

Universidad de las Ciencias Informáticas

Facultad 4



**SISTEMA DE CONTