

- I. Una condición o capacidad que necesita el usuario para resolver algún problema o alcanzar un objetivo.
- II. Condición o capacidad que debe cumplir o poseer un sistema o componente del sistema para satisfacer un contrato, estándar, especificación u otro documento formalmente impuesto.
- III. Una representación documental de una condición o capacidad como en I o en II.” (IEEE, 1998)

Los requisitos deben ser:

- Especificados por escrito.
- Posibles de probar o verificar.
- Descrito como una característica del sistema a entregar (qué es lo que el sistema debe hacer y no cómo lo debe hacer).
- Lo más abstracto y conciso posible. (Booch, y otros, 2000)

Tipos de Requisitos

En la IR existen varios tipos de requisitos de software como son:

Requisitos Funcionales: Son capacidades o condiciones que el sistema debe cumplir.

Requisitos No Funcionales: son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Estos son fundamentales en el éxito del producto, normalmente están vinculados a requerimientos funcionales, es decir una vez se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser.

Existen múltiples categorías para clasificar a los requerimientos no funcionales, siendo las siguientes representativas de un conjunto de aspectos que se deben tener en cuenta, aunque no limitan a la definición de otros.

- Requerimientos de software.
- Requerimientos de hardware.
- Restricciones en el diseño y la implementación.
- Requerimientos de apariencia e interfaz externa.
- Requerimientos de seguridad.
- Requerimientos de usabilidad.
- Requerimientos de soporte.

1.2.4. Actividades de la Ingeniería de Requisitos

El proceso de IR puede ser descrito a través de 4 actividades fundamentales, las cuales se explican a continuación:

1. Identificación de Requisitos

Esta actividad se refiere a la captura y descubrimiento de los requisitos que deberán ser implementados. Es una actividad más “humana” que técnica, se identifican a los interesados (stakeholders) y se establecen las primeras relaciones entre ellos y el equipo de trabajo. En principio parece una tarea relativamente fácil, preguntar al cliente, usuarios y a todos aquellos involucrados en el negocio y sean expertos en el tema, qué es lo que desean que haga el sistema que se va a desarrollar. (Pressman, 2002)

En realidad es mucho más que eso, a continuación Christel y Kang (Christel, y otros, 1992) plantean algunos problemas que se presentan en esta etapa, dando una medida de lo compleja que puede ser esta tarea y de los aspectos que hay que tener en cuenta para su realización:

- **Problemas de alcance:** El límite del sistema está mal definido o los detalles técnicos innecesarios, que han sido aportados por los clientes/usuarios, pueden confundir más que clarificar los objetivos del sistema.
- **Problemas de comprensión:** Los clientes/usuarios no están completamente seguros de lo que necesitan, tienen una pobre comprensión de las capacidades y limitaciones de su entorno de computación, no existe un total entendimiento del dominio del problema, existen dificultades para comunicar las necesidades al ingeniero del sistema, la especificación de requisitos está en conflicto con las necesidades de otros clientes/usuarios, o se especifican requisitos ambiguos o poco estables.
- **Problemas de volatilidad:** Los requisitos cambian con el tiempo.

El objetivo fundamental de esta actividad es la obtención de una especificación preliminar detallada (informal) de las necesidades de los usuarios del software a desarrollar. (Ávila, y otros)

2. Análisis de Requisitos y Negociación

Es la actividad de la IR en la cual se estudia la información extraída durante la actividad anterior, para identificar la presencia de áreas no detectadas, requisitos contradictorios y peticiones que aparecen como vagas e irrelevantes. (Ávila, y otros) Una vez recopilados los requisitos, el producto obtenido configura la base del análisis de requisitos, estos se agrupan por categorías y se organizan en subconjuntos, se estudia cada uno en relación con el resto, se examinan si existen conflictos en cuanto a su consistencia, completitud y ambigüedad y se clasifican en base a las necesidades de los clientes/usuarios. (Pressman, 2002)

El ingeniero del sistema debe resolver estos conflictos a través de un proceso de negociación. Los clientes, usuarios y el resto de los intervinientes deberán clasificar sus requisitos y discutir los posibles conflictos según su prioridad. Los riesgos asociados con cada requisito serán identificados y analizados. Se efectúan estimaciones del esfuerzo de desarrollo para valorar el impacto de cada requisito en el coste del proyecto y en el plazo de entrega. Utilizando un procedimiento iterativo, se irán eliminando requisitos, combinando y modificando para conseguir satisfacer los objetivos planteados.

El análisis de requisitos permite al ingeniero de sistemas especificar las características operacionales del software (función, datos y rendimientos), indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software. (Pressman, 2002)

El objetivo del análisis es descubrir problemas en la declaración informal de requisitos generados durante la captura de los mismos. (Ávila, y otros)

3. Especificación de Requisitos

La especificación de requisitos, puede ser un documento escrito, un modelo gráfico, una colección de escenarios de uso, un prototipo o una combinación de lo anteriormente citado, donde se describe lo que hay que desarrollar, no el cómo ni el cuándo. (Pressman, 2002)

El objetivo de esta actividad es obtener un documento de especificación de requisitos de software (ERS) que defina, de forma completa, precisa y verificable, los requisitos que debe cumplir el sistema, tanto funcional como no funcional. No debe incluir requisitos innecesarios, que no hayan sido solicitados por el cliente, ni incluir detalles sobre el diseño del sistema. (Ávila, y otros)

4. Validación de Requisitos

El resultado del trabajo realizado es consecuencia de la ingeniería de requisitos y es evaluada su calidad en la fase de validación. La validación de requisitos examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencias, sin omisiones, que los errores detectados hayan sido corregidos, y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto. (Pressman, 2002)

Además se comprueba que la ERS se ajusta a las necesidades de clientes/usuarios y otros implicados, o sea, se verifica con los stakeholders que sus necesidades fueron adecuadamente interpretadas.

Los objetivos de esta actividad son la comprobación de la consistencia, completitud, corrección, precisión del documento, así como el descubrimiento de problemas en él, antes de comprometer recursos en su implementación. (Ávila, y otros)

1.2.5. Técnicas para llevar a cabo la Ingeniería de Requisitos

La Ingeniería de Requisitos tiene un valor elevado en el desarrollo de software, por lo tanto es necesario realizar la misma de la forma correcta. Es por esta razón que a continuación se fundamentan las técnicas más utilizadas para llevar a cabo las actividades de la IR, las cuales han sido aplicadas al presente trabajo.

Entrevistas

Es una de las técnicas de identificación más usada, la cual consiste en establecer un canal de comunicación directa entre las personas destinatarias del sistema y el Equipo de Desarrollo. Las entrevistas planeadas generalmente se dan de forma iterativa y realimentada. Las entrevistas son dirigidas normalmente por el personal más experto del Equipo de Desarrollo, quienes junto con un equipo interdisciplinario de profesionales de otras áreas, como la psicología y el derecho, son los encargados de orientar las entrevistas de tal forma que la información obtenida a través de ellas sea relevante al proceso. En esta técnica se pueden identificar tres fases: la preparación, la realización y el análisis de la información obtenida. (Komer, 1993)

Grabaciones de video y audio

Existen dos formas de utilizar las grabaciones: como registro y apoyo de las entrevistas, y para analizar algún proceso en particular. En cuanto a su función de apoyo, es importante porque permite centrar la atención en la entrevista en sí en vez de distraerse tomando notas de todo lo que se dice. Esta técnica permite analizar los temas con más detenimiento y con una visión más global. Cuando se trata de analizar algún proceso en particular, su ayuda es inestimable (sobre todo las filmaciones de video) ya que permite ver y analizar en detalle ese proceso la cantidad de veces que sea necesario. Además que filmando el lugar de trabajo se está capturando el proceso de trabajo, lo que evita que impongamos nuestras expectativas y preferencias.

Por último, es necesario mencionar que siempre es conveniente comenzar las grabaciones con preguntas poco importantes que sirvan para "relajar el ambiente", ya que el entrevistado puede ponerse nervioso durante los primeros minutos de grabación. Se les debe dar tiempo a las personas para que se sientan cómodas con la idea de ser grabados o filmados. (Cueto, y otros, 2007)

Sistemas Existentes

Esta técnica consiste en analizar distintos sistemas ya desarrollados que estén relacionados con el sistema a ser construido. Algunas de las ventajas de esta técnica son:

- Se pueden analizar las interfaces de usuario, observando el tipo de información que se maneja y cómo es manejada. Esto puede ser útil para descubrir información importante a tener en cuenta, información que tal vez el cliente/usuario haya fallado en comunicar.
- Es útil analizar las distintas salidas que los sistemas producen (listados, consultas, etc.), porque siempre pueden surgir nuevas ideas sobre la base de estas.
- Estos sistemas ya están en producción, ya han pasado por la curva de aprendizaje del dominio del problema. Entonces, cuando se analizan estos sistemas, se tiene que tratar de pensar, por ejemplo, por qué manejan cierta información y para qué sirve, lo que resultará de suma utilidad para el cliente.

Cuando se utiliza esta técnica se puede realizar a priori sin que intervenga el cliente/usuario para ello, existen en internet cantidad de demos de productos que pueden resultar similares y, también, se pueden establecer contactos con profesionales que desarrollan sistemas de características comparables, aunque esto requiere de cierto grado de trabajo (investigación y análisis). También es recomendable que luego de haber analizado el sistema, se le muestre al cliente/usuario, ya que por su experiencia puede sugerir importantes ideas nuevas. (Cueto, y otros, 2007)

Tormenta de ideas (Brainstorming)

Esta es una técnica que se usa para generar ideas. La intención en su aplicación es la de generar la máxima cantidad posible de requerimientos para el sistema. No hay que detenerse en pensar si la idea es o no del todo utilizable. La intención de este ejercicio es generar, en una primera instancia, muchas ideas. Luego, se irán eliminando en base a distintos criterios como, por ejemplo, "caro", "impracticable", "imposible", etc.

Las reglas básicas a seguir son:

- Los participantes deben pertenecer a distintas disciplinas y, preferentemente, deben tener mucha experiencia. Esto trae aparejado la obtención de una cantidad mayor de ideas creativas.
- Conviene suspender el juicio crítico y se debe permitir la evolución de cada una de las ideas, porque si no se crea un ambiente hostil que no alienta la generación de ideas.

- Por más locas o salvajes que parezcan algunas ideas, no se las debe descartar, porque luego de maduras probablemente se tornen en un requerimiento sumamente útil.
- A veces ocurre que una idea resulta en otra idea, y otras veces se pueden relacionar varias ideas para generar una nueva.
- Escribir las ideas sin censura (Robertson, y otros, 1999)(Arango, 2002)

Arqueología de documentos

Con la aplicación de esta técnica se trata de determinar posibles requerimientos sobre la base de inspeccionar la documentación utilizada por la empresa; por ejemplo, manuales de procedimientos, reglamentos, boletas, facturas etc. Esta técnica sirve más que nada como complemento de las demás técnicas, y ayuda a obtener información que de otra manera sería sumamente difícil conseguir. En definitiva, se debe recolectar cualquier formulario o documento que sea utilizado para registrar o enviar información.

Para el análisis de cada uno de estos documentos, se deben realizar algunas preguntas, como:

- ¿Cuál es el propósito de este documento?
- ¿Quién lo usa? ¿Por qué? ¿Para qué?
- ¿Cuáles son las tareas que realizan con este documento?
- ¿Se puede encontrar una relación entre los documentos?
- ¿Cuál es el proceso que realiza la conexión?
- ¿Cuál es el documento que da más problemas a los usuarios? (Cueto, y otros, 2007)

Talleres de Trabajo basados en los Casos de Uso

Estos talleres de trabajo se realizan entre el cliente/usuario y el equipo de requerimientos. La primera parte del taller de trabajo consiste en generar los escenarios o casos de uso, los cuales describirán cómo el sistema será utilizado. Para esto se necesita la información que tiene para brindar el usuario/cliente. La idea es conversar por medio de los casos de uso y extraer de los usuarios las cosas esenciales que suceden cuando ocurre un evento determinado. Así, se trata de definir la serie de usuarios y reconocer los pasos que se realizan para el caso de uso en estudio. Luego se pregunta si los pasos registrados están bien o si hay que cambiarlos o mejorarlos. Como resultado de este proceso se obtiene un excelente bosquejo del caso de uso. (Cueto, y otros, 2007)

Prototipos

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

Los prototipos se pueden clasificar en:

- **Prototipo evolutivo:** Que no es más que realizar evoluciones sobre la base del mismo prototipo hasta determinar claramente los requerimientos.
- **Prototipo Bosquejado:** El analista de requerimientos simula las respuestas del sistema y realiza bosquejos de las interfaces de usuario y por otro lado el usuario, que es quien realiza las entradas ("utiliza el prototipo"). También se puede llevar el caso de uso y bosquejar la interface de usuario y mediante el diálogo, manejar la interactividad entre el usuario y el sistema.
- **Prototipo (Tangible y Usable):** Los términos tangible y usable se refieren a desarrollar una aplicación (software) con la cual pueda interactuar como si fuera la aplicación final.

Cualquiera sea la herramienta de software que se elija utilizar para desarrollar el prototipo, se debe tener en cuenta los siguientes puntos:

- Debe demandar poco esfuerzo para realizar los cambios.
- Debe poseer amplia flexibilidad para el manejo de las interfaces de usuario.
- Debe consumir poco tiempo para generar un nuevo prototipo (maqueta). (Robertson, y otros, 1999)

Documento Especificación de Requisitos | Casos de uso

El objetivo del documento ERS (Especificación de Requerimientos de Software) es especificar los requerimientos del sistema. En este documento se incluye una lista con los requerimientos del producto con las respectivas referencias a los documentos de todos los casos de uso que satisfacen los requerimientos. El documento de casos de uso es un documento narrativo que describe 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.

Los requerimientos se pueden expresar de diferentes formas, desde texto sin formato estricto hasta expresiones en un lenguaje formal, pasando por todas las formas intermedias. La mayoría de los requerimientos funcionales, se pueden expresar con casos de uso. (Craig, 1999) (Jacobson, 1992)

1.3. Tendencias actuales sobre metodologías, herramientas case y lenguaje de modelado para el desarrollo de software.

El siglo XXI se enfrenta a la creciente implantación de la sociedad del conocimiento aplicado al desarrollo de las nuevas tecnologías. La era del conocimiento en que se vive no sólo está cambiando la sociedad en si misma, sino que los nuevos modelos de negocios requieren la reformulación de nuevos conceptos. El software se ha convertido en el elemento clave del desarrollo, mediante la evolución de los sistemas y productos informáticos, ha pasado de ser una resolución de problemas especializada y una herramienta de análisis de información, a ser una industria por si misma. (Pressman, 2002)

En la actualidad en la Industria de Software hay tendencia al crecimiento del volumen y complejidad de los productos, los proyectos están excesivamente tarde, se exige mayor calidad y productividad en menos tiempo y hay insuficiente personal calificado; para enfrentar esta situación las empresas requieren desarrollar o adquirir una disciplina en el desarrollo del software, lo que implica mejorar el proceso de desarrollo de software haciendo uso de métodos y herramientas que permitan obtener software de calidad. (Booch, y otros, 2000)

1.3.1. Procesos de Desarrollo de Software

En los últimos tiempos la cantidad y variedad de los procesos de desarrollo ha aumentado de forma impresionante, sobre todo teniendo en cuenta el tiempo que estuvo en vigor como ley única el famoso desarrollo en cascada. Se podría decir que en estos últimos años se han desarrollado dos corrientes en lo referente a los procesos de desarrollo, los llamados *métodos pesados* y los *métodos ligeros*.

La diferencia fundamental entre ambos es que mientras lo métodos pesados intentan conseguir el objetivo común por medio de orden y documentación, los métodos ligeros (también llamados métodos ágiles) tratan de mejorar la calidad del software por medio de una comunicación directa e inmediata entre las personas que intervienen en el proceso. (Molpeceres, 2002)

Un proceso define *quién* está haciendo *qué*, *cuando* y *cómo* alcanzar un determinado objetivo. (Booch, y otros, 2000)

Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software.

- Partiendo de estos conceptos un proceso de desarrollo de software debe:
- Servir como guía para todos los implicados en el proyecto.
- Ser comprendido por el equipo de trabajo de forma que entiendan el papel que cumplen en el desarrollo del software.
- Ser fácilmente aplicable en cualquier entorno.
- Ser adaptable a cualquier tipo de proyecto software según las necesidades del mismo.

A continuación se realiza un análisis de los dos procesos de desarrollo más utilizados tanto en procesos ágiles como pesados, donde se muestran las características de cada uno de ellos.

Proceso Unificado de Rational (RUP)

RUP es uno de los procesos más generales de los existentes actualmente, es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del lenguaje de modelado unificado y trabajo de muchas metodologías utilizadas por los clientes, este está pensado para adaptarse a cualquier proyecto y no tan solo de software.

RUP como proceso, en su modelación define como sus principales elementos:

- **Trabajadores (“quién”)**: Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.
- **Actividades (“cómo”)**: Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- **Artefactos (“qué”)**: Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- **Flujo de actividades (“Cuándo”)**: Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable. (Booch, y otros, 2000)

Un proyecto realizado siguiendo RUP se divide en 4 fases y 9 flujos de trabajos:

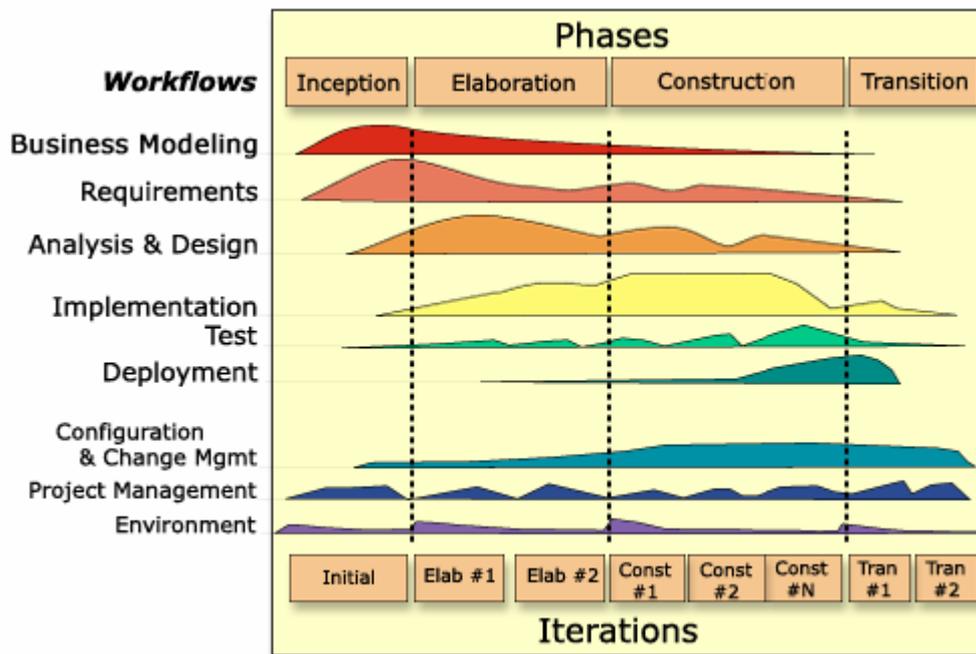


Figura 1.1: Fases y flujos de trabajo de la Metodología RUP

Fases

- **Inicio (Objetivos):** Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- **Elaboración (Arquitectura):** Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen.
- **Construcción (Funcionalidad Operativa):** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario.
- **Transición (Release o Liberación del Sistema):** El producto software ya está listo para su instalación en las condiciones reales.

El proceso de desarrollo se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo está compuesto por las 4 fases descritas anteriormente, el resultado final de un ciclo es una versión del sistema. En cada fase se ejecutarán una o varias iteraciones (de tamaño variable según el proyecto), y dentro de cada una de ellas se seguirá un modelo de cascada donde se llevarán a cabo los flujos de trabajo que se describen a continuación.

Flujos de Trabajo

Los 6 primeros flujos de trabajo se consideran flujos de ingeniería y los 3 restantes flujos de apoyo.

- **Modelación del negocio:** Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- **Requerimientos:** Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- **Análisis y diseño:** Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.
- **Implementación:** Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- **Prueba (Testeo):** Busca los defectos a lo largo del ciclo de vida.
- **Instalación:** Produce la liberación del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.) para entregar el software a los usuarios finales.
- **Administración del proyecto:** Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.
- **Administración de configuración y cambios:** Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc.
- **Ambiente:** Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización. (Booch, y otros, 2000)

El ciclo de vida de RUP se caracteriza por:

- **Dirigido por casos de uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos, a partir de aquí los casos de uso guían el proceso de desarrollo.
- **Centrado en la Arquitectura:** La arquitectura muestra la visión común del sistema completo, 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. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura.
- **Iterativo e Incremental:** RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Es práctico dividir el trabajo en partes más pequeñas o mini-proyectos. Cada mini-proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. (Booch, y otros, 2000)

Programación Extrema (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad, utilizada para proyectos de corto plazo y equipos de trabajo pequeños. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

XP intenta minimizar el riesgo de fallo del proceso por medio de la disposición permanente de un representante competente del cliente a disposición del equipo de desarrollo. Este representante debería estar en condiciones de contestar rápida y correctamente a cualquier pregunta del equipo de desarrollo de forma que no se retrase la toma de decisiones. (Molpeceres, 2002)

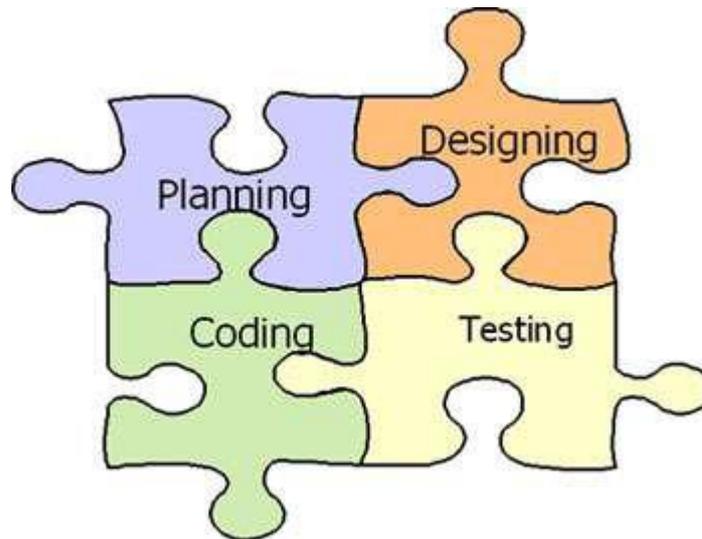


Figura 1.2: Flujos de trabajo de la Metodología XP

Características de XP

- **Pruebas Unitarias:** Se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se puedan hacer pruebas de las fallas que pudieran ocurrir. Es como adelantarse a obtener los posibles errores.
- **Prefabricación:** Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa. (Sanchez, 2004)

Lo que propone

- Empieza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.
- El cliente o el usuario se convierte en miembro del equipo.

Lo fundamental en este tipo de metodología

- La comunicación, entre los usuarios y los desarrolladores.
- La simplicidad, al desarrollar y codificar los módulos del sistema.
- La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

En XP se programa solo la funcionalidad que es requerida para la liberación actual del software. Es decir, una gran flexibilidad y capacidad de configuración solo será implementada cuando sea necesaria para cumplir los requerimientos del release. Se sigue un diseño evolutivo con la siguiente premisa: conseguir la funcionalidad deseada de la forma más sencilla posible. De ahí una variación adecuada del famoso *KISS (Keep It Simple Stupid), Keep things as simple as possible* (mantén las cosas tan sencillas como sea posible). Este diseño evolutivo hace que no se le de apenas importancia al análisis como fase independiente, puesto que se trabaja exclusivamente en función de las necesidades del momento. (Sanchez, 2004)

1.3.2. Lenguaje de Modelado

En cualquier proyecto de ingeniería como la construcción de un gran edificio, un avión, la construcción de un procesador de textos o un software de comunicaciones para Internet, requieren de etapas de modelado que permitan experimentar y visualizar el sistema que se construirá. (Barrientos Enríquez)

Modelado: Es el diseño de un proyecto antes de su construcción, es la visualización de lo que se quiere construir. (Geraldo)

Desde los inicios de la informática se han estado utilizando distintas formas de representar los diseños con algún tipo de modelo gráfico, la falta de estandarización en la representación gráfica de un modelo impedía que los diseños gráficos realizados se pudieran compartir fácilmente entre distintos diseñadores y entre los implicados en el desarrollo del software, para que fueran de entendimiento común para todos, con ese objetivo se creó lo que se conoce como lenguaje de modelado que no es más que un conjunto estandarizado de símbolos y de modos de disponerlos para modelar (parte de) un diseño de software.

Lenguaje Unificado de Modelado (UML)

UML es el lenguaje de modelado de sistemas de software más conocido en la actualidad; es el estándar internacional aprobado por la OMG (Object Management Group) traducido al español Grupo de Gestión de Objetos, consorcio creado en 1989 responsable de la creación, desarrollo y revisión de especificaciones para la industria del software.

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra gran cantidad de software. (Jacobson, y otros, 2000) UML es un grupo de especificaciones de notación orientadas a objeto, las cuales están compuesta por diversos elementos gráficos que se combinan para conformar distintos diagramas, que representan las diferentes etapas del desarrollo de un proyecto de software. Debido a que el UML es un lenguaje, cuenta con reglas para combinar tales elementos. Es importante destacar que un modelo UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema.

UML tiene un vocabulario en el que se identifican:

- **Elementos:** Abstracciones que constituyen los bloques básicos de construcción.
- **Relaciones:** Ligan los elementos.
- **Diagramas:** Es la representación gráfica de un conjunto de elementos y sus relaciones. Visualizan un sistema desde diferentes perspectivas.

UML propone los siguientes diagramas:

- Diagrama de Clases.
- Diagrama de Objetos.
- Diagrama de Casos de Usos.
- Diagrama de Interacción (Secuencia y Colaboración).
- Diagrama de Estados.
- Diagrama de Actividades.
- Diagrama de Componentes.
- Diagrama de Despliegue. (Jacobson, y otros, 2000)

1.3.3. Herramientas de Modelado

El desarrollo de software ha progresado bastante en la última década, y las herramientas de modelado forman un componente importante en el entorno de desarrollo de hoy en día.

1.3.4. Herramientas CASE

CASE: Computer Aided Software Engineering. En su traducción al español significa Ingeniería de Software Asistida por Computación.

Se considera CASE como la aplicación de métodos y técnicas a través de las cuales se hacen útiles a las personas comprender las capacidades de las computadoras, por medio de programas, de procedimientos y su respectiva documentación. Concentrando su atención en el uso de estas herramientas, para el desarrollo de proyectos informáticos que tengan como objetivo la automatización de procedimientos administrativos; se puede decir que: Las herramientas CASE representan una forma que permite modelar los procesos de negocios de las empresas y desarrollar los Sistemas de Información Gerenciales.

Algunos de los componentes de las herramientas CASE permiten:

- Confeccionar la definición de requerimientos de los usuarios.
- Mejorar el diseño de los sistemas.
- Mejorar la eficiencia en la programación (por su generación automática de códigos).
- Otorgar a la administración un mejor soporte en la documentación.
- Propiedades que deben abarcar las herramientas CASE:
 - Tener una interfaz gráfica y textual, que le permita al usuario manejar los objetos de diseño.
 - Contar con un Diccionario de Datos, a fin de rastrear y controlar los objetos diseñados.
 - Disponer de un conjunto de herramientas que permitan chequear las reglas del diseño y analizar la lógica del diseño.

Se considera que las herramientas CASE son un elemento muy importante, que permiten al administrador de un proyecto informático, desarrollar este de forma eficaz y eficiente, debido a su capacidad de representar objetos de datos de negocio, sus relaciones y la forma en que fluyen estos objetos de datos entre distintas zonas del negocio. (Pressman, 2002)

A continuación se analizan varios tipos de herramientas CASE donde se muestran sus características, ventajas y desventajas.

Rational Rose

Rational Rose es una herramienta de producción y comercialización establecidas por Rational Software Corporation (actualmente parte de IBM). Rose es un instrumento operativo conjunto que utiliza el Lenguaje Unificado de Modelado (UML) como medio para facilitar la captura de dominio de la semántica, la arquitectura y el diseño. (Reque, y otros, 2007)

- Herramienta CASE que da soporte al modelado visual con UML ofreciendo distintas perspectivas del sistema.
- Da soporte al Proceso Unificado de Rational (RUP)
 - Modelado de Negocio.
 - Captura de Requisitos (parcial).
 - Análisis y Diseño (Completo).
 - Implementación (como ayuda).
 - Control de Cambio y gestión de configuración.

Rational Rose ofrece:

- Diseño dirigido por modelos que redundan en una mayor productividad de los desarrolladores, admitiendo UML, COM, OMT y Booch.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad en múltiples plataformas. (Vizcaíno, y otros)

Visual Paradigm

El Visual Paradigm es una Suite completa de herramientas CASE. Independiente de la plataforma y dotada de una buena cantidad de productos o módulos para facilitar el trabajo durante la confección de un software, así como garantizar la calidad del producto final. Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste.

Permite generar:

- Diagramas de Casos de Uso.
- Diagramas de Clases.
- Diagramas de Secuencia.
- Diagramas de Comunicación.
- Diagramas de Estado.
- Diagramas de Componentes.
- Diagramas de Despliegue.
- Diagramas de Objetos.
- Diagramas de Interacción.
- Diagramas de Entidad Relación.
- Diagramas ORM (Object-relational mapping).
- Diagramas de Procesos del Negocio.
- Diagramas de EJB (Enterprise JavaBeans).

Principales características de Visual Paradigm:

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio, Proceso, Decisión, Actor de Negocio, Documento.
- Modelado colaborativo con CVS (Concurrent Versions System) y Subversión (nueva característica).
- Ingeniería inversa - Código a modelo, código a diagrama.
- Ingeniería inversa Java, C++, Esquemas XML (Extensible Markup Language), XML, NET exe/dll, CORBA IDL.

- Generación de código - Modelo a código, diagrama a código.
- Editor de Detalles de Casos de Uso, entorno todo en uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- Diagramas EJB - Visualización de sistemas EJB.
- Generación de código y despliegue de EJB's - Generación de beans para el desarrollo y despliegue de aplicaciones.
- Diagramas de flujo de datos.
- Soporte ORM - Generación de objetos Java desde la base de datos.
- Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos - Desde Sistemas Gestores de Bases de Datos existentes a diagramas de Entidad-Relación.
- Generador de informes para generación de documentación.
- Distribución automática de diagramas - Reorganización de las figuras y conectores de los diagramas UML.
- Importación y exportación de ficheros XMI.
- Integración con Visio - Dibujo de diagramas UML con plantillas (stencils) de MS Visio.

El Visual Paradigm permite escribir toda la especificación de un caso de uso sin necesidad de utilizar una herramienta externa como editor de texto. Es posible crear Especificaciones de Casos de Uso utilizando plantillas que se encuentran definidas, o que pueden ser creadas por los usuarios. (freedownloadmanager)

Enterprise Architect

El desarrollo de Software es un proceso complejo y a menudo difícil que requiere la síntesis de muchos sistemas. Desde el modelado y diseño hasta el código, administración del proyecto, pruebas, despliegue, administración de cambios y más, Enterprise Architect (EA) es una herramienta de modelado basada en UML que se ha convertido en una parte esencial para administrar esa complejidad.

EA combina el poder de la última especificación UML 2.1 con alto rendimiento, interfaz intuitiva, para traer modelado avanzado al escritorio, y para el equipo completo de desarrollo e implementación, esta herramienta puede equipar al equipo entero, incluyendo analistas, evaluadores, administradores de proyectos, personal del control de calidad, equipo de desarrollo y más.

Es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multiusuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad.

Provee trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, a través de la implementación y el despliegue. Combinados con la ubicación de recursos y tareas incorporados, los equipos de Administradores de Proyectos y Calidad están equipados con la información que ellos necesitan para ayudarles a entregar proyectos en tiempo.

Soporta los 13 diagramas de UML 2 y más:

Diagramas Estructurales

- Clase.
- Objeto.
- Compuesto.
- Paquete.
- Componente.
- Despliegue.

Diagramas de Comportamiento

- Casos de Uso.
- Comunicación.
- Secuencia.
- Descripción de la Interacción.
- Actividad.
- Estado.
- Tiempo.

Extendidos

- Análisis (actividad simple).
- Personalizado (para requisitos, cambios, UI).

Soporta generación e ingeniería inversa de código fuente para muchos lenguajes populares, incluyendo C++, C#, Java, Delphi, VB.Net, Visual Basic y PHP. Con un editor de código fuente con "resaltador de sintaxis" incorporado, EA permite navegar y explorar el modelo de código fuente en el mismo ambiente. Para aquellos que trabajan en Eclipse o Visual Studio.Net. Las plantillas de generación de código permiten personalizar el código fuente generado a las especificaciones de su compañía.

El objetivo de Enterprise Architect es proveer todos estos elementos juntos en un entorno que sea tanto coherente como flexible. Un soporte extenso para la notación de UML 2.1 se combina con las herramientas de administración de procesos que le permiten decidir sobre una metodología, además soporta un amplio rango de diagramas del UML 2.0, permitiendo modelar casi cualquier sistema. (sparxsystems)

Analizados los procesos de desarrollo expuestos anteriormente se llega a la conclusión que el más óptimo para desarrollar este sistema es RUP, ya que este es un proceso bien definido y estructurado; a la vez que es un producto que provee un marco de proceso adaptable a las necesidades y características de cada proyecto específico, tiene como particularidad que en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software. Se utiliza como lenguaje de modelado el UML ya que este, fue diseñado para usarse con un proceso iterativo, incremental, guiado por casos de uso y centrado en la arquitectura como es el caso de RUP. De todas las herramientas expuestas anteriormente se considera que la más óptima para darle solución a este trabajo es Enterprise Architect pues esta permite una trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, lo que da la posibilidad de realizar una gestión de requisitos de forma que permita controlar y seguir los requisitos y sus cambios en cualquier momento. Esta es una herramienta colaborativa, por lo que soporta múltiples usuarios trabajando sobre el mismo proyecto además que permite el control de versiones.

Conclusiones

Después de realizar un estudio del proceso de Tesorería que se desarrolla actualmente en Venezuela y de soluciones software que automatizan procesos como este, se llega a la conclusión a partir de las características del mismo, que es necesario utilizar una metodología como RUP, con la cual se crearán los artefactos correspondientes a los flujos de trabajos de modelación del negocio y requerimientos a través del lenguaje de modelado unificado, usando como herramienta CASE el Enterprise Architect y empleando la IR y las técnicas que se aplican en la ejecución de cada una las actividades que en ella se llevan a cabo durante todo la etapa de levantamiento de requisitos. Durante este proceso, se logrará identificar los requisitos funcionales y no funcionales del software y a partir de esto se realizará su análisis y validación lo que permitirá comprobar si en realidad se le dio cumplimiento a las necesidades expresadas por los clientes.

CAPÍTULO 2: MODELACIÓN DEL NEGOCIO

El primer paso en el proceso de desarrollo de software es precisamente alcanzar cierto nivel de conocimientos sobre el problema en cuestión, esto se logra a través de la Modelación del Negocio, primer flujo de trabajo que propone la metodología RUP (la cual se escogió para realizar este trabajo) y objetivo fundamental a desarrollar en este capítulo. En este flujo se identifican artefactos como: Modelo de Casos de Uso del Negocio que contiene Casos de Uso y Actores del Negocio, Modelo de Objetos el cual está compuesto por Entidades y Trabajadores, Reglas del Negocio y lleva a cabo la Realización de Casos de Usos, los cuales se explican más adelante a medida que se vayan desarrollando.

2.1. ¿En que consiste la modelación del negocio?

Un sistema por pequeño que sea generalmente es complicado por lo que es necesario dividirlo en modelos que permitan comprenderlo y gestionar su complejidad, además de que organicen y presenten los detalles importantes de problemas reales que se vinculen con el sistema informático a construir. El proceso de modelación da la posibilidad de obtener una visión de la organización lo que permite definir los procesos, roles y responsabilidades que en ella se llevan a cabo, mostrándolos en los modelos de casos de uso del negocio y de objetos.

En la modelación del negocio se describe el negocio en términos de casos de usos del negocio, que corresponde a lo que generalmente se le llama procesos. En ella se describen los procesos, existentes u observados, con el propósito de comprenderlos. Se especifican qué procesos del negocio soportará el sistema, además de identificar los objetos del dominio o del negocio, implicados en el mismo. Este modelo establece las competencias que se requieren de cada proceso: sus trabajadores, sus responsabilidades y las operaciones que llevan a cabo. (Booch, y otros, 2000)

A continuación se muestra el flujo de actividades que se desarrollarán en este flujo.



Figura 2.1: Flujo de actividades de la Modelación del Negocio

2.1.1. Objetivos del Modelado del Negocio

El modelado de negocio tiene varios objetivos que son necesarios tener en cuenta durante su realización como son:

- Comprender la estructura y la dinámica de la organización en la cual se va a implantar el sistema.
- Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- Asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.
- Derivar los requerimientos del sistema que va a soportar la organización. (Booch, y otros, 2000)

2.2. Modelado del Negocio de Tesorería

2.2.1. Alcance

En la modelación del negocio del módulo de Tesorería se identifican y describen, ya sea, de forma textual o gráfica los principales procesos (Casos de Usos) que ocurren y las personas que intervienen en la realización de los mismos (Actores). A partir de la realización de los Casos de Usos del Negocio se podrá determinar las actividades automatizables, lo que dará una medida de las necesidades expresadas por el cliente, siendo esto uno de los objetivos principales de este trabajo.

2.2.2. Descripción del Negocio

La Tesorería es una de las funciones críticas dentro del área financiera, debido a la importancia y los cambios constantes que tiene la información que en ella se maneja, por lo que el negocio que se desarrolla alrededor de ella es bastante complejo, este se encarga fundamentalmente del registro y control de las cuentas bancarias que pertenecen a las Unidades Administradoras, ya sean a las Desconcentradas o a la Central, y de las operaciones que se realizan en estas cuentas, además del pago de las obligaciones contraídas por estas unidades con proveedores debido a la compra de un bien o la solicitud de un servicio, determinando para esto en todo momento la disponibilidad financiera con que cuenta la unidad para realizar dichas operaciones. Este negocio se desarrolla básicamente en tres escenarios: Unidad de Tesorería (UT), Unidad Administradora (UA) y Banco.

El Banco en coordinación con la UT son los encargados de registrar las cuentas bancarias que han sido aperturadas en las UA, siendo la UT la responsable de registrar en el Libro Auxiliar -que se le crea a dicha cuenta-, las operaciones bancarias que se realizan sobre esta. A partir del registro de las operaciones bancarias por parte del Banco, en los Estados de Cuentas y del registro de las operaciones bancarias por parte de la UT, en el Libro Auxiliar, la UT realiza el proceso de conciliación bancaria, detectando de esta forma si hubo errores en las transacciones realizadas.

Otra de las funciones del proceso de Tesorería es darle solución a las obligaciones por pagar de las Unidades Administradoras, esto se lleva a cabo cuando la UT solicita a las UA las prioridades de sus deudas para la elaboración del Plan de Pago, las UA a su vez emiten las solicitudes de órdenes de pago a la UT, esta en base al flujo de caja y la disponibilidad financiera autoriza el pago de las mismas a los proveedores con los cuales han sido contraídas dichas obligaciones.

A partir del negocio descrito anteriormente se identificarán los actores, casos de uso, trabajadores y entidades que en él se encuentran implicados.

2.2.3. Actores y Trabajadores del Negocio

Actor del Negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.

Son actores candidatos del negocio:

- Socios.
- Proveedores.
- Autoridades (legales, reguladoras, etc.).
- Propietarios, si no están dentro del negocio que se modela.
- Sistemas de información externos al negocio.
- Otras partes del negocio si este es grande y esas partes no están dentro del campo de acción del negocio que se modela. (Booch, y otros, 2000)

A continuación se muestran los actores identificados en el negocio.

Actores del Negocio	Descripción
Funcionario de la Unidad de Tesorería	Inicia las acciones que dan comienzo al proceso de realización de la colocación financiera.
Funcionario de la Unidad Administradora	Es la persona encargada de solicitar el registro y apertura de las cuentas bancarias, de la realización de la conciliación bancaria, además de solicitar la cancelación de las órdenes de pago y de solicitar la realización de la custodia de valores.
Funcionario de Presupuesto	Es el encargado de solicitar la realización de la formulación y reformulación del flujo de caja.
Funcionario del Banco	Es la persona encargada de dar inicio a los procesos de registro de chequera y solicitud de mantenimiento de la misma.

Beneficiario	Es el encargado de iniciar los procesos de solicitud de entrega de cheque y suspensión de cheque.
Tiempo	Es el actor encargado de avisar periódicamente para que se realice la revisión de los cheques que se encuentran en custodia de la Unidad de Caja.

Tabla 2.1: Descripción de los actores del negocio

Trabajador del Negocio

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores y manipulando entidades del negocio.

Entidad del Negocio:

No son más que los objetos donde se obtiene o registra información, los cuales son necesarios para llevar a cabo el flujo de actividades del negocio

A continuación se muestran los trabajadores identificados. (Booch, y otros, 2000)

Trabajadores del Negocio	Descripción
Responsable de la Unidad de Tesorería	Es la persona encargada de realizar todas las actividades que se llevan a cabo en dicha unidad entre ellas: elaborar el Plan de Pago y conformar el Flujo de Caja definitivo para saldar las obligaciones por pagar de las Unidades Administradoras (UA), registrar las cuentas bancarias pertenecientes a las UA así como abrir el libro auxiliar correspondiente a dicha cuenta para registrar las operaciones que se realizan sobre ella.
Responsable de la Unidad Administradora	Es la persona encargada en la Unidad Administradora de priorizar las obligaciones que se deben pagar y enviarlas a la Unidad de Tesorería para que éstas sean saldadas.
Comité de Planificación Financiera	Es la persona encargada de definir los pagos que se van a realizar a las UA y de determinar si existe disponibilidad financiera para realizar colocación.
Autoridad Competente del Ente	Es el encargado de revisar, autorizar y firmar todas aquellas

	decisiones que se lleven a cabo en la entidad.
Responsable del Banco	Es la persona encargada de registrar las cuentas bancarias pertenecientes a las UA, así como las firmas autorizadas que operarán con ella.
Oficina Nacional del Tesoro	Es la encargada de asignar el número de control ONT cuando se registra una cuenta en una UA.
Entidades Bancarias	Es la encargada de enviar la cotización y de realizar todos los trámites con la UT para llevar a cabo el proceso de colocación financiera.
Responsable de la Unidad de Contabilidad	Es la persona encargada en conjunto con la UT del registro y control de las operaciones bancarias, además se encarga de llevar a cabo el proceso de conciliación bancaria.
Tesorero	Encargado de firmar los documentos relacionados con las ordenes de pago solicitadas por las UA.
Responsable de la Unidad de Caja	Encargado de la custodia de los cheques y sus respectivos comprobantes hasta su entrega al beneficiario.

Tabla 2.2: Descripción de los trabajadores del negocio

2.2.4. Casos de Uso del Negocio

Un proceso de negocio es un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y manera y que emplean los recursos de la organización para dar resultados en apoyo a sus objetivos. Para identificar los procesos de negocio es muy importante tener en cuenta que deben generar un valor para el negocio o mitigar los costos del negocio.

Un caso de uso representa a un proceso, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores. Desde la perspectiva de un actor individual, define un flujo de trabajo completo que produce resultados deseables.

A continuación se muestran los casos de uso que han sido extraídos del negocio de Tesorería.

Casos de Uso del Negocio	Descripción
Realizar Formulación y Reformulación del Flujo de Caja	Este caso de uso comienza cuando el Funcionario de Presupuesto envía el plan de ejecución del presupuesto para que se elabore el Plan de Pago y a partir de esto saldar las obligaciones por pagar emitidas por las UA.
Solicitar Apertura de Cuenta Bancaria	Este caso de uso consiste en la solicitud por parte del Funcionario de la Unidad Administradora de la apertura y registro de una cuenta bancaria.
Realizar Conciliación Bancaria	Este caso de uso permite realizar la conciliación bancaria de las cuentas a partir de los estados de cuentas enviados por el banco y de las operaciones bancarias registradas en el libro auxiliar de la cuenta.
Solicitar Orden de Pago	Este caso de uso se inicia cuando el Funcionario de la UA emite solicitud de pago para la cancelación de sus obligaciones por pagar.
Realizar Custodia de Valores	Este caso de uso consiste en la selección de la chequera con la que se va a operar sobre una cuenta y de las personas que van trabajar con ella.
Realizar Colocación Financiera	Este caso de uso consiste en la solicitud por parte del Funcionario de la Unidad de Tesorería de la realización de una inversión de fondos a través de colocación financiera.
Revisar Cheques en Custodia	Este caso de uso consiste en la revisión de los cheques que están en custodia de la Unidad de Caja para determinar si existen cheques vencidos para que se efectúe su anulación.
Registrar Chequera	Este caso de uso consiste en la solicitud por parte del Funcionario del Banco del registro de la chequera.
Solicitar Mantenimiento de Chequera	Este caso de uso consiste en la solicitud por parte del Funcionario del Banco del mantenimiento de la chequera.

Entregar Cheque	Este caso de uso consiste en la solicitud de la entrega de cheque por parte del Beneficiario debido a una obligación por pagar que la UA contrajo con el mismo.
Suspender Cheque	Este caso de uso consiste en la solicitud de la suspensión de cheque por parte del Beneficiario debido la pérdida del mismo.
Anular Cheque	Este caso de uso consiste en anular cheques debido a que han caducado, por error en el documento o por la pérdida del mismo.

Tabla 2.3: Descripción de los casos de uso del negocio

2.2.5. Modelo de Casos de Uso del Negocio

A continuación se estructura el modelo de Casos de Uso del Negocio, en el cual describen los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), tales como socios y clientes, es decir, describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios.

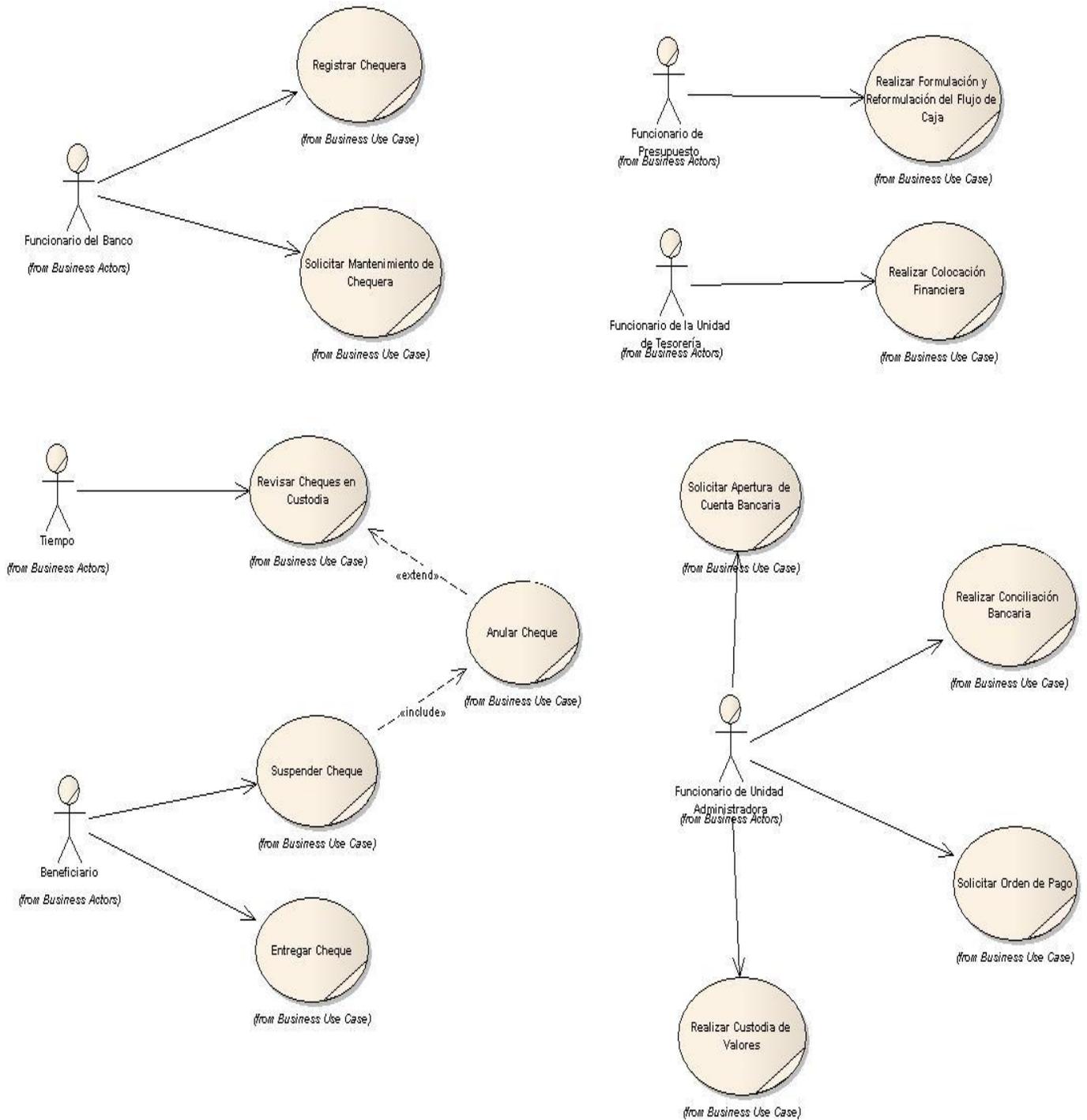


Figura 2.2: Diagrama de casos de uso del negocio

2.2.6. Realización de los Casos de Uso del Negocio

A continuación se muestra la realización de los casos de uso, en la cual se describe el flujo de eventos que ocurre en cada proceso y cómo colaboran los trabajadores y entidades del negocio para ejecutar el mismo, esta se realiza de forma textual o gráfica mediante plantillas y diagramas de actividades respectivamente. Estos diagramas permiten identificar las actividades automatizables correspondientes a cada caso de uso, lo que da una primera aproximación a los requisitos del sistema.

A continuación se muestra la descripción y los diagramas de actividades de los casos de uso más importantes.

Realizar Formulación y Reformulación del Flujo de Caja

Caso de Uso:	Realizar Formulación y Reformulación del Flujo de Caja	
Actores:	Funcionario de Presupuesto	
Resumen:	Este caso de uso comienza cuando el funcionario de presupuesto envía el plan de ejecución del presupuesto para que se elabore el Plan de Pago y a partir de esto saldar las obligaciones por pagar emitidas por las UA.	
CU asociados:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. Solicita a la Unidad de Tesorería (UT) la Formulación y Reformulación del Flujo de Caja.	2. La UT solicita el Programa de Ejecución de Presupuesto (PEP).	
3. Envía el PEP.	4. La UT recibe información y solicita a la Unidad Administradora (UA) las prioridades de sus deudas del corriente ejercicio presupuestario como el del año anterior.	
	5. La UA recibe la solicitud y establece las prioridades sobre las deudas adquiridas que no han sido canceladas y las envía a la UT.	
	6. La UT recibe la información y prepara el escenario del Flujo de Caja	
	7. La UT elabora el Plan de Pagos de acuerdo al orden de prioridad entre los gastos y determina el monto a ser	

	distribuido entre las UEL.
	8. La UT convoca reunión con el Comité de Planificación Financiera (CPF) para revisar los saldos financieros y los escenarios del Flujo de Caja.
	9. El CPF define los pagos a realizar y lo remiten a la UT
	10. La UT recibe los pagos a realizar y conforma el Flujo de Caja Definitivo. Terminando así el caso de uso.
Flujos Alternos	
Flujo Alterno 1 “”	
Acción del Actor	Respuesta del Negocio
Prioridad:	Alta

Tabla 2.4: Descripción del caso de uso Realizar Formulación y Reformulación del Flujo de Caja

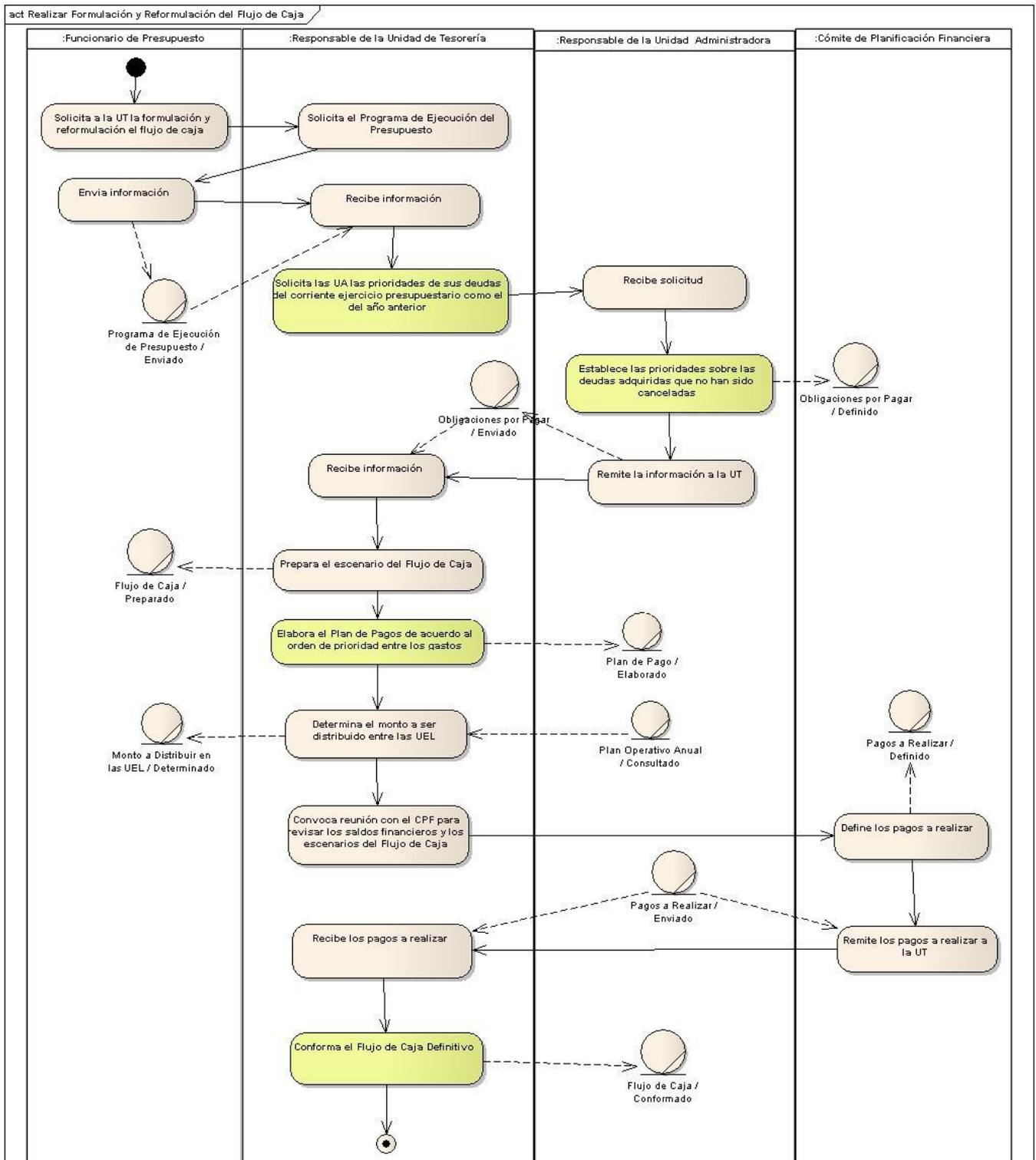


Figura 2.3: Diagrama de actividades del caso de uso Realizar Formulación y Reformulación del Flujo de Caja.

Solicitar Apertura de Cuenta Bancaria

Caso de Uso:	Solicitar Apertura de Cuenta Bancaria	
Actores:	Funcionario de la Unidad Administradora (UA)	
Resumen:	Este caso de uso consiste en la solicitud por parte del funcionario de la UA de la apertura y registro de una cuenta bancaria.	
CU asociados:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. Emite solicitud de apertura y registro de una cuenta bancaria.	2. La Autoridad Competente del Ente (ACE) recibe la solicitud y analiza las razones de la apertura.	
	3. La ACE autoriza la apertura, en caso contrario Ver Flujo Alterno 1.	
	4. La ACE procede a firmar la solicitud y la envía al Responsable de la UA.	
5. Recibe la Solicitud de Apertura y la envía a la UT.	6. La UT recibe solicitud de apertura y solicita cédula de los firmantes y acta de nombramiento como autoridad.	
7. Envía datos.	8. Recibe información y emite oficio al Banco indicando la apertura de una cuenta.	
	9. La UT en coordinación con el Banco proceden a registrar la cuenta bancaria y las firmas autorizadas para operar la misma.	
	10. La UT emite comunicación notificando a la Oficina Nacional del Tesoro (ONT) acerca de la apertura de la cuenta.	
	11. La ONT recibe comunicación y asigna a la cuenta el Nro. de Control ONT.	
	12. La UT abre Libro Auxiliar correspondiente a la cuenta bancaria. Terminando así el caso de uso.	
Flujos Alternos		
Flujo Alterno 1 ""		
Acción del Actor	Respuesta del Negocio	

	1. La ACE emite observaciones y las envía al Funcionario de la UA para ser archivada.
2. Recibe observaciones y procede al archivo de las mismas. Terminando así el caso de uso.	
Prioridad:	Alta

Tabla 2.5: Descripción del caso de uso Solicitar Apertura de Cuenta Bancaria

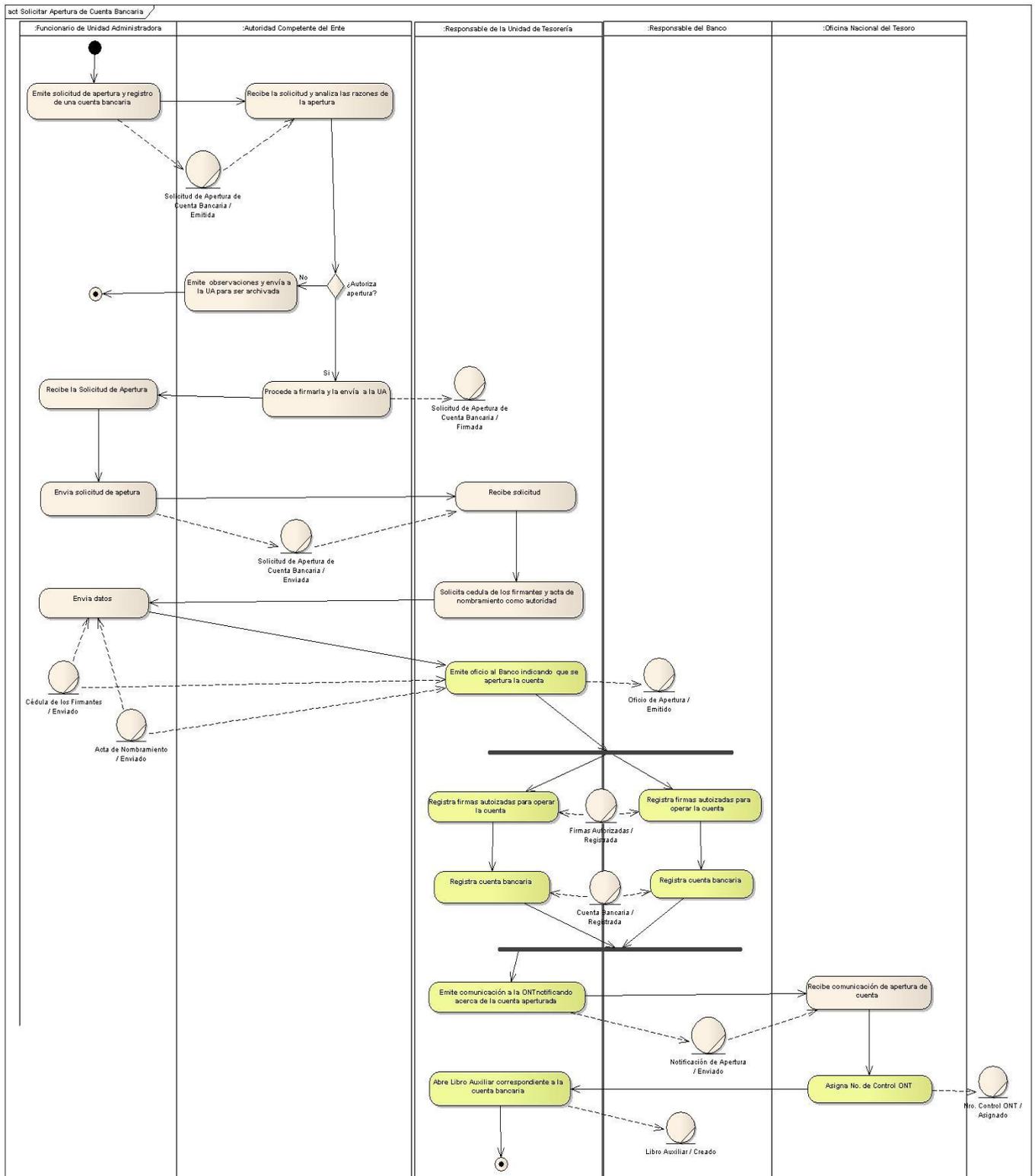


Figura 2.4: Diagrama de actividades correspondiente al caso de uso Solicitar Apertura de Cuenta Bancaria.

Realizar Conciliación Bancaria

Caso de Uso:	Realizar Conciliación Bancaria	
Actores:	Funcionario de la Unidad Administradora	
Resumen:	El caso de uso permite realizar la conciliación bancaria de las cuentas a partir de los estados de cuentas enviados por el banco y de las operaciones bancarias registradas en el libro auxiliar de la cuenta.	
CU asociados:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. Solicita realizar conciliación bancaria.	2. El Banco envía estados de cuenta.	
3. Recibe estados de cuenta y los remite a la Unidad de Contabilidad (UC).	4. La UC accede directamente a las cuentas con sus movimientos.	
	5. La UC ejecuta proceso de conciliación y determina que no existen diferencias, en caso de que existan Ver Flujo Alterno 1.	
	6. La UC realiza cuadro de conciliación y envía información a la UT.	
	7. La UT recibe información y verifica. Terminando así el caso de uso.	
Flujos Alternos		
Flujo Alterno 1 “Si existen diferencias en la conciliación”		
Acción del Actor	Respuesta del Negocio	
	1. La UC verifica el error de la diferencia, si es error del Banco, envía oficio a este para que proceda a corregir el error, en caso que no sea del Banco Ver Flujo Alterno 2.	
Flujo Alterno 2 “Si no es error del Banco”		
Acción del Actor	Respuesta del Negocio	
	1. Si el error no es del Banco la UC solicita información a la UT para verificar las transacciones con su soporte.	
	2. La UT envía información.	

	3. La UC determina las diferencias y realiza los ajustes. Terminado así el caso de uso.
Prioridad:	Alta

Tabla 2.6: Descripción del caso de uso Realizar Conciliación Bancaria

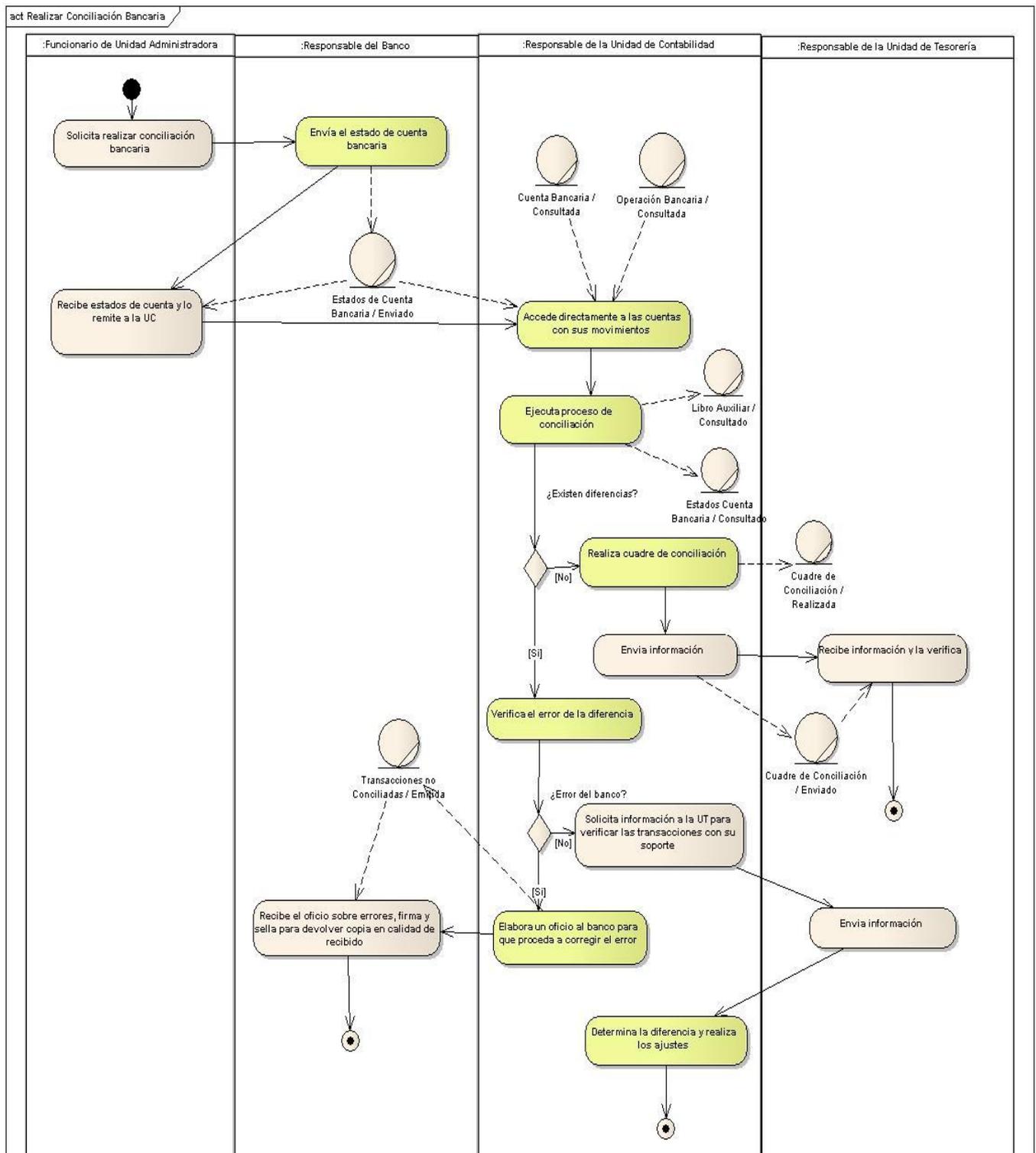


Figura 2.5: Diagrama de actividades correspondiente al caso de uso Realizar Conciliación Bancaria.

Solicitar Orden de Pago

Caso de Uso:	Solicitar Orden de Pago	
Actores:	Funcionario de la Unidad Administradora (UA)	
Resumen:	Este caso de uso se inicia cuando el Funcionario de la UA emite solicitud de pago para la cancelación de sus obligaciones por pagar.	
CU asociados:		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. Emite solicitud de pago.	2. La UT determina el tipo de solicitud de pago. Si es transferencia establece como se realizará la misma, sino Ver Flujo Alterno 1.	
	3. La UT determina si la transferencia es por Carta de Orden y genera dos ejemplares de la misma, sino Ver Flujo Alterno 2.	
	4. La UT solicita la firma de la Carta de Orden por parte del Tesorero y de la ACE.	
	5. Ambos firman el documento y lo envía a la UT.	
	6. La UT entrega los dos (2) ejemplares de la carta de orden y soportes al banco para su proceso.	
	7. El Banco firma los ejemplares y envía uno a la UT	
	8. La UT recibe el ejemplar y lo remite a la Unidad de Contabilidad. Terminando así el caso de uso.	
Flujos Alternos		
Flujo Alterno 1 “Solicitud de Pago por Cheque”		
Acción del Actor	Respuesta del Negocio	
	1. La UT emite cheques junto a sus comprobantes, tres ejemplares.	
	2. Verifica los datos emitidos en el cheque, en caso de ser correctos, emite una Relación de Cheques Emitidos por Cuenta Bancaria., sino Ver Flujo Alterno 3.	
	3. La UT solicita firma del Tesorero. Este firma y	

	remite cheques a la ACE para su firma.
	4. La ACE recibe, revisa y firma los cheques, remitiendo esto a la UT.
	5. La UT recibe cheques firmados y envía un ejemplar de la Relación de Cheques Emitidos por Cuenta Bancaria al Banco.
	6. El Banco recibe ejemplar de cheque.
	7. La UT entrega cheques, comprobantes y relación de cheques emitidos a la Unidad de Caja.
	8. La Unidad de Caja recibe cheque(s), comprobante(s) y la relación de cheques emitidos y los guarda en custodia en espera de su entrega al beneficiario.
	9. La UT emite un ejemplar de la Relación de Cheques Emitidos al Banco para efectos de control y seguridad. Terminando así el caso de uso.
Flujo Alterno 2 “La Transferencia no es por Carta de Orden”	
Acción del Actor	Respuesta del Negocio
	1. La UT genera documentos de realización de transferencia y solicita firma del Tesorero y de la ACE.
	2. Ambos firman documentos.
	3. La UT notifica al banco de la Transferencia firmada y anexa cualquier otra información necesaria para su culminación.
	4. El Banco envía Nota de Crédito y Nota de Débito a la UT una vez realizada la transferencia.
	5. La UT recibe notas de crédito y de débito y las registra en el libro auxiliar de la cuenta. Terminando así el caso de uso.
Flujo Alterno 3 “Los datos del cheque no son válidos”	

Acción del Actor	Respuesta del Negocio
	1. La UT procede a anular el cheque y actualiza el estado de los mismos registrándolos como anulados.
	2. Coloca sello de anulado en el original y copias del cheque y posteriormente lo archiva para su envío a la Unidad de Contabilidad.
	3. La UT emite nuevo(s) cheque(s). Continuar a partir del Paso 2 Flujo Alterno1 .
Prioridad:	Alta

Tabla 2.7: Descripción del caso de uso Solicitar Orden de Pago

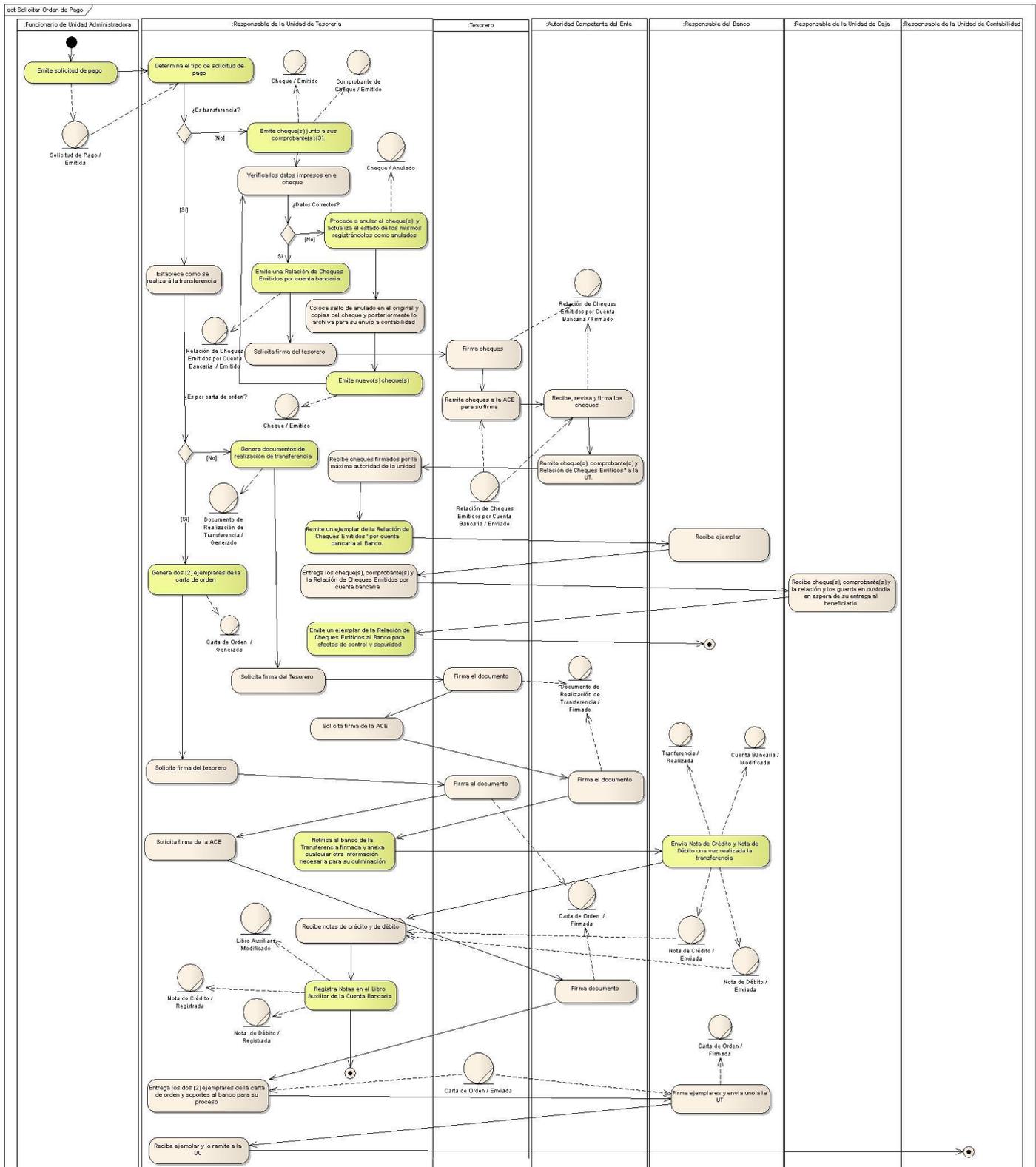


Figura 2.6: Diagrama de actividad correspondiente al caso de uso Solicitar Orden de Pago.

2.2.7. Modelo de Clases de Análisis del Negocio

A medida que se va llevando a cabo la realización de los casos de uso se van identificando los trabajadores y las entidades que participan en cada uno de ellos. El diagrama de clases de análisis del negocio como artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y como colaboran estos para llevar a cabo la realización del caso de uso.

Las entidades del negocio muestran un acercamiento a lo que serían las entidades del sistema, es decir aquellos objetos que se mantendrán persistentes en el sistema a desarrollar y de los cuales sería necesario conservar su información. (Booch, y otros, 2000)

Aunque se puede construir un único diagrama de objetos, se recomienda confeccionarlo para cada caso de uso del negocio para una mejor claridad. A continuación se muestran los diagramas de objetos correspondientes a los casos de uso más importantes.

Realizar Formulación y Reformulación del Flujo de Caja.

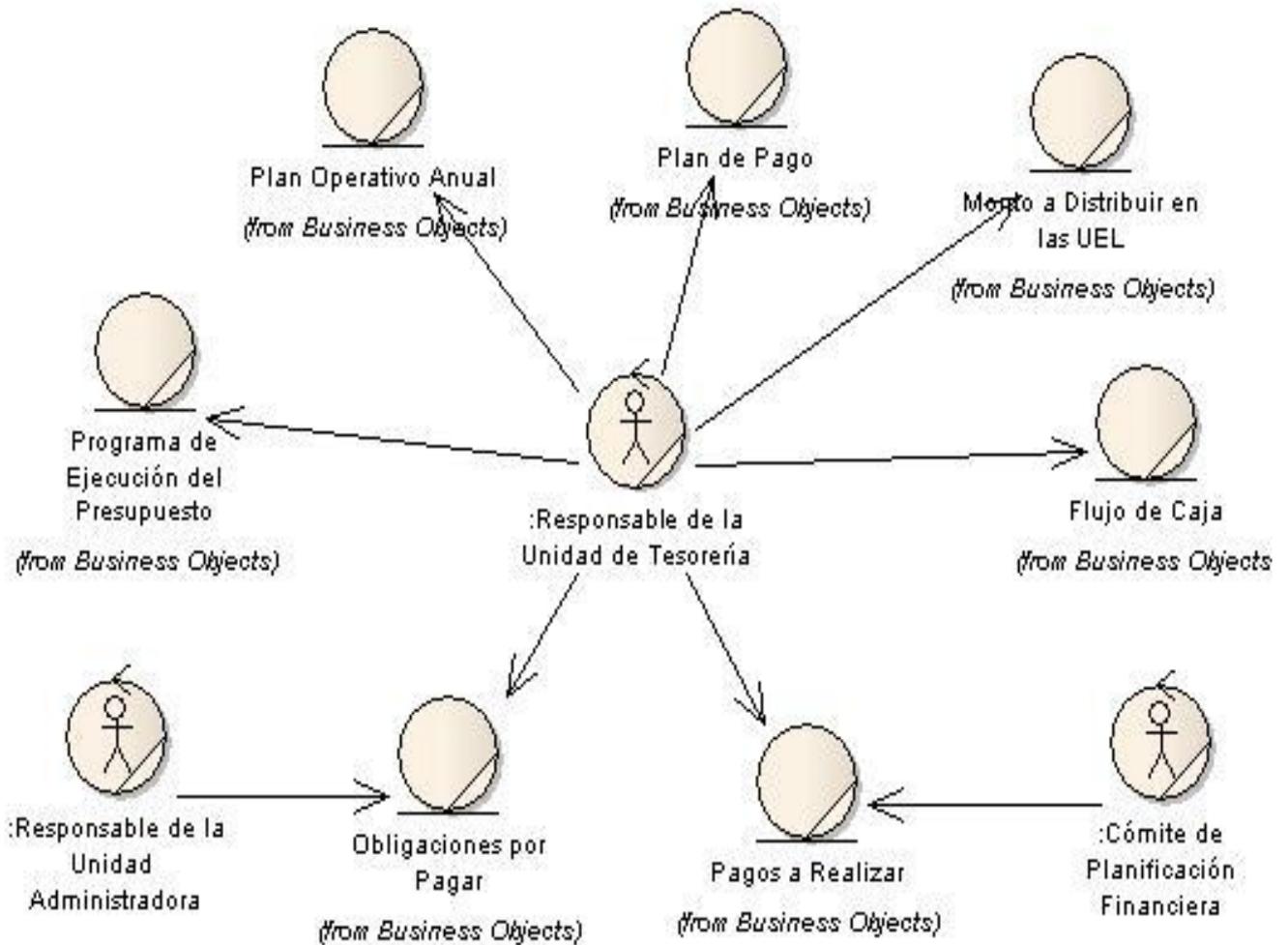


Figura 2.7: Diagrama de objetos del caso de uso Realizar Formulación y Reformulación del Flujo de Caja

Solicitar Apertura de Cuenta Bancaria.

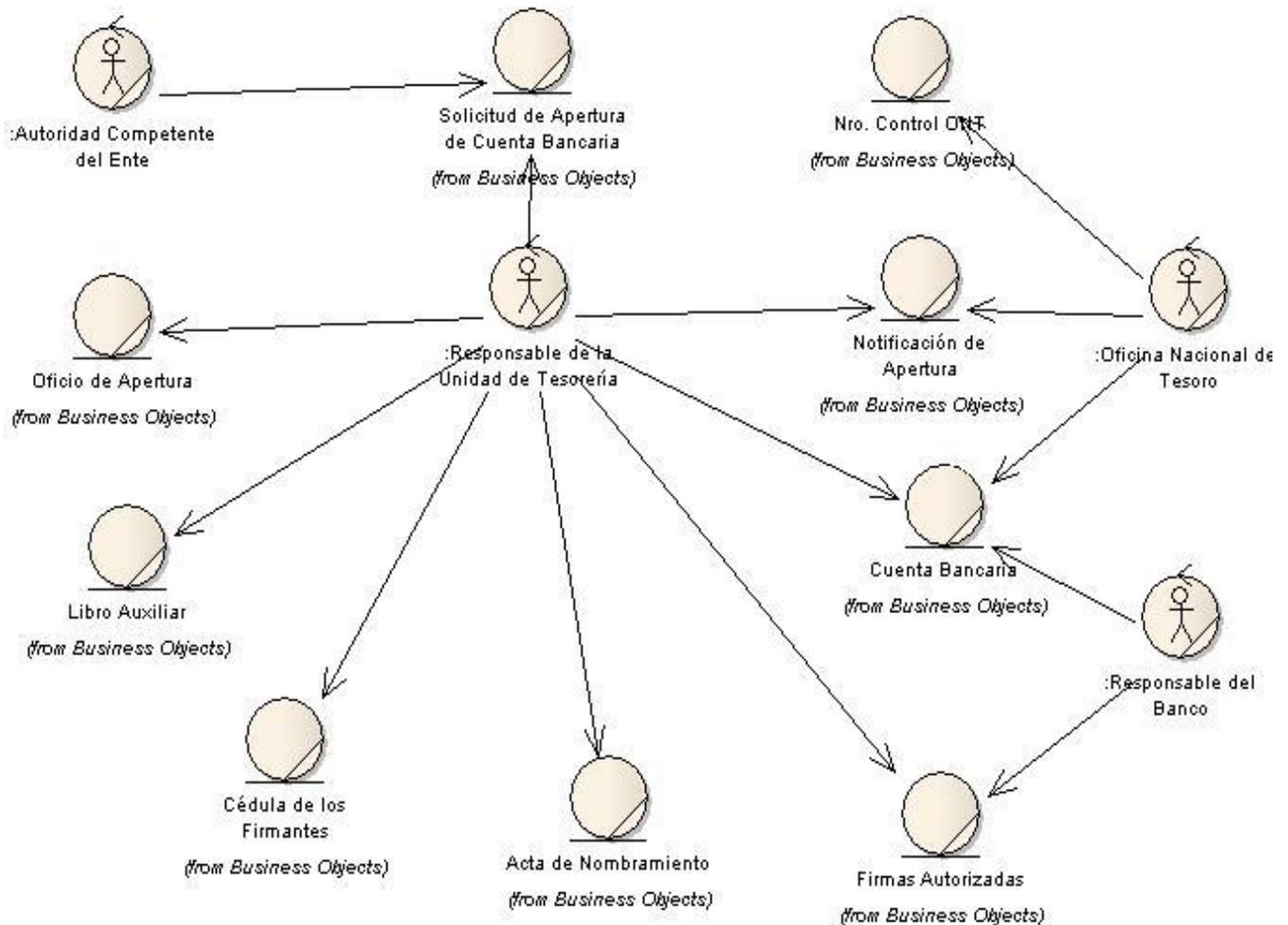


Figura 2.8: Diagrama de objetos correspondiente al caso de uso Solicitar Apertura de Cuenta Bancaria

Realizar Conciliación Bancaria.

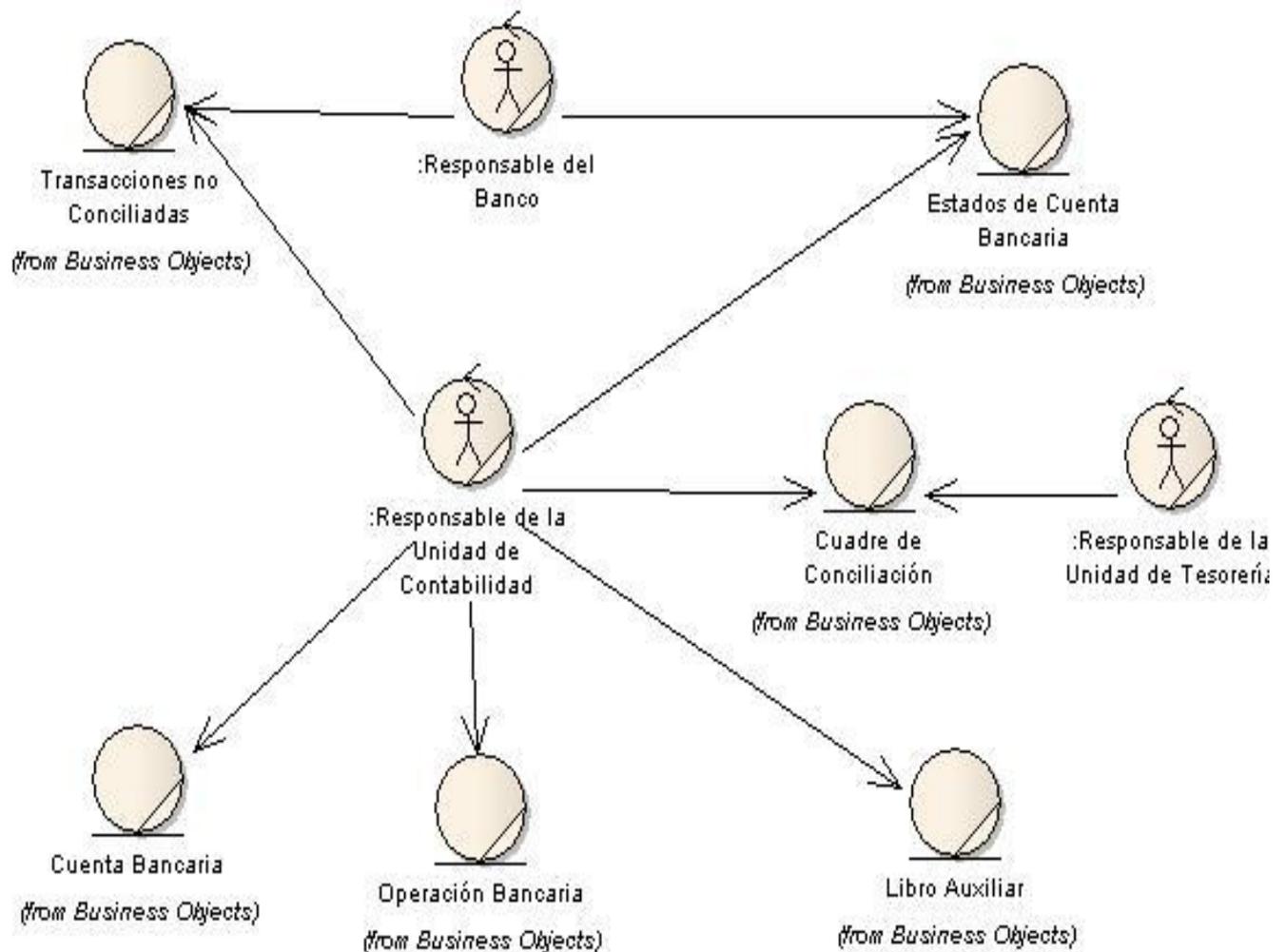


Figura 2.9: Diagrama de objetos correspondiente al caso de uso Realizar Conciliación Bancaria.

Diagrama de objetos Solicitar Orden de Pago.

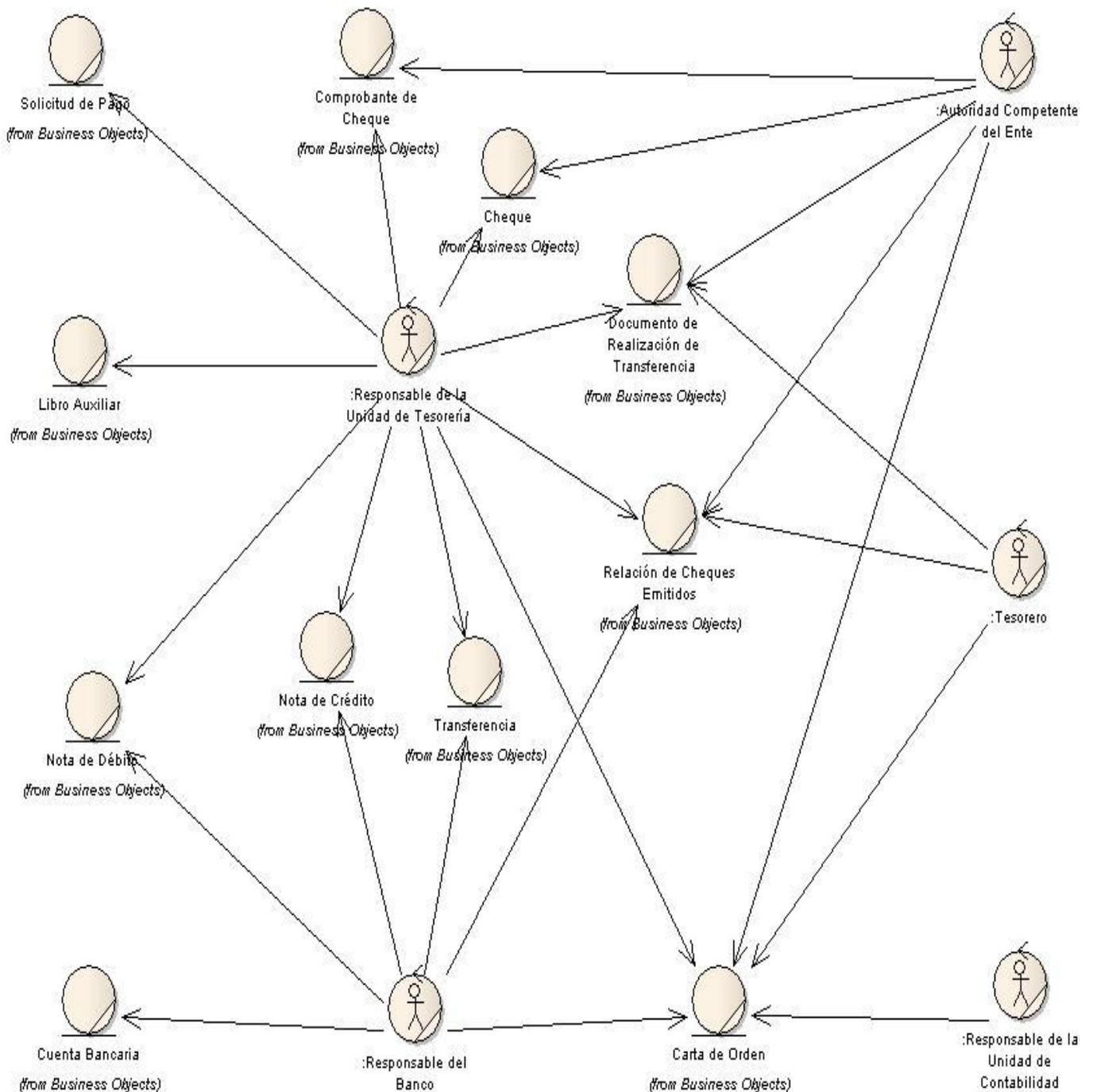


Figura 2.10: Diagrama de objetos correspondiente al caso de uso Solicitar Orden de Pago.

2.2.8. Reglas del Negocio

Las reglas del negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. El proceso de especificación implica que hay que “identificarlas” dentro del negocio, “evaluar” si son relevantes dentro del campo de acción que se está modelando e “implementarlas” en la propuesta de solución. (Booch, y otros, 2000)

Luego de haber estudiado los procesos del negocio de Tesorería se identificaron las siguientes reglas del negocio.

- El cheque solo puede ser entregado al beneficiario al cual va dirigido o a una persona autorizada por el mismo.
- Se autoriza la colocación financiera solo si existe disponibilidad financiera para llevarla a cabo.
- La chequera no se registra si esta contiene cheques anulados.
- La chequera no puede ser utilizada hasta que no se encuentre activa.
- Una cuenta bancaria no puede ser registrada hasta que no se autorice la apertura de la misma.

Conclusiones

Después de realizado el flujo de trabajo Modelación del Negocio, en el cual se realizó un estudio profundo del Negocio de Tesorería y de todos los procesos que en él se llevan a cabo, se realizaron las actividades correspondientes al mismo logrando desarrollar artefactos tales como: Casos de Uso, Actores, Trabajadores y Entidades del Negocio, Modelo de Casos de Uso, Modelo de Objetos, Reglas del Negocio y Realización de Casos de Uso, dando la posibilidad de identificar aquellas actividades que pueden ser automatizadas. Esto brinda una primera versión de lo que podrían ser los requisitos del sistema, con lo cual se refuerza la idea de que sea el propio negocio lo que determine los requisitos del sistema a desarrollar.

CAPÍTULO 3: REQUERIMIENTOS

La clave del éxito en la producción de un software es lograr una comunicación efectiva entre los usuarios y el equipo de proyecto con el objetivo de llegar a un entendimiento de lo que hay que hacer, es precisamente este el objetivo fundamental del flujo de trabajo que se desarrolla en este capítulo, el cual le da continuidad al flujo Modelación del Negocio desarrollado en el capítulo anterior. A partir de los resultados obtenidos en el capítulo anterior se desarrollan los artefactos correspondientes a este flujo de trabajo como son: Requerimientos de Software, Casos de Uso, los cuales se validan a través de la técnica de prototipo de interfaz de usuario y la aplicación de métricas orientadas a objetos.

3.1. Flujo de Trabajo de Requerimientos

En este flujo de trabajo a partir de las actividades automatizables identificadas en la modelación del negocio, se definen los requerimientos que tendrá el software a construir, los cuales se especifican a través de casos de uso, además se diseña un prototipo de interfaz de usuario mediante el cual se muestran y validan estos requisitos. A partir de ahí se define un sistema que responda a los intereses del negocio estudiado.

A continuación se muestran las actividades que se desarrollarán en este flujo de trabajo.

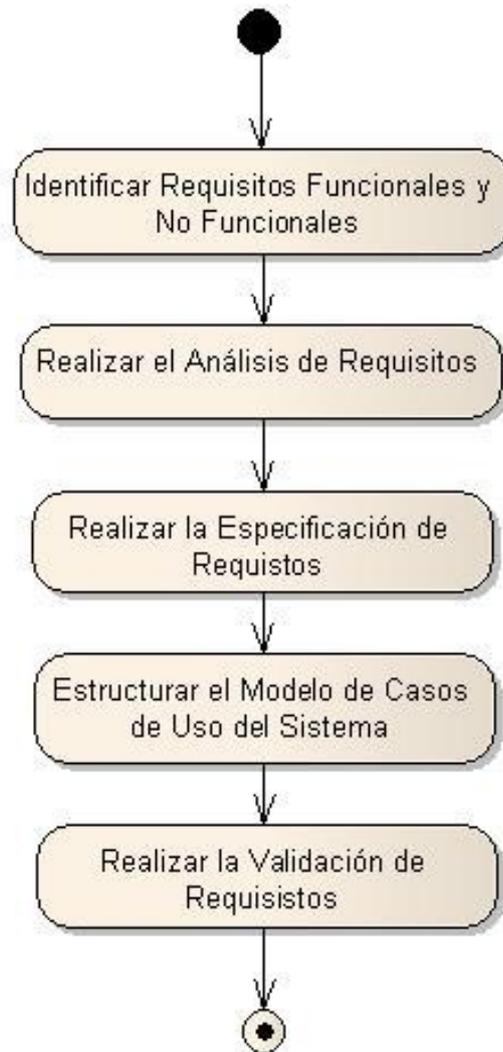


Figura 3.1: Flujo de actividades de Requerimientos

3.1.1. Objetivos del Flujo de Trabajo Requerimientos

- Definir el ámbito del sistema.
- Establecer y mantener un acuerdo entre clientes y otros involucrados sobre lo que el sistema debería hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requerimientos del sistema.
- Definir una interfaz de usuario para el sistema, enfocada a las necesidades y metas del usuario.

3.1.2. Requerimientos Funcionales y No Funcionales

Los Requerimientos son una condición o capacidad para resolver un problema o lograr un objetivo. Los mismos se clasifican en dos tipos: funcionales y no funcionales. Los Requisitos Funcionales representan capacidades que debe cumplir el sistema, y los No Funcionales definen las restricciones del mismo. Dentro de estos existen varias clasificaciones como por ejemplo: seguridad, soporte, interfaz, etc.

Requerimientos Funcionales

RF-1 Registrar una cuenta bancaria.

RF-2 Modificar una cuenta bancaria.

RF-3 Eliminar una cuenta bancaria.

RF-4 Seleccionar un tipo de fondo para una cuenta bancaria.

RF-5 Eliminar un tipo de fondo de una cuenta bancaria.

RF-6 Agregar las firmas autorizadas de una cuenta bancaria.

RF-7 Eliminar las firmas autorizadas de una cuenta bancaria.

RF-8 Modificar las firmas autorizadas de una cuenta bancaria.

RF-9 Cancelar una cuenta bancaria.

RF-10 Validar que la cuenta bancaria a cancelar cumpla con las siguientes condiciones:

- Que no tenga saldo.
- Que esté conciliada con el banco.
- Que no tenga documentos en tránsito.

RF-11 Realizar una búsqueda de las cuentas bancarias asociadas a un banco, un tipo de unidad, un tipo de cuenta y un tipo de fondo.

RF-12 Visualizar un reporte de la conciliación de una cuenta bancaria.

RF-13 Crear tipos de documentos.

RF-14 Modificar un tipo de documento.

RF-15 Eliminar un tipo de documento.

RF-16 Agregar talonarios.

RF-17 Modificar un talonario.

RF-18 Eliminar un talonario.

RF-19 Anular un documento que pertenece a un talonario.

RF-20 Realizar una búsqueda de todos los talonarios asociados a una cuenta bancaria, un tipo de documento y un período de fecha.

RF-21 Visualizar el estatus de los documentos del talonario.

RF-22 Calcular la cantidad de documentos de un talonario.

RF-23 Actualizar la cantidad de documentos emitidos, anulados y sin usar del talonario.

RF-24 Actualizar el estatus del talonario en dependencia de la cantidad de documentos que tenga sin usar.

RF-25 Actualizar los documentos al momento de usarlos para que se lleve un orden correlativo.

RF-26 Agregar un tipo de operación bancaria.

RF-27 Eliminar un tipo de operación bancaria.

RF-28 Modificar un tipo de operación bancaria.

RF-29 Generar un reporte que muestre los tipos de operaciones bancarias con las siguientes características:

- Número.
- Denominación de la operación.
- Tipo de documento.
- Contrapartida de la operación
- Origen/Destino del documento.

RF-30 Crear un criterio contable.

RF-31 Asociar a cada criterio contable los tipos de operaciones bancarias.

RF-32 Modificar un criterio contable.

RF-33 Eliminar un criterio contable.

RF-34 Realizar una búsqueda de las órdenes de pagos por id del beneficiario, o denominación del Beneficiario, un intervalo de fecha y el estatus.

RF-35 Aprobar una orden de pago.

RF-36 Anular una orden de pago.

RF-37 Visualizar un reporte donde se muestre por estatus, la orden de pago con sus facturas asociadas.

RF-38 Realizar una búsqueda de los ingresos por cuenta bancaria o denominación de la cuenta bancaria, por tipo de operación, por estatus del documento y en un intervalo de fecha predeterminado.

RF-39 Agregar un nuevo ingreso.

RF-40 Seleccionar el tipo de ingreso que se desea registrar.

RF-41 Eliminar un ingreso siempre que no haya sido confirmado.

RF-42 Modificar un ingreso que esté en estado de edición.

RF-43 Confirmar un ingreso que se encuentre en estado de edición.

RF-44 Validar que un documento de ingreso al confirmarse se active como conciliado si está reflejado como diferencias con el banco en alguno de los estados de cuenta.

RF-45 Anular un ingreso que se encuentre en estado de confirmación.

RF-46 Generar un reporte que muestre el número de la operación, el tipo de operaciones bancarias, la fecha y el monto por estatus y cuenta bancaria.

RF-47 Realizar una búsqueda de los gastos por cuenta bancaria o denominación de la cuenta bancaria, por tipo de operación, por estatus del documento y en un intervalo de fecha predeterminado.

RF-48 Agregar un nuevo gasto.

RF-49 Selecciona el tipo de gasto que se desea registrar.

RF-50 Eliminar un gasto siempre que no esté confirmado.

RF-51 Modificar un gasto que esté en estado de edición.

RF-52 Confirmar un gasto que se encuentre en estado de edición.

RF-53 Validar que un documento de gasto al confirmarse se active como conciliado si está reflejado como diferencias con el banco en alguno de los estados de cuenta.

RF-54 Anular un gasto que se encuentre en estado de confirmación.

RF-55 Generar un reporte que muestre el número de la operación, el tipo de operaciones bancarias, la fecha y el monto por estatus y cuenta bancaria.

RF-56 Editar o seleccionar los siguientes datos:

- Cuenta Bancaria y/o Denominación de la Cuenta Bancaria.
- Número del Documento.
- Fecha de Emisión.
- Monto.
- Orden de Pago.
- Descripción de la Operación

RF-57 Seleccionar el RIF y la denominación del proveedor.

RF-58 Mostrar las facturas asociadas a la orden de pago seleccionada.

RF-59 Realizar el asiento presupuestario.

RF-60 Generar los asientos contables correspondientes.

RF-61 Seleccionar el RIF y la denominación del Ente Recaudador.

RF-62 Agregar una retención por pagar.

RF-63 Seleccionar la factura que se desea pagar y editar el monto a pagar.

- RF-64** Eliminar una retención por pagar.
- RF-65** Mostrar el número del documento si se refiere a un tipo de documento con talonario.
- RF-66** Agregar una cuenta contable.
- RF-67** Eliminar una cuenta contable.
- RF-68** Modificar una cuenta contable.
- RF-69** Selecciona el carnet de identidad o el pasaporte y el nombre y apellidos de la persona.
- RF-70** Editar la denominación de otro origen.
- RF-71** Seleccionar la o las cuentas bancarias a la cual se le va a realizar la transacción y editar el monto de dicha transacción.
- RF-72** Agregar las partidas de egreso para realizar el asiento presupuestario.
- RF-73** Eliminar una partida de egreso.
- RF-74** Realizar una búsqueda de las partidas de egreso por el código o fragmento de la denominación de la partida.
- RF-75** Seleccionar las partidas de egreso que van a ser asociadas al tipo de operación para realizar el asiento presupuestario.
- RF-76** Seleccionar de cada partida de egreso el proyecto o la acción centralizada, la acción específica, la Unidad Ejecutora Local y el monto de dicha partida.
- RF-77** Agregar las partidas de recursos para realizar el asiento presupuestario.
- RF-78** Eliminar una partida de recurso.
- RF-79** Realizar una búsqueda de las partidas de recurso por el código o fragmento de la denominación de la partida.
- RF-80** Seleccionar las partidas de recursos que van a ser asociadas al tipo de operación para realizar el asiento presupuestario.
- RF-81** Seleccionar de cada partida de recurso el tipo de ingreso y el monto.
- RF-82** Realizar una búsqueda de los pagos anticipados por el Rif del proveedor o denominación del proveedor y un intervalo de fechas.
- RF-83** Mostrar las facturas por pagar relacionadas con el proveedor.
- RF-84** Asociar las facturas con el pago seleccionado para liquidarlo.
- RF-85** Generar un reporte que muestre los pagos anticipados de acuerdo a la búsqueda realizada.
- RF-86** Capturar los estados de cuentas recibidos de los diferentes bancos por cada cuenta bancaria.
- RF-87** Capturar el estado de cuenta siempre que el último estado de la cuenta bancaria esté validado.
- RF-88** Visualizar los detalles de un estado de cuenta.
- RF-89** Validar el último estado de cuenta para determinar si el mismo tiene descuadre o no.

- RF-90** Eliminar un estado de cuenta, solo si este es el último.
- RF-91** Modificar un estado de cuenta, solo si este es el último.
- RF-92** Realizar una búsqueda de los estados de cuenta asociados a una cuenta bancaria.
- RF-93** Editar los datos del estado de cuenta.
- RF-94** Capturar las diferencias que resulten de las operaciones relacionadas en el estado de cuenta.
- RF-95** Capturar las correcciones relacionadas con diferencias anotadas en estados de cuentas anteriores.
- RF-96** Eliminar y modificar las diferencias creadas a través de las correcciones.
- RF-97** Eliminar una corrección.
- RF-98** Asociar los documentos a conciliar con el estado de cuenta.
- RF-99** Eliminar los depósitos conciliados de la interfaz donde se seleccionan.
- RF-100** Realizar una búsqueda de las cuentas bancarias por banco y tipo de cuenta.
- RF-101** Generar un reporte donde se visualicen por cada banco o para todos los bancos y por tipo de cuenta, las cuentas bancarias con sus características.
- RF-102** Realizar una búsqueda de todas las operaciones asociadas a una cuenta bancaria en un intervalo de fechas.
- RF-103** Generar un reporte con los resultados obtenidos de la búsqueda realizada que muestre todas las operaciones contables por estatus, relacionadas con la cuenta bancaria seleccionada.
- RF-104** Realizar una búsqueda de las operaciones de ingreso por cuenta bancaria, tipo de operación, estatus y un rango de fechas.
- RF-105** Generar un reporte con los resultados obtenidos en la búsqueda realizada que muestre las operaciones de ingreso.
- RF-106** Realizar una búsqueda de las operaciones de gasto por cuenta bancaria, tipo de operación, estatus y un rango de fechas.
- RF-107** Generar un reporte con los resultados obtenidos en la búsqueda realizada que muestre las operaciones de gasto.
- RF-108** Realizar una búsqueda de todas las operaciones confirmadas asociadas a una cuenta bancaria y a un intervalo de fechas.
- RF-109** Generar un reporte que muestre todas las operaciones confirmadas contables de ingresos y gastos relacionadas con la cuenta bancaria seleccionada.
- RF-110** Realizar una búsqueda de las cuentas bancarias por banco, tipo de unidad, tipo de cuenta y tipo de fondo.

RF-111 Generar un reporte con los resultados obtenidos en la búsqueda realizada donde se muestre la conciliación de las cuentas bancarias.

RF-112 Visualizar reporte.

RF-113 Imprimir el reporte

RF-114 Exportar el reporte a formato PDF, XLS y DOC.

Requerimientos No Funcionales

Requerimientos de Software:

Los requerimientos de software vienen dados por las cualidades (atributos de calidad) que se quieren obtener con el sistema:

- El software tiene que ser flexible.
- El software tiene que ser escalable.
- El software tiene que ser seguro.

Requerimientos de Hardware:

Los requerimientos de hardware vienen dados por la arquitectura base de Registros y Notarías y comprende:

- El sistema tiene que interactuar con dispositivos de impresión (impresora LaserJet 2430).
- El sistema estará montado sobre una red WAN corporativa utilizando una Red Privada Virtual (VPN).
- El sistema se apoyará en una red de tecnología inalámbrica para el funcionamiento de las oficinas (Unidades Ejecutoras Locales).
- El sistema requiere de una memoria RAM, de 128 Mb para las estaciones simples, y hasta 512 Mb para las estaciones de captura y de impresión.

Requerimientos en el Diseño y la Implementación:

- Se utilizará Enterprise Architect como herramienta case para el modelado y obtención de los diferentes artefactos que requiere el software.
- El software estará programado en C# como lenguaje de programación.
- El software se desarrollará sobre Visual Studio .net específicamente con el framework 1.1.

- Se utilizará la herramienta Crystal Report para la generación de reportes.
- El sistema se desarrollará sobre un framework Común.
- Se utilizará el framework NHibernate v1.1 para la lógica de persistencia.
- Se utilizará el framework Spring.net v1.1 para la conexión entre las estructuras más significativas que lo componen.

Requerimientos de Apariencia o Interfaz Externa:

- El sistema tiene que ofrecer una interfaz amigable, fácil de operar.
- El sistema tiene que mantener la línea de diseño establecida para la institución que mantiene la uniformidad y representatividad de la misma.

Requerimientos de Seguridad:

- El sistema debe mantener en todo momento la seguridad de la información asegurando la autenticidad de la misma.
- La seguridad se establecerá por roles que se le asignarán a los usuarios que interactúen con el sistema.
- El software brindará solamente aquellas funcionalidades que competen a la Unidad Ejecutora donde esté implantado.
- El sistema mantendrá en todo momento las trazas que se corresponden con las diferentes situaciones críticas que puedan ocurrir.

Requerimientos de Usabilidad:

- Existirán servidores locales con capacidad necesaria para el procesamiento de las solicitudes del conjunto de aplicaciones de las diferentes oficinas.
- Las aplicaciones siempre solicitarán los datos a través del servidor local.
- Desde cada servidor local se establecerá la conexión con servidores centrales para mantener la actualización de los datos en ambos sentidos.

Requerimientos de Soporte:

- Los servidores a nivel central poseerán Oracle Enterprise Edición 10g R2 Real Application Cluster, y en los servidores locales de las oficinas se poseerá el Oracle Standard Edition ONE versión 10g R2.
- En las estaciones de trabajo el sistema estará montado sobre el sistema operativo Windows XP Sp2.
- Los servidores locales estarán montados sobre Windows Server 2003 y los centrales sobre RedHat AS v4.0.

3.1.3. Actores del Sistema

A continuación se muestra el actor identificado en el proceso de tesorería, el mismo representa a las personas que interactúan con el sistema que se modela.

Actores	Descripción
Responsable de la Unidad de Tesorería	Es la persona encargada de la Unidad de Tesorería ubicada en SAREN que controla la tesorería a nivel central.

Tabla 3.1: Descripción de los actores del sistema

3.1.4. Patrones de Casos de Uso

La experiencia en la identificación de los casos de uso ha evolucionado de manera creciente, donde se han definido un conjunto de patrones que permiten con más precisión reflejar los requisitos reales, haciendo más fácil el trabajo con los sistemas, y mucho más simple su mantenimiento. Dado un contexto y un problema a resolver, estas técnicas han mostrado ser la solución adoptada en la comunidad del desarrollo de software, donde las mismas se presentan a modo de herramientas que permiten resolver los problemas que se les planteen a los desarrolladores de una forma ágil y sistemática. Estos patrones se enfocan hacia el diseño y las técnicas utilizadas en modelos de alta calidad, y no en cómo modelar usos específicos. Utilizando estos patrones de casos de uso los arquitectos, analistas, ingenieros, y gerentes pueden lograr mejores resultados de forma más rápida.

¿Qué es un patrón?

Un patrón es una pareja de problema/solución con un nombre, que codifica (estandariza) buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades.

Se entiende por la palabra Patrón la siguiente definición: Solución común a un problema común de un determinado contexto. (Booch, y otros, 2000)

A continuación se muestran los patrones de casos de uso utilizados:

Reglas de Negocio

Se basan en la extracción de información originada de las políticas, reglas y regulaciones del negocio de la descripción del flujo y describe la información como una colección de reglas del negocio referenciadas a partir de las descripciones de los casos de uso.

Este patrón es aplicado a todos los casos de uso modelando los servicios que son afectados por las reglas del negocio definidas en la organización. Sin embargo, este patrón no influye en la estructura del modelo de casos de uso. Las reglas son descritas en un documento separado, referenciadas por las descripciones de los casos de usos relevantes. Este patrón es apropiado utilizarlo cuando no hay necesidad de cambiar dinámicamente las reglas del negocio mientras el sistema se este utilizando.

Concordancia (Commonality)

Extrae una subsecuencia de acciones que aparecen en diferentes lugares del flujo de casos de uso y es expresado por separado.

Reuso

Consta de 3 casos de uso. El primero llamado subsecuencia común, modela una secuencia de acciones que aparecerán en múltiples casos de uso en el modelo. Los otros casos de uso modelan el uso del sistema que comparte la subsecuencia común de acciones. De manera que deben existir al menos dos de ellos.

Adición

En el caso de este patrón alternativo, la subsecuencia común de casos de uso, extiende los casos de uso compartiendo la subsecuencia de acciones. Los otros casos de uso modelan el flujo que será expandido con la subsecuencia. Este patrón es preferible usarlo cuando otros casos de uso se encuentran propiamente completos, o sea, que no requieren de una subsecuencia común de acciones para modelar los usos completos del sistema.

Especialización

Otro patrón de concordancia que contiene casos de uso del mismo tipo. En este caso, estos son modelados como una especialización de casos de uso de tipo de uso común. Todas las acciones en estos casos de uso son heredadas por los casos de uso hijos, donde otras acciones serán adicionadas o acciones heredadas que serán especializadas. Este patrón es aplicable cuando la utilización de los casos de uso que han sido modelados son del mismo tipo, y este tipo debe hacerse visible en el modelo.

CRUD (Creating, Reading, Updating, Deleting)

Este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual. Este consta de un caso de uso, llamado **Gestionar información** modela todas las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación. Suele ser utilizado cuando todos los flujos contribuyen al mismo valor del negocio, y estos a su vez son cortos y simples. (ÖVERGAARD, y otros, 2004)

3.1.5. Casos de Uso del Sistema

Después de realizar la identificación y análisis de los requisitos estos fueron agrupados en casos de uso.

Nro.	Casos de Usos	Descripción
CU-1	Gestionar Cuentas Bancarias.	Este caso de uso consiste en definir las cuentas bancarias que se utilizarán en el sistema, las cuales van a ser definidas en la UT. Además se muestra la conciliación de las mismas.
CU-2	Visualizar Conciliación de la Cuenta Bancaria.	Permite visualizar un reporte con la conciliación de la cuenta bancaria.
CU-3	Gestionar Registro de Tipos de Documentos Bancarios.	Este caso de uso consiste en registrar en un codificador los distintos documentos de cobro o pago que puedan utilizarse en las operaciones relacionadas con el banco.
CU-4	Gestionar Registro de Talonarios.	Permite crear los talonarios que van a ser utilizados por cada cuenta bancaria.
CU-5	Gestionar Tipos de Operaciones Bancarias.	Consiste en registrar todos los tipos de operaciones bancarias que se utilizarán en el sistema.
CU-6	Gestionar Registro de	Este caso de uso consiste en registrar criterios contables

	Criterios de Contabilización de Banco.	para asociar a los tipos de operaciones bancarias.
CU-7	Gestionar Registro de Órdenes de Pago Solicitadas.	Este caso de uso consiste en registrar la relación de todas las órdenes de pagos solicitadas.
CU-8	Registrar Ingresos	Este caso de uso consiste en registrar todas las operaciones de ingreso para que las mismas sean contabilizadas.
CU-9	Registrar Gastos	Este caso de uso consiste en registrar todas las operaciones de gasto para que las mismas sean contabilizadas.
CU-10	Registrar Relación Proveedores - Obligaciones por Pagar.	Este caso de uso consiste en capturar los datos de la operación bancaria cuando el documento se emite a un proveedor que se le va a liquidar obligaciones por pagar.
CU-11	Registrar Relación Proveedores – Cuentas Contables.	Consiste en capturar los datos de la operación bancaria cuando el documento se emite o recibe de un proveedor y se relaciona en la contrapartida con cuentas contables.
CU-12	Registrar Relación Ente Recaudador – Retenciones por Pagar.	Este caso de uso consiste en registrar los datos de la operación bancaria cuando se emite un documento de pago para liquidar las retenciones por pagar relacionadas con los Entes Recaudadores.
CU-13	Registrar Relación Persona – Cuenta Contable.	Este caso de uso consiste en capturar los datos correspondientes a la operación bancaria que esté relacionada con personas, como trabajadores o clientes.
CU-14	Registrar Relación Otros – Cuenta Contable.	En este caso de uso se capturan los datos de la operación bancaria correspondiente relacionada con otros orígenes y destinos que no estén clasificados cuya contrapartida de la operación pueda ser cualquier cuenta contable.
CU-15	Registrar Relación Otros – Cuenta de la Institución.	Se captura la operación cuando se transfiere de una cuenta bancaria a otra dentro de la institución.
CU-16	Registrar Asiento Presupuestario del Gasto.	Este caso de uso consiste en registrar el asiento presupuestario relacionado con la operación bancaria de gasto.

CU-17	Registrar Asiento Presupuestario del Ingreso.	Este caso de uso consiste en registrar el asiento presupuestario relacionado con la operación bancaria de ingreso.
CU-18	Gestionar Estados de la Cuenta Bancaria.	Este caso de uso consiste en crear, modificar, eliminar y validar los estados de cuentas recibidos por el banco para cada una de las cuentas bancarias tanto de forma manual como automática.
CU-19	Capturar Estado de la Cuenta Bancaria.	Este caso de uso consiste en capturar las diferencias, correcciones y los documentos a conciliar de cada estado de cuenta que se cree.
CU-20	Visualizar Registro de Pagos Anticipados.	Este caso de uso consiste en visualizar la relación de todos los pagos anticipados que se han realizado a los proveedores a través de las operaciones de gasto.
CU-21	Visualizar Reporte de Cuentas Bancarias.	Consiste en mostrar un reporte donde se visualicen por cada banco o todos los bancos las características de las cuentas bancarias seleccionadas.
CU-22	Visualizar Registro de Ingresos	Este caso de uso permite visualizar un reporte de todas las operaciones de ingresos realizadas a una cuenta bancaria o a todas en un período de tiempo seleccionado.
CU-23	Visualizar Registro de Gastos	Este caso de uso permite visualizar un reporte de todas las operaciones de gastos realizadas a una cuenta bancaria o a todas en un período de tiempo seleccionado.
CU-24	Visualizar Reporte de Operaciones Bancarias.	Este caso de uso consiste en visualizar un reporte donde se muestre por cuenta bancaria todas las operaciones asociadas a la misma independientemente del estatus en el que se encuentren.
CU-25	Visualizar Reporte Libro Auxiliar de Bancos.	Este caso de uso consiste en visualizar un reporte donde se muestre por cuenta bancaria las operaciones confirmadas asociadas a la misma.
CU-26	Visualizar Reporte Conciliaciones de Cuentas Bancarias.	Este caso de uso consiste en generar un reporte donde se visualice la conciliación de las cuentas bancarias.

CU-27	Gestionar Reportes.	Permite generar un reporte y a la vez imprimirlo o exportarlo en determinado formato.
-------	---------------------	---

Tabla 3.2: Descripción de los casos de uso del sistema

3.1.6. Modelo de Casos de Uso del Sistema

Posteriormente se muestra como quedaron los casos de uso agrupados por paquetes según las funcionalidades que estos implementan.

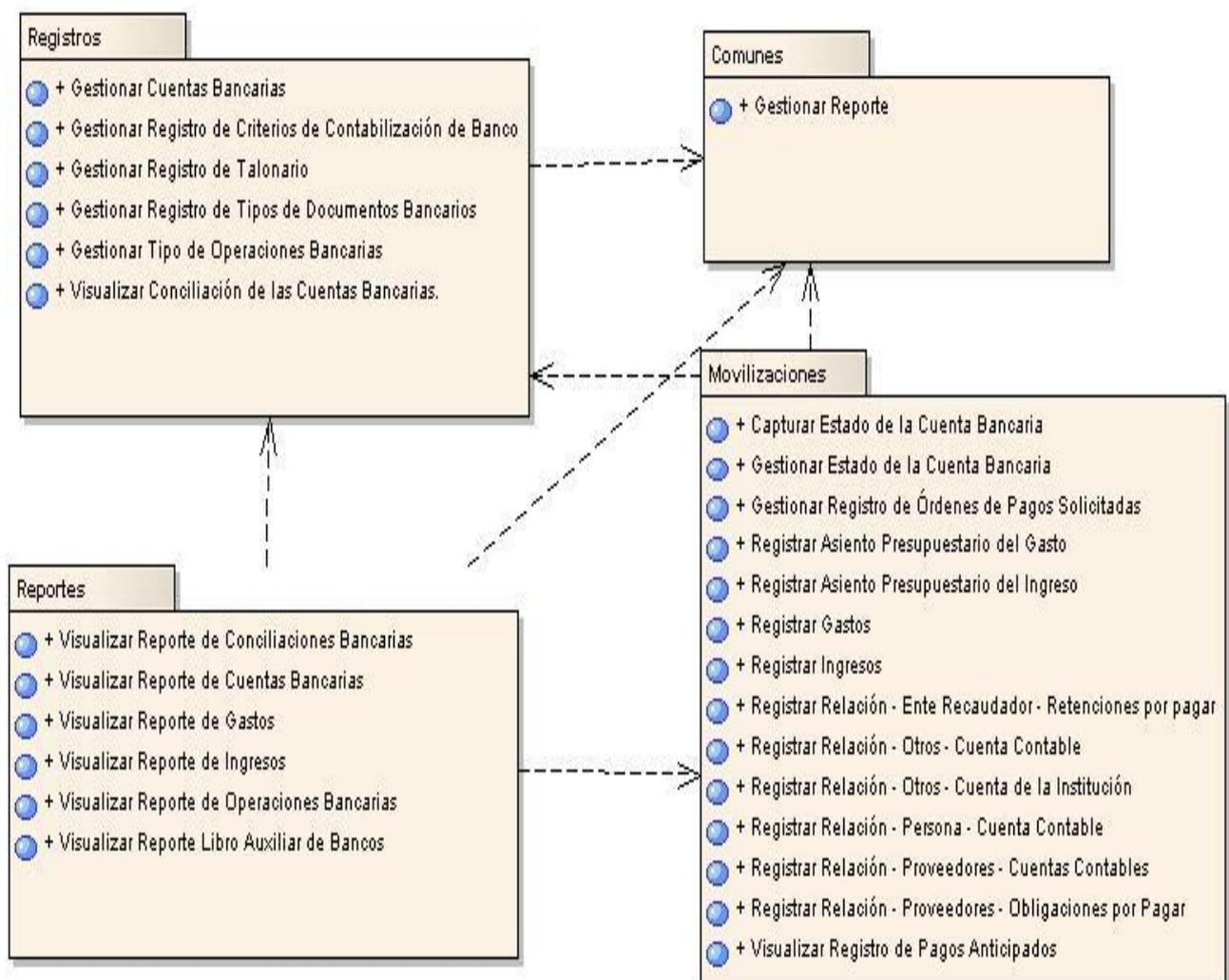


Figura 3.2: Diagrama de Paquetes del Sistema

Seguidamente se muestran los diagramas de casos de uso correspondiente a cada paquete.

Paquete Registros

En este paquete se encuentran aquellos casos de uso que son en su mayoría codificadores, los cuales son fundamentales para llevar a cabo el proceso de Tesorería.

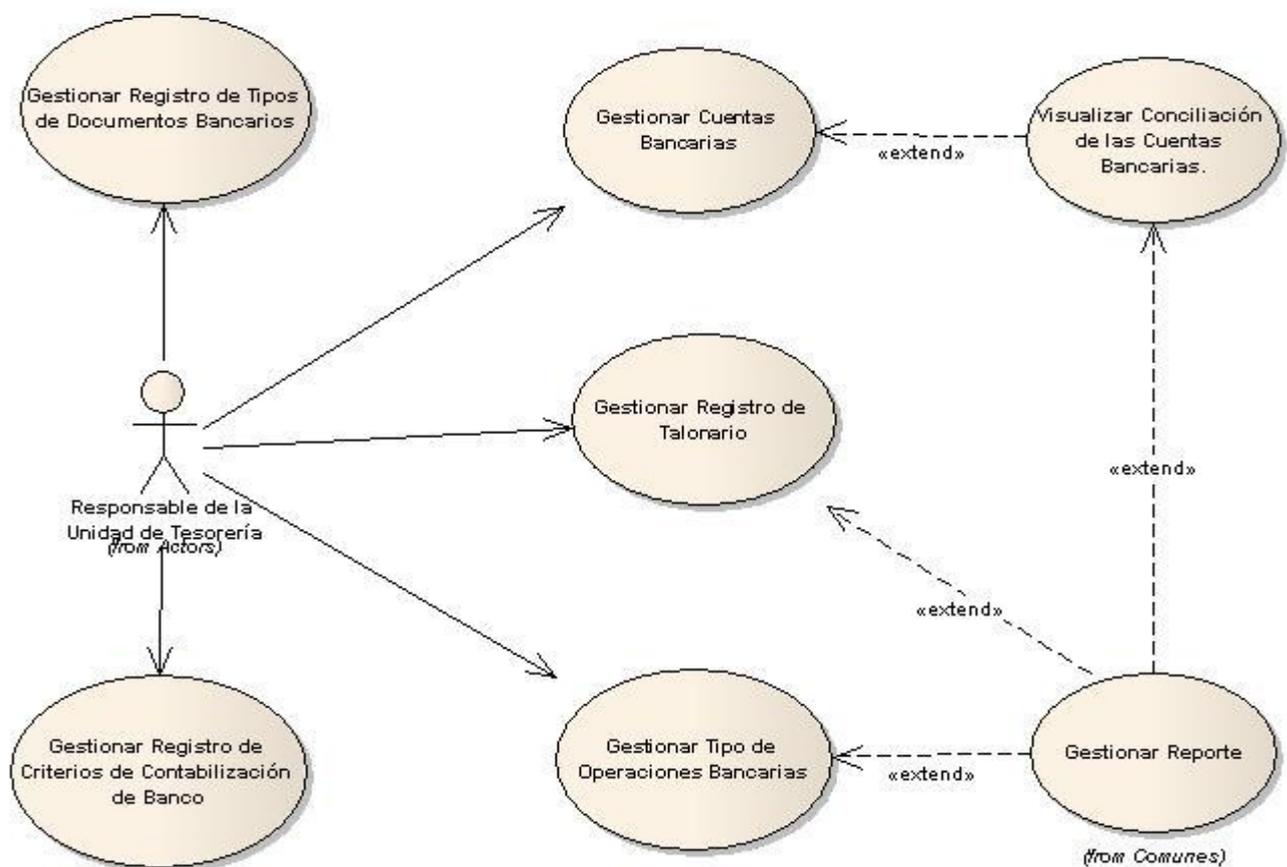


Figura 3.3: Diagrama de casos de uso correspondiente al Paquete Registros

Paquete Movilizaciones

En este paquete se encuentran aquellos casos de uso que registran el flujo de efectivo llevado a cabo en la tesorería.

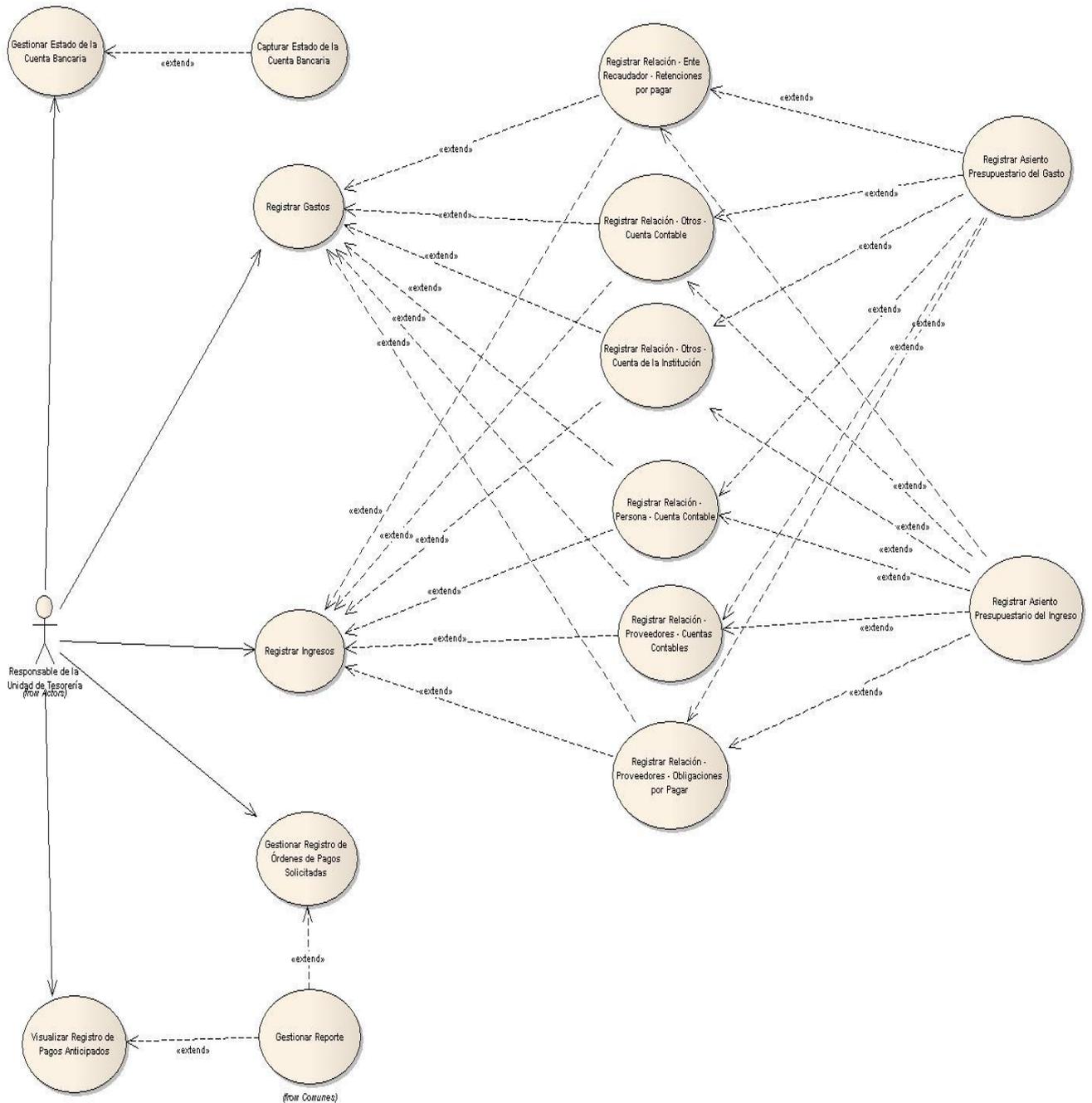


Figura 3.4: Diagrama de Casos de Uso correspondiente al Paquete Movilizaciones.

Paquete Reportes

En este paquete se encuentran los casos de uso donde se visualiza la información generada por los demás casos de uso a través de reportes.

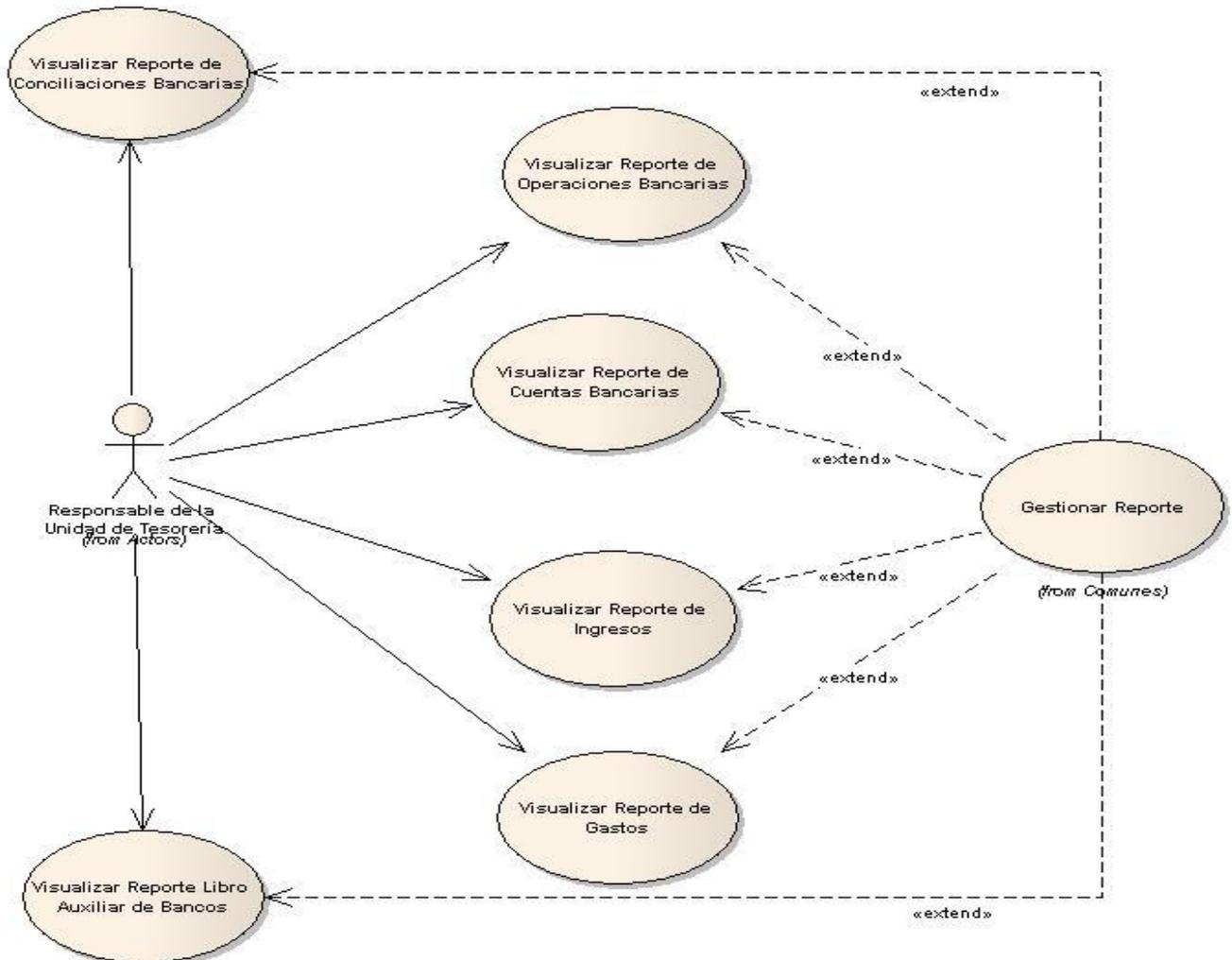


Figura 3.5: Diagrama de Casos de Uso correspondiente al Paquete Reportes

En el caso del paquete Comunes, este contiene un solo caso de uso el cual es común para los demás paquetes por lo que se encuentra en los restantes diagramas.

3.1.7. Realización de los Casos de Uso del Sistema

A continuación se describen los casos de uso más significativos del sistema con sus respectivos prototipos.

Gestionar Cuenta Bancaria

CU-1	Gestionar Cuentas Bancarias.
Actores:	Responsable de la Unidad de Tesorería.
Resumen:	Este caso de uso consiste en definir las cuentas bancarias que se utilizarán en el sistema, las cuales van a ser definidas en la UT. Además muestra la conciliación de las mismas.
Referencia:	RF-1 -- RF-11
CU asociados:	<ul style="list-style-type: none"> • Caso de Uso Extendido Visualizar Conciliación de la Cuenta Bancaria.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1 Ordena Gestionar las Cuentas Bancarias.	2 Muestra la interfaz Consulta y Registro de Cuentas Bancarias con los siguientes datos a seleccionar: <ul style="list-style-type: none"> • Banco. • Denominación del Banco. • Tipo de Unidad. • Tipo de Cuenta. • Tipo de Fondo. Y con las opciones: <ul style="list-style-type: none"> • Buscar. • Registrar. • Eliminar. • Modificar. • Cancelar. • Conciliar.
3 Selecciona los datos.	
4 Selecciona la opción Buscar.	5 Muestra una lista de cuentas bancarias de acuerdo a la

	<p>búsqueda realizada con los siguientes datos:</p> <ul style="list-style-type: none"> • Cuenta Bancaria. • Denominación. • Banco. • Tipo de Cuenta. • Código ONT. • Cancelada.
<p>6 Selecciona la opción Registrar Si selecciona una Cuenta Bancaria de la lista puede:</p> <ul style="list-style-type: none"> • Eliminar una Cuenta Bancaria Ver Sección 1. • Modificar una Cuenta Bancaria Ver Sección 2. • Cancelar una Cuenta Bancaria Ver Sección 6. • Conciliar una Cuenta Bancaria (Se llama al Caso de Uso Extendido Visualizar Conciliación de la Cuenta Bancaria). 	<p>7 Muestra la interfaz Cuenta Bancaria con los siguientes datos:</p> <ul style="list-style-type: none"> • Código de la Cuenta. • Denominación de la Cuenta Bancaria. • Código ONT. • Fecha de Apertura. • Tipo de Cuenta. • Tipo de Unidad. • Observaciones. • Cuenta Contable. • Denominación de la Cuenta Contable. • Saldo de Apertura. • Saldo de Libro. • Saldo en Banco. • Tipos de Fondo. <ul style="list-style-type: none"> ○ Código. ○ Tipo de Fondo. <p>Y las opciones:</p> <ul style="list-style-type: none"> • Seleccionar. • Eliminar. <ul style="list-style-type: none"> • Firmas Autorizadas. <ul style="list-style-type: none"> ○ Nombre(s) y Apellido(s). ○ Cédula de Identidad. ○ Tipo de Firma. ○ Fecha. <p>Y con las opciones:</p>

	<ul style="list-style-type: none"> • Agregar. • Eliminar. • Modificar.
8 Edita los datos.	
9 Selecciona la opción Seleccionar en la parte de Tipos de Fondo. <ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. Si selecciona un Tipo de Fondo de la lista puede: <ul style="list-style-type: none"> • Seleccionar la opción Eliminar Ver Sección 3. 	10 Muestra la interfaz Seleccionar Tipos de Fondo con el código y tipo de fondos y opción de seleccionarlos.
11 Selecciona los Tipos de Fondo a incluir.	
12 Selecciona la opción Aceptar. <ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. 	13 Acepta y actualiza la operación. Vuelve a la interfaz anterior y muestra los tipos de fondo seleccionados.
14 Selecciona la opción Agregar en la parte de Firma Autorizada. <ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. Si selecciona una Firma Autorizada de la lista puede: <ul style="list-style-type: none"> • Seleccionar la opción Eliminar Ver Sección 4. • Seleccionar la opción Modificar Ver Sección 5. 	15 Muestra la interfaz Datos de la Firma Autorizada con los datos a llenar: <ul style="list-style-type: none"> • Cédula de Identidad. • Nombre(s) y Apellido(s). • Tipo de Firma. • Fecha.
16 Selecciona los datos del Registro de Responsables.	
17 Selecciona la opción Aceptar.	18 Acepta y actualiza la operación, creando un responsable.

<ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. 	Vuelve a la interfaz anterior y muestra las Firmas Autorizadas actualizadas.
19 Selecciona la opción Aceptar. <ul style="list-style-type: none"> • Si selecciona opción Cancelar Ver Flujo Alterno 1. 	20 Acepta y actualiza la operación, creando una Cuenta Bancaria. Vuelve a la interfaz anterior y muestra las Cuentas Bancarias que han sido creadas.
21 Selecciona opción Cerrar	22 Cierra la interfaz y finaliza así el caso de uso.
Sección	
Sección 1 “Eliminar Cuenta Bancaria”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Eliminar.	2 Comprueba si la Cuenta Bancaria tiene operaciones asociadas, si las tiene no se puede eliminar. Continuar a partir del Paso 2 del Flujo Normal de los Eventos.
Sección 2 “Modificar Cuenta Bancaria”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Modificar.	Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 3 “Eliminar Tipos de Fondo”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Eliminar.	2 Elimina el Tipo de Fondo de la Cuenta Bancaria. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 4 “Eliminar Firma Autorizada”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Eliminar.	2 Elimina la Firma Autorizada de la Cuenta Bancaria. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 5 “Modificar Firma Autorizada”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Modificar.	2 Continuar a partir del Paso 15 del Flujo Normal de los Eventos.
Sección 6 “Cancelar la Cuenta Bancaria”	
Acción del Actor	Respuesta del Sistema
1 Selecciona la opción Cancelar	2 Muestra la interfaz Cancelar Cuenta Bancaria donde se

cuenta bancaria.	muestran los datos de la cuenta bancaria seleccionada, además muestra un mensaje indicando si está seguro de cancelar la cuenta.
3 Selecciona la opción Aceptar. • Si selecciona la opción Cancelar Ver Flujo Alterno 1.	4 Comprueba que la cuenta bancaria: no tenga saldo, esté conciliada con el banco, no tenga documentos en tránsito, si no cumple las verificaciones muestra información de que no puede cerrar la cuenta bancaria, en caso contrario, cancela la Cuenta Bancaria seleccionada. Continuar a partir del Paso 2 del Flujo Normal de los Eventos.
Flujos Alternos	
Flujo Alterno 1 "Cancelar"	
Acción del Actor	Respuesta del Sistema
	1 No realiza ningún cambio, cierra la interfaz correspondiente y regresa a la interfaz anterior.
Prioridad:	Alta

Tabla 3.3: Descripción del caso de uso Gestionar Cuenta Bancaria

Figura 3.8: Pantalla Seleccionar Tipos de Fondo

Figura 3.9: Pantalla Datos de la Firma Autorizada

Gestionar Registro de Tipos de Documentos Bancarios

CU-3	Gestionar Registro de Tipos de Documentos Bancarios.
Actores:	Responsable de la Unidad de Tesorería.
Resumen:	Este caso de uso consiste en registrar los distintos documentos de cobro o pago que puedan utilizarse en las operaciones relacionadas con el banco.
Referencia:	RF-13 – RF-15
CU asociados:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Ordena Gestionar los Tipos de Documentos Bancarios.	2. Muestra la interfaz Consulta y Registro de Tipos de Documentos Bancarios con una lista de documentos que han sido creados anteriormente con los siguientes datos: <ul style="list-style-type: none"> • Denominación.

	<ul style="list-style-type: none"> • Referencia Bancaria. • Talonario. • Ingreso. • Gasto. <p>Y con las opciones:</p> <ul style="list-style-type: none"> • Agregar. • Eliminar. • Modificar.
<p>3. Selecciona la opción Agregar.</p> <p>Si selecciona un tipo de documento de la lista puede:</p> <ul style="list-style-type: none"> • Seleccionar la opción Eliminar Ver Sección 1. • Seleccionar la opción Modificar Ver Sección 2. 	<p>4. Muestra la interfaz Tipo de Documento Bancario con los siguientes datos a editar:</p> <ul style="list-style-type: none"> • Denominación. • Abreviatura • Si utiliza o no talonario. • Tipo de Documento (Si es de ingreso o de gasto).
<p>5. Selecciona la opción Aceptar.</p> <ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. 	<p>6. Crea el tipo de documento y vuelve a la interfaz anterior y adiciona el documento a la lista. Continuar a partir del Paso 3 del Flujo Normal de los Eventos.</p>
<p>7. Selecciona la opción Cerrar.</p>	<p>8. Cierra la interfaz. Terminando así el caso de uso.</p>
Sección	
Sección 1 “Eliminar Tipo de Documento”	
Acción del Actor	Respuesta del Sistema
<p>1. Selecciona la opción Eliminar.</p>	<p>2. Comprueba si el tipo de documento tiene operaciones asociadas, si las tiene no se puede eliminar. Continuar a partir del Paso 2 del Flujo Normal de los Eventos.</p>
Sección 2 “Modificar Tipo de Documento”	
Acción del Actor	Respuesta del Sistema
<p>1. Selecciona la opción Modificar.</p>	<p>2. Continuar a partir del Paso 4 del Flujo Normal de los Eventos.</p>
Flujos Alternos	
Flujo Alterno 1 “Cancelar”	

Acción del Actor	Respuesta del Sistema
	1. No realiza ningún cambio. Cierra la interfaz correspondiente y vuelve a la interfaz anterior.
Prioridad:	Alta

Tabla 3.4: Descripción del caso de uso Gestionar Registro de Tipos de Documentos Bancarios

Figura 3.10: Pantalla Consulta y Registro de Tipos de Documentos Bancarios

Figura 3.11: Pantalla Tipo de Documento Bancario

Gestionar Tipo de Operaciones Bancarias

CU-5	Gestionar Tipos de Operaciones Bancarias.
Actores:	Responsable de la Unidad de Tesorería.
Resumen:	Consiste en registrar todos los tipos de operaciones bancarias que se utilizarán en el sistema.
Referencia:	RF-26 -- RF-29
CU asociados:	<ul style="list-style-type: none"> • Caso de uso extendido Gestionar Reportes.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Ordena Gestionar los Tipos de Operaciones Bancarias.	<p>2. Muestra la interfaz Consulta y Registro de Tipos de Operaciones Bancarias que visualiza los tipos de operaciones bancarias existentes con los siguientes datos:</p> <ul style="list-style-type: none"> • Número. • Denominación. • Tipo de Documento. • Contrapartida. • Origen/Destino. <p>Y las siguientes opciones:</p> <ul style="list-style-type: none"> • Agregar. • Eliminar. • Modificar.
<p>3. Selecciona la opción Agregar.</p> <ul style="list-style-type: none"> • Si selecciona un tipo de operación bancaria se puede: <ul style="list-style-type: none"> ✓ Seleccionar la opción Eliminar Ver Sección 1. ✓ Seleccionar la opción Modificar Ver Sección 2. • Si desea Visualizar un reporte donde se muestren todas las operaciones 	<p>4. Muestra la interfaz Tipo de Operación Bancaria para editar los siguientes datos:</p> <ul style="list-style-type: none"> • Denominación. • Criterio Contable. • Tipo de Documento. • Contrapartida de la Operación. • Origen/Destino de la Operación.

bancarias Ver Sección 3.	
5. Edita los datos correspondientes.	
6. Ordena aceptar la operación. <ul style="list-style-type: none"> • Si desea cancelar la operación Ver Flujo Alterno 1. 	7. Acepta la operación. Continuar a partir del Paso 2 del Flujo Normal de Eventos.
8. Selecciona la opción Cerrar.	9. Cierra la interfaz. Termina así el caso de uso.
Sección	
Sección 1 “Eliminar una Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Ordena eliminar la operación bancaria seleccionada.	2. Elimina la operación bancaria seleccionada. Continuar a partir del Paso 3 del Flujo Normal de Eventos.
Sección 2 “Modificar una Operación Bancaria”	
1. Ordena modificar la operación bancaria seleccionada. Continuar a partir del Paso 4 del Flujo Normal de Eventos.	
Sección 3 “Visualizar Reporte”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Visualizar.	2. Muestra un Reporte de Tipos de Operaciones Bancarias donde se visualizan todos los tipos de operaciones bancarias (Ver Caso de Uso Extendido Gestionar Reportes).
Flujos Alternos	
Flujo Alterno 1 “Cancelar la operación”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Cancelar.	2. Cancela la operación. Continuar a partir del Paso 2 del Flujo Normal de Eventos.
Prioridad:	Alta

Tabla 3.5: Descripción del caso de uso Gestionar Tipos de Operaciones Bancarias

Consulta y Registro de Operaciones Bancarias

Agregar Eliminar Modificar

Nro.	Denominación	Tipo de Documento	Contrapartida	Origen/ Destino

Ayuda Visualizar Cerrar

Figura 3.12 Pantalla Consulta y Registro de Operaciones Bancarias

Tipo de Operación Bancaria

Nro. Operación Denominación

Criterio Contable Tipo Documento

Contrapartida de la Operación Origen/Destino de la Operación

Ayuda Aceptar Cancelar

Figura 3.13: Pantalla Tipo de Operación Bancaria

Gestionar Registro de Órdenes de Pagos Solicitadas.

CU-7	Gestionar Registro de Órdenes de Pago Solicitadas.
Actores:	Responsable de la Unidad de Tesorería.
Resumen:	Este caso de uso consiste en registrar la relación de todas las órdenes de pagos solicitadas.
Referencia:	RF-34 – RF-37
CU asociados:	<ul style="list-style-type: none"> • Caso de Uso Extendido Gestionar Reportes.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Ordena Gestionar el Registro de de Órdenes de Pago Solicitadas.	2. Muestra la interfaz Consulta y Registro de Órdenes de Pago Solicitadas con los siguientes datos a seleccionar: <ul style="list-style-type: none"> • Tipo de Beneficiario • ID del Beneficiario. • Denominación del Beneficiario. • Fecha Desde. • Fecha Hasta. • Estatus Y la opción: <ul style="list-style-type: none"> • Buscar. • Aprobar. • Anular. • Visualizar.
3. Selecciona los datos.	
4. Selecciona la opción Buscar. <ul style="list-style-type: none"> • Si selecciona la opción Aprobar Ver Sección 1. • Si selecciona la opción Anular Ver Sección 2. • Si selecciona la opción Visualizar Ver Sección 3. 	5. Visualiza las órdenes de pago de acuerdo a la selección realizada.
6. Ordena cerrar la interfaz.	7. Cierra la interfaz. Terminando así el caso de uso.

Sección	
Sección 1 “Aprobar Orden de Pago”	
1. Selecciona la opción Aprobar.	2. Muestra la interfaz Aprobación de la Orden de Pago para que se acepte o no la operación.
3. Ordena aceptar la operación. <ul style="list-style-type: none"> • Si selecciona la opción cancelar Ver Flujo Alterno 1. 	4. Acepta la operación. Continuar a partir del Paso 2 del Flujo Normal de los Eventos.
Sección 2 “Anular Orden de Pago”	
1. Selecciona la opción Anular.	2. Muestra la interfaz Anulación de la Orden de Pago con los siguientes datos para editar o seleccionar: <ul style="list-style-type: none"> • Explicación de la Anulación.
3. Edita o selecciona los datos. Continuar a partir del Paso 4 de la Sección 1.	
Sección 3 “Visualizar Reporte”	
1. Selecciona la opción Visualizar.	2. Visualiza el Reporte de Órdenes de Pago Solicitadas y llama Caso de Uso Gestionar Reporte. Los datos del reporte son los siguientes: <ul style="list-style-type: none"> • Denominación de la UA. • Código de la UA. • Fecha Desde. • Fecha Hasta. • Estatus. <ul style="list-style-type: none"> ○ No. de la Orden de Pago. ○ ID del Beneficiario. ○ Denominación del beneficiario. ○ Fecha. ○ Monto Total. ○ Monto Retenido. ○ Monto a Pagar. • Total por estatus. Continuar a partir del Paso 2 del Flujo Normal de los Eventos.

Registrar Ingresos

CU-8	Registrar Ingresos.	
Actores:	Responsable de la Unidad de Tesorería	
Resumen:	Este caso de uso consiste en registrar todas las operaciones de ingreso para que las mismas sean contabilizadas.	
Referencia:	RF-38 –RF-46	
CU asociados:	<ul style="list-style-type: none"> • Caso de Uso Extendido Registrar Relación Proveedores - Cuentas Contables. • Caso de Uso Extendido Registrar Relación Persona - Cuenta Contable. • Caso de Uso Extendido Registrar Relación Otros - Cuenta Contable. • Caso de Uso Extendido Registrar Relación Otros - Cuenta de la Institución. • Caso de Uso Extendido Registrar Relación Proveedores - Obligaciones por Pagar. • Caso de Uso Extendido Registrar Relación Ente Recaudador - Retenciones. 	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. Selecciona Registrar Ingresos.	2. Muestra la interfaz Consulta y Registro de Ingresos con datos a seleccionar: <ul style="list-style-type: none"> • Cuenta Bancaria. • Denominación de la Cuenta Bancaria. • Tipo de Operación. • Estatus. <ul style="list-style-type: none"> ✓ Edición. ✓ Confirmada. ✓ Anulada. • Fecha Desde. • Fecha Hasta. 	

	<p>Y las opciones:</p> <ul style="list-style-type: none"> • Buscar. • Agregar • Eliminar • Modificar • Confirmar • Anular • Visualizar.
<p>3. Selecciona Código o Denominación de la Cuenta Bancaria, el Tipo de Operación, Estatus, Fecha Desde y Fecha Hasta.</p>	<p>4. Muestra Denominación o Código de la Cuenta Bancaria en dependencia a lo seleccionado.</p>
<p>5. Selecciona la opción Buscar.</p>	<p>6. Muestra lista de Tipos de Operaciones Bancarias con los datos:</p> <ul style="list-style-type: none"> • Número. • Operación Bancaria. • Fecha. • Monto • Estatus.
<p>7. Selecciona la opción Agregar. Si selecciona un Tipo de Operación Bancaria de la lista puede:</p> <ul style="list-style-type: none"> • Seleccionar la opción Eliminar Ver Sección 1. • Seleccionar la opción Modificar Ver Sección 2. • Seleccionar la opción Confirmar Ver Sección 3. • Seleccionar la opción Anular Ver Sección 4. • Si selecciona la opción Visualizar Ver Sección 5. 	<p>8. Muestra la interfaz Tipos de Operaciones de Ingresos, donde se cargan todos los tipos de ingresos.</p>
<p>9. Selecciona un Tipo de Ingreso de la lista.</p>	

<ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. 	
10. Selecciona la opción Aceptar.	11. Muestra la interfaz de relación de acuerdo a la contrapartida de la operación, origen y destino del documento asociados al tipo de operación bancaria seleccionada, las mismas pueden ser: <ul style="list-style-type: none"> • Ver Caso de Uso Extendido Registrar Relación Proveedores - Cuentas Contables. • Ver Caso de Uso Extendido Registrar Relación Persona - Cuenta Contable. • Ver Caso de Uso Extendido Registrar Relación Otros - Cuenta Contable. • Ver Caso de Uso Extendido Registrar Relación Otros - Cuenta de la Institución. • Caso de Uso Extendido Registrar Relación Ente Recaudador - Retenciones. • Caso de Uso Extendido Registrar Relación Proveedores - Obligaciones por Pagar.
12. Selecciona la opción Cerrar.	13. Cierra la interfaz. Terminando así el caso de uso.
Sección	
Sección 1 “ Eliminar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Eliminar.	2. Comprueba si el Tipo de Operación Bancaria tiene operaciones asociadas, si las tiene no se puede eliminar sino se elimina. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 2 “Modificar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Modificar.	Continuar a partir del Paso 2 de la Sección 6.
Sección 3 “Confirmar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema

1. Selecciona la opción Confirmar.	2. Muestra lista actualizada de operaciones bancarias al cambiar su estatus y genera un asiento contable. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 4 “Anular Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
	1. Comprueba que el tipo de operación bancaria este en estado confirmado, si esta activa el botón Anular, sino no.
2. Selecciona la opción Anular.	3. Anula el tipo de operación bancaria que esta en confirmado revirtiendo esta operación. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 5 “Visualizar”	
Acción del Actor	Respuesta del Sistema
	1. Visualiza el Reporte de Ingresos y llama Caso de Uso Gestionar Reportes. Los datos del reporte son los siguientes: <ul style="list-style-type: none"> • Fecha Desde. • Fecha Hasta. • Cuenta Bancaria. • Denominación de la Cuenta Bancaria. <ul style="list-style-type: none"> • Tipo Operación. • Número de Operación. • Fecha. • Monto. • Estatus. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Flujos Alternos	
Flujo Alternativo 1 “Cancelar”	

Tipos de Operaciones de Ingreso	
Tipos de Operaciones de Ingreso	
Recaudación	
Intereses bancarios	
Transferencias recibidas	

Figura 3.16: Pantalla Tipos de Operaciones de Ingreso

Registrar Gastos.

CU-9	Registrar Gastos.
Actores:	Responsable de la Unidad de Tesorería
Resumen:	Este caso de uso consiste en registrar todas las operaciones de gasto para que las mismas sean contabilizadas.
Referencia:	RF-47 – RF-55
CU asociados:	<ul style="list-style-type: none"> • Caso de Uso Extendido Registrar Relación Proveedores - Obligaciones por pagar. • Caso de Uso Extendido Registrar Relación Proveedores - Cuentas Contables. • Caso de Uso Extendido Registrar Relación Ente Recaudador - Retenciones. • Caso de Uso Extendido Registrar Relación Persona - Cuenta Contable. • Caso de Uso Extendido Registrar Relación Otros - Cuenta Contable. • Caso de Uso Extendido Registrar Relación Otros - Cuenta de la Institución.

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona Registrar Gastos.	<p>2. Muestra la interfaz Consulta y Registro de Gastos con datos a seleccionar:</p> <ul style="list-style-type: none"> • Cuenta Bancaria. • Denominación de la Cuenta Bancaria. • Tipo de Operación. • Estatus. <ul style="list-style-type: none"> ✓ Edición. ✓ Confirmada. ✓ Anulada. • Fecha Desde. • Fecha Hasta. <p>Y las opciones:</p> <ul style="list-style-type: none"> • Buscar • Agregar • Eliminar • Modificar • Confirmar • Anular • Visualizar.
3. Selecciona Código o Denominación de la Cuenta Bancaria, el Tipo de Operación, Estatus, Fecha Desde y Fecha Hasta.	4. Muestra Denominación o Código de la Cuenta Bancaria en dependencia a lo seleccionado.
5. Selecciona la opción Buscar.	<p>6. Muestra lista de Tipos de Operaciones Bancarias con los datos:</p> <ul style="list-style-type: none"> • Número. • Tipo de Operación Bancaria. • Fecha. • Monto • Estatus.

<p>7. Selecciona la opción Agregar. Si selecciona un Tipo de Operación Bancaria de la lista puede:</p> <ul style="list-style-type: none"> • Seleccionar la opción Eliminar Ver Sección 1. • Seleccionar la opción Modificar Ver Sección 2. • Seleccionar la opción Confirmar Ver Sección 3. • Seleccionar la opción Anular Ver Sección 4. • Si selecciona la opción Visualizar Ver Sección 5. 	<p>8. Muestra la interfaz Tipos de Operaciones de Gastos, donde se cargan todos los tipos de gastos.</p>
<p>9. Selecciona un Tipo de Gasto de la lista.</p> <ul style="list-style-type: none"> • Si selecciona la opción Cancelar Ver Flujo Alterno 1. 	
<p>10. Selecciona la opción Aceptar.</p>	<p>11. Muestra la interfaz de relación de acuerdo a la contrapartida de la operación y origen y destino del documento asociados al tipo de operación bancaria seleccionada, las mismas pueden ser:</p> <ul style="list-style-type: none"> • Ver Caso de Uso Extendido Registrar Relación Proveedores - Obligaciones por pagar. • Ver Caso de Uso Extendido Registrar Relación Proveedores - Cuentas Contables. • Ver Caso de Uso Extendido Registrar Relación Ente Recaudador - Retenciones. • Ver Caso de Uso Extendido Registrar Relación Persona - Cuenta Contable. • Ver Caso de Uso Extendido Registrar Relación Otros - Cuenta Contable. • Ver Caso de Uso Extendido Registrar Relación Otros - Cuenta de la Institución.

12. Selecciona la opción Cerrar.	13. Cierra la interfaz. Terminando así el caso de uso.
Sección	
Sección 1 “ Eliminar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Eliminar.	2. Comprueba si el Tipo de Operación Bancaria tiene operaciones asociadas, si las tiene no se puede eliminar sino se elimina. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 2 “Modificar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Modificar.	Continuar a partir del Paso 2 de la Sección 6.
Sección 3 “Confirmar Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Confirmar.	2. Muestra lista actualizada de operaciones bancarias al cambiar su estatus y genera un asiento contable. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 4 “Anular Tipo de Operación Bancaria”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Anular.	2. Comprueba que el tipo de operación bancaria este en estado confirmado, si esta activa el botón Anular, sino no.
	3. Anula el tipo de operación bancaria que esta en confirmado revirtiendo esta operación. Continuar a partir del Paso 7 del Flujo Normal de los Eventos.
Sección 5 “Visualizar”	
Acción del Actor	Respuesta del Sistema
	1. Visualiza el Reporte de Gastos y llama al Caso de Uso Gestionar Reportes.

	<p>Los datos del reporte son los siguientes:</p> <ul style="list-style-type: none"> • Fecha Desde. • Fecha Hasta. • Cuenta Bancaria. • Denominación de la Cuenta Bancaria. <ul style="list-style-type: none"> • Tipo Operación. • Número de Operación. • Fecha. • Monto. • Estatus. <p>Continuar a partir del Paso 7 del Flujo Normal de los Eventos.</p>
Flujos Alternos	
Flujo Alternos 1 “Cancelar”	
Acción del Actor	Respuesta del Sistema
	1. Cierra la interfaz correspondiente y regresa a la interfaz anterior.
Prioridad:	Alta

Tabla 3.8: Descripción del caso de uso Registrar Gastos

Consulta y Registro de Gastos

Cuenta Bancaria Denominación de la Cuenta Bancaria

Tipo de Operación Estatus

Fecha Desde Fecha Hasta

Número	Tipo de Operaciones Bancarias	Fecha	Monto	Estatus
				Edición
				Confirmada
				Anulada

Figura 3.17: Pantalla Consulta y Registro de Gastos

Tipos de Operaciones de Gastos

Tipos de Operaciones de Gastos
Pago a proveedores
Reposición de caja chica
Transferencias emitidas

Figura 3.18: Pantalla Tipos de Operaciones de Gastos

3.2. Validación de los Resultados Obtenidos

3.2.1. Validación Mediante Prototipo

Los prototipos surgen para validar los requerimientos identificados, estos son simulaciones del posible producto, que luego son utilizados por el usuario final, permitiendo conseguir una importante retroalimentación en cuanto a si el sistema diseñado en base a los requerimientos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva. El prototipo no funcional da una idea de cómo quedará el sistema a desarrollar, sobre este se realizan evoluciones y modificaciones hasta determinar claramente los requisitos.

El prototipo no funcional del módulo de Tesorería fue desarrollado con la herramienta Microsoft Office Visio 2003, donde para cada caso de uso del sistema descrito se crearon un grupo de pantallas donde refleja el flujo que se lleva a cabo en los mismos, permitiendo verificar los requisitos que se cumplen en cada una de ellas. Además el módulo cuenta con el Documento Prototipo de Interfaz de Usuario donde se describen las funcionalidades de cada una de las interfaces correspondientes a cada caso de uso.

3.2.2. Aplicación de Métricas

La medición es fundamental para cualquier disciplina de ingeniería, y la ingeniería del software no es una excepción. La medición permite tener una visión más profunda proporcionando un mecanismo para la evaluación objetiva. (Pressman, 2002)

En la actualidad en el proceso de desarrollo de software existen un conjunto de métricas, las cuales se utilizan para la validación de los requisitos identificados en la realización de un software, estas métricas permiten a los analistas validar de una manera correcta que los requisitos identificados durante el proceso de desarrollo, tienen la calidad requerida y cumplen con las normas internacionales. Además existen métricas que permiten validar también el modelo de casos de uso. En este epígrafe se mencionarán algunas de las métricas que fueron aplicadas a los requisitos de software y al modelo de casos de uso del sistema del módulo de Tesorería del sistema de Administración Financiera de SAREN.

Métricas de la calidad de la especificación

$$R_t = R_f + R_{nf}$$

$$140 = 114 + 26$$

Donde:

Rt: Total de requisitos.

Rf: Cantidad de requisitos funcionales.

Rnf: Cantidad de requisitos no funcionales.

Para determinar la especificidad (ausencia de ambigüedad) de los requisitos. Davis (Davis, 1993) sugiere una métrica basada en la consistencia de la interpretación de los revisores para cada requisito: Se calcula Q1 para determinar la especificidad de los requisitos.

Q1 = R_{ui} / R_t

0.97 = 136/140

Donde:

R_{ui}: Número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas.

Q1: Ausencia de ambigüedad.

Grupo de Revisores que aplicaron esta métrica:

- Ing. Yosvany Márquez
- Lic. Ismary Lorenzo
- Ing. Yaumarys Pino
- Ing. Lourdes Julia Perojo

Cuanto más cerca de 1 esté el valor de Q1, menor será la ambigüedad de la especificación.

El valor de Q1 = 0.97, esto demuestra que los requisitos se encuentran con un alto nivel de especificidad.

Modelo de Métricas para Análisis y Diseño Orientado a Objetos Basado en UML

El modelo tiene por objetivo medir la calidad de los productos intermedios generados en un proyecto de software. El modelo define cuatro atributos genéricos de propiedades de calidad: consistencia, correctitud, completitud y complejidad, que tienen un significado concreto de acuerdo al tipo de artefacto software y al nivel de abstracción que éste describe. Un atributo se analiza en términos de un conjunto de factores cada uno de los cuales tendrá asociada una métrica. Se explican seguidamente de forma resumida los factores que serán evaluados.

Completitud: Grado en que se ha logrado detallar todos los casos de uso relevantes.

Consistencia: Grado en que los casos de uso del sistema describen las interacciones adecuadas entre el usuario y el sistema.

Correctitud: Grado en que las interacciones actor/sistema soportan adecuadamente el proceso del negocio.

Complejidad: Grado de claridad en la presentación de los elementos que describen el contexto y la claridad del sistema.

A continuación se aplican un grupo de métricas al diagrama de caso de uso del sistema.

Aplicación de Métricas Orientadas a Objetos al Diagrama de Casos de Uso del Sistema

Factores de Completitud	Métricas Asociadas	Módulo de Tesorería
Factor 6. ¿Se presenta una descripción resumida (descripción de alto nivel) de todos los casos de uso del negocio?	Métrica 6: Número de casos de uso que no tiene descripción resumida Umbral: < 10% Acción sugerida: Completar la descripción resumida del caso de uso	Total de Casos de Uso: 27 Número de casos de uso que no tiene descripción resumida: 0 Representa: 0%
Factor 7. ¿Están definidos todos los requisitos que justifican la funcionalidad del caso de uso?	Métrica 7: Número de requisitos omitidos por caso de uso Umbral < 10% Métrica 8: Número de casos de uso que tienen requisitos omitidos Umbral < 10% Acción sugerida: Revisar la lista de requisitos para determinar cuáles serán apoyados por cada caso de uso	Total de Requisitos: 114 Total de Casos de Uso: 27 Número de requisitos omitidos por caso de uso: 0 Representa: 0% Número de casos de uso que tienen requisitos omitidos: 0 Representa: 0%
Factor 8. ¿Existen requisitos que no han sido considerados en algún caso de uso?	Métrica 9: Número de requisitos que no son considerados en ningún caso de uso.	Total de Requisitos: 114 Número de requisitos que no son considerados en ningún caso de uso: 0 Representa: 0%
Factor 10. ¿Se presenta una	Métrica 11: Número de casos de uso	Total de Casos de Uso: 27

descripción detallada (descripción extendida esencial) de todos los casos de uso del negocio?	que no poseen una descripción extendida. Umbral < 20% Acción sugerida: Interactuar con el usuario para realizar la definición extendida del caso de uso que sea consistente con la definición a alto nivel.	Número de casos de uso que no poseen una descripción extendida: 0 Representa: 0%
Factor 12. ¿Se describen las condiciones de excepción relevantes que debe contemplar cada flujo de eventos?	Métrica 14: Número de casos de uso que no describen condiciones de excepción relevante. Umbral < 20% Acción sugerida: Revisar las excepciones presentadas en el flujo de eventos que producen un mensaje de error al usuario.	Total de Casos de Uso: 27 Número de casos de uso que no describen condiciones de excepción relevantes: 5 Representa: 18.5%
Factor 13. ¿Todos los casos de uso del negocio han sido clasificados de acuerdo a su relevancia (primario / secundario / opcional)?	Métrica 15: Número de casos de uso que no han sido clasificados. Umbral: < 10% Acción sugerida: Hacer reuniones con los usuarios para analizar y priorizar los requisitos de acuerdo a su relevancia.	Total de Casos de Uso: 27 Número de casos de uso que no han sido clasificados: 0 Representa: 0%
Factores de Consistencia	Métricas Asociadas	Módulo de Tesorería
Factor 14. ¿El nombre dado a los casos de uso es una expresión verbal que describe alguna funcionalidad relevante en el contexto del usuario?	Métrica 16: Número de casos de uso que tienen un nombre incorrecto. Umbral < 20% Acción sugerida: Modifique el nombre del caso de uso de tal manera que signifique una acción desde el punto de vista del usuario.	Total de Casos de Uso: 27 Número de casos de uso que tienen un nombre incorrecto: 0 Representa: 0%
Factor 15. ¿Representa el caso de uso una interacción	Métrica 17: Número de casos de uso que no representan una interacción	Total de Casos de Uso: 27 Número de casos de uso

observable por un actor?	observable por un actor. Umbral < 5% Acción sugerida: Elimine el caso de uso e incorpore su funcionalidad como una responsabilidad del sistema dentro de otro caso de uso.	que no representan una interacción observable por un actor: 0 Representa: 0%
Factor 19. ¿La descripción del flujo de eventos se inicia con la descripción de una acción externa originada por un actor o por una condición interna del sistema claramente identificable?	Métrica 23: Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema. Umbral: < 10% Acción sugerida: Complete la definición del caso de uso incluyendo la acción fuera del sistema que da inicio al caso de uso o la condición interna que el sistema tiene control para dar inicio al caso de uso.	Total de Casos de Uso: 27 Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema: 0 Representa: 0%
Factor 21. ¿Existe una adecuada separación entre el flujo básico de eventos y los flujos alternos y/o flujos subordinados?	Métrica 25: Número de casos de uso complejos que no tienen separación del flujo básico y de flujos alternos. Umbral: < 20% Acción sugerida: Estructure el caso de uso de manera que separe su funcionalidad básica (caso de uso base) de la funcionalidad repetitiva o alternativa. Si hay pasos repetitivos forme un caso de uso que lo incluya y los pasos alternativos formen un caso de uso que lo extienda.	Total de Casos de Uso: 27 Número de casos de uso complejos que no tienen separación del flujo básico y de flujos alternos: 0 Representa: 0%
Factores de Correctitud	Métricas Asociadas	Módulo de Tesorería
Factor 23. ¿Representa el caso de uso requisitos	Métrica 28: Número de casos de uso en que los requisitos representados no son	Total de Casos de Uso: 27 Número de casos de uso en

comprensibles por el usuario?	comprensibles por el usuario. Umbral: < 5% Acción sugerida: Discuta con el usuario la interacción que describe el caso de uso y ajuste dicha descripción de manera que sea comprensible por el usuario.	que los requisitos representados no son comprensibles por el usuario: 0 Representa: 0%
Factores de Complejidad	Métricas Asociadas	Módulo de Tesorería
Factor 29. ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?	Métrica 36: Número de elementos del diagrama que requieren reubicación. Umbral: < 30% Acción sugerida: Modifique la ubicación de los elementos del diagrama de manera que los elementos relacionados se encuentren lo más cercano posible.	Total de Casos de Uso: 27 Número de elementos del diagrama que requieren reubicación: 0 Representa: 0%

Tabla 3.9: Métricas Orientadas a Objetos aplicadas al diagrama de casos de uso del sistema

Seguidamente se muestra un gráfico donde se evidencia mediante valores numéricos el resultado de la medición de los factores de calidad que han sido evaluados.

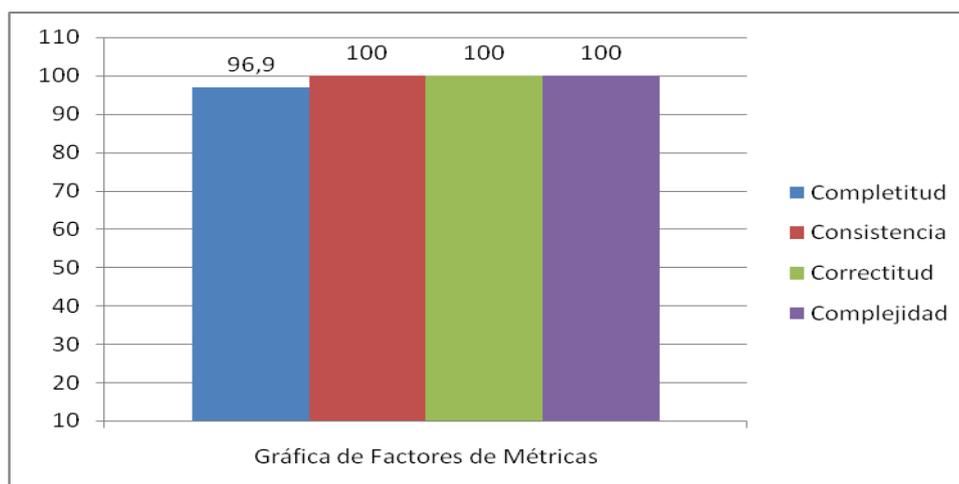


Figura 3.19: Gráfica de resultados por factores de las métricas aplicadas.

Con la aplicación de estas métricas se pudieron evaluar los factores completitud, consistencia, correctitud y complejidad del diagrama de casos de uso del sistema lo que dio la posibilidad de demostrar que los requisitos identificados son implementados en al menos un caso de uso, abarcando de esta forma todas las necesidades expresadas por el cliente, que los casos de uso fueron descritos detalladamente mostrando el flujo alterno a parte del flujo básico lo que da una mayor legibilidad de los mismos, evidenciándose además que estos son iniciados por la interacción del usuario con el sistema o por un evento interno dentro del mismo. Todo esto expuesto anteriormente permitió demostrar que el artefacto caso de uso del sistema se encuentra con la calidad requerida para entrar al flujo de trabajo análisis y diseño donde se comenzaría la realización del software.

Conclusiones

Después de haber realizado el flujo de trabajo de Requerimientos aplicando las actividades y técnicas que se llevan a cabo en la Ingeniería de Requisitos, se logró captar las necesidades expresadas por el cliente, identificando de esta forma los requisitos tanto funcionales como no funcionales, los cuales se especificaron a través de Casos de Uso y validaron mediante la técnica de prototipos de interfaz de usuario y la aplicación de métricas, demostrando que cuentan con la calidad requerida para llevar a cabo las exigencias planteadas por el cliente.

CONCLUSIONES

Una vez concluido el desarrollo del presente trabajo se puede llegar a las siguientes conclusiones:

- El estudio de las tendencias de los procesos de Tesorería y de la organización así como los diferentes talleres realizados con los especialistas, permitió determinar las funcionalidades del sistema utilizando tecnologías actuales y mejores prácticas.
- Mediante la representación del Modelo del Negocio se construyeron los artefactos correspondientes a este flujo y se identificaron las actividades automatizables así como los actores que ejecutan las diferentes funciones del sistema.
- Se logró obtener los requerimientos de software a través de la modelación del sistema, los cuales fueron especificados mediante casos de usos y validados a través de prototipos y métricas.

RECOMENDACIONES

- Se exhorta dar continuidad a los flujos de trabajo propuestos por RUP logrando de esta forma una solución informática que sea capaz de resolver los problemas reales para los cuales fue creada.
- Se recomienda realizar la gestión de requisitos para tener un control sobre los riesgos que puedan atentar con el buen desempeño del software, ya que este es un negocio muy cambiante debido a todas las transformaciones que en él se están llevando a cabo constantemente.

REFERENCIAS

- Barrientos Enríquez, Lic. Aleida Mirian.** El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML. <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml>. [En línea]
- Banco Central de Cuba. 2007.** Estadísticas. *Banco Central de Cuba*. [Online] 2007. <http://www.bc.gov.cu/Espanol/estadisticas.asp>.
- Booch, Grady, Jacobson, Ivar y Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000.
- Brooks, Fredrick P.Jr. 1987.** *No hay Balas de Plata. Esencia y Accidentes de la Ingeniería de Software*. 1987.
- Christel, M.G. and Kang, , KC. 1992.** *Issues in Requirements Elicitation*. 1992.
- Craig, L. 1999.** *UML Y PATRONES. Introducción al análisis y diseño orientado a objetos*. España : Pearson, 1999.
- Cueto, Yaumarys Pino and Bacallao Martínez, Maylín. 2007.** *Propuesta de un proceso para la Captura de Requisitos de la segunda fase del desarrollo del proyecto de Modernización de los Registros y Notarías de la República Bolivariana de Venezuela*. Caracas, Venezuela : s.n., 2007.
- Davis, A. 1993.** *Identifying and Measuring Quality*. 1993.
- freedownloadmanager.** Visual Paradigm for UML. [http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_\(M%C3%8D\)_14720_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/). [En línea]
- Geraldo, Angel.** Diseño y Modelación de un Proyecto de Software. Utilizando el lenguaje UML. <http://www.monografias.com/trabajos24/software-uml/software-uml.shtml>. [En línea]
- IEEE, IEEE Guide for developing System Requirements Specifications. 1998.** [Online] 1998.
- Jacobson, I. 1992.** *Object Oriented Software Engineering. A Use Case Driven Approach*. s.l. : Addison Wesley, 1992.
- Jacobson, Ivar, Booch, Grady and Rumbaugh, James. 2000.** *El Lenguaje Unificado de Modelado, Manual de Referencia*. 2000.
- Komer, P. 1993.** *Dirección de Mercadotenia*. España : Prentice Hall, 1993.
- Menéndez, MBA José Jorge Rodríguez and Monteagudo, Lic. José P. Carrillo.** Los flujos de efectivo. Su importancia en las decisiones empresariales. *Revista Betsime: La Revista del Empresario Cubano*. [Online] http://www.betsime.disaic.cu/secciones/fin_ja_06.htm.
- Molpeceres, Alberto. 2002.** Procesos de desarrollo: RUP, XP y FDD. [Online] 2002. <http://www.willydev.net/descargas/articulos/general/cualxpfddrup.PDF>.
- ÖVERGAARD, Gunnar and PALMKVIST, Karin., 2004.** *Use Cases: Patterns and Blueprints*. s.l. : Addison Wesley, 2004.
- Pérez, Alexei Zubizarreta. 2005.** *Proyecto Técnico-Económico para la Modernización de Registros y Notarías Nacional*. 2005.
- Pressman, Roger S. 2002.** *Ingeniería del Software. Un enfoque práctico*. 2002. 5ª Edición.
- Reque, Juan Miguel Salazar and Torres, María Esther Valderrama. 2007.** *Diagrama de clases y generación de código*. 2007.
- Robertson, J. and Robertson, S. 1999.** *Mastering the Requirements Process*. Inglaterra : Pearson, 1999.

- Rojas, MSc Carlos Mora, González García, Lic. Rosana and González Echagarrúa, Jennifer. 2003.** Sistema Automatizado para el Control de la Tesorería. *Biblioteca Central en Salud*. [Online] 2003. http://www.bvs.sld.cu/revistas/san/vol7_1_03/san54103.htm.
- Sanchés, María A. Mendosa. 2004.** *Metodologías de Desarrollo de Software*. 2004.
- sparxsystems.** Enterprise Architect - Herramienta de diseño UML. <http://www.sparxsystems.com.ar/products/ea.html>. [En línea]
- Terry, CP. Efraín Palenque and Morales, MSc. Mirtha Cepero. 2006.** Los Sistemas de Información de Gestión y la eficiencia empresarial. *Info 2008*. [Online] 2006. <http://www.congreso-info.cu/UserFiles/File/Info/Info2006/Ponencias/104.pdf>.
- Tripier, Benjamin. 2005.** La importancia de la Tesorería. *deGerencia.com*. [Online] 2005. http://www.degerencia.com/articulo/la_importancia_de_la_tesoreria.
- Vizcaíno, Aurora and Caballero, Ismael.** Una Herramienta CASE para ADOO: Rational Rose. [Online] http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/LabTr1_RationalRose.pdf.

BIBLIOGRAFÍA

Arango, J. 2002. "Tormenta de Ideas."

<http://www.eafit.edu.co/tda/boletin/TORMENTA%20DE%20IDEAS.htm>. [Online] 2002.

Ávila, Dra. Lourdes García and Fernández Sánchez, Ing. Leidy. *Procedimiento para el desarrollo del proceso de ingeniería de requisitos en un proyecto software (PROCIR)*.

Booch, Grady, Jacobson, Ivar y Rumbaugh, James. 2000. *El Proceso Unificado de Desarrollo de Software*. s.l. : Addison Wesley, 2000.

Christel, M.G. and Kang, , KC. 1992. *Issues in Requirements Elicitation*. 1992.

Craig, L. 1999. *UML Y PATRONES. Introducción al análisis y diseño orientado a objetos*. España : Pearson, 1999.

Coello González, Zayda y Hernández León, Rolando Alfredo. 2002. *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA*. Ciudad de la Habana : Editorial Universitaria, 2002.

Cueto, Yaumarys Pino and Bacallao Martínez, Maylín. 2007. *Propuesta de un proceso para la Captura de Requisitos de la segunda fase del desarrollo del proyecto de Modernización de los Registros y Notarías de la República Bolivariana de Venezuela*. Caracas, Venezuela : s.n., 2007.

Escalona, M.J. and Koch, N. *Ingeniería de Requerimientos en Aplicaciones Web: Un estudio comparativo*.

Jacobson, I. 1992. *Object Oriented Software Engineering. A Use Case Driven Approach*. s.l. : Addison Wesley, 1992.

Jacobson, Ivar, Booch, Grady and Rumbaugh, James. 2000. *El Lenguaje Unificado de Modelado, Manual de Referencia*. 2000.

Molpeceres, Alberto. 2002. *Procesos de desarrollo: RUP, XP y FDD*. [Online] 2002.

<http://www.willydev.net/descargas/articulos/general/cualxpfddrup.PDF>.

Pérez, Alexei Zubizarreta. 2005. *Proyecto Técnico-Económico para la Modernización de Registros y Notarías Nacional*. 2005.

Pressman, Roger S. 2002. *Ingeniería del Software. Un enfoque práctico*. 2002. 5ª Edición.

Sanchés, María A. Mendosa. 2004. *Metodologías de Desarrollo de Software*. 2004

GLOSARIO

ALBA: La Alternativa Bolivariana para América Latina y el Caribe o ALBA es una propuesta de integración enfocada para los países de América Latina y el Caribe que pone énfasis en la lucha contra la pobreza y la exclusión social.

CASE: Computer Aided Software Engineering. En su traducción al español significa Ingeniería de Software Asistida por Computación.

Caso de Uso: Representa a un proceso, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores. Desde la perspectiva de un actor individual, define un flujo de trabajo completo que produce resultados deseables.

Cuenta Bancaria: Una cuenta bancaria es un contrato financiero con una entidad bancaria en virtud del cual se registran el balance y los subsiguientes movimientos de dinero del cliente. Existen diferentes tipos de cuentas bancarias, como por ejemplo las cuentas corrientes, para efectuar las operaciones bancarias del día a día, o las cuentas de ahorro para encomendar la custodia de fondos de un cliente.

DCUS: Diagrama de Caso de Uso del Sistema.

ERP (Enterprise Resourcing Planning): Planificación de Recursos de la Empresa, estos sistemas se caracterizan por gozar de un alto nivel de integración entre sus módulos, además que abarcan la mayoría de los procesos de negocios que se llevan a cabo en la organización.

Estado de Cuenta: Es la copia textual del movimiento de la cuenta de un cliente. Los bancos están obligados a pasar a sus clientes mensualmente la relación completa del movimiento de sus cuentas corrientes y éstos a remitir la conformidad que viene anexa a dicho documento.

Ingeniería de Software: La Ingeniería de Software es una tecnología multicapa en la que, según Pressman, se pueden identificar: los métodos, el proceso y las herramientas.

Ingeniería de Requerimiento (IR): es el uso sistemático de procedimientos, técnicas, lenguajes y herramientas para obtener con un coste reducido el análisis, documentación, evolución continua de las necesidades del usuario.

Libro Auxiliar: Es un libro complementario a los principales libros de contabilidad, en el cual se registra cada uno de los movimientos hechos en una cuenta bancaria, como son el giro de cheques, notas débito, notas crédito, anulación de cheques y consignaciones, etc.

Metodologías: Se encargan de elaborar estrategias de desarrollo de software centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente.

MPPRIJ: Ministerio del Poder Popular para Relaciones Interiores y Justicia

Operación Bancaria: Son aquellas operaciones de crédito practicadas por un banco de manera profesional, como eslabón de una serie de operaciones activas y pasivas similares. Activas: Cuando el banco otorga el crédito (préstamos, descuentos, anticipo, apertura de créditos, etc.), el banco puede entregar dinero bajo diversas condiciones (con garantías o sin ellas). Pasivas: Son aquellos fondos depositados directamente por la clientela, de los cuales el banco puede disponer para la realización de sus operaciones de activo.

Proceso de Conciliación Bancaria: La conciliación bancaria es un proceso que permite confrontar y conciliar los valores que la empresa tiene registrados, de una cuenta de ahorros o corriente, con los valores que el banco suministra por medio del extracto bancario.

SAREN: Servicio Autónomo de Registros y Notarías.

Software: Es la suma total de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de cómputo y que “un producto de software es un producto diseñado para un usuario”.

UAC: Unidad Administradora Central.

UAD: Unidad Administradora Desconcentrada.

UEL: Unidad Ejecutora Local.

UML: El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.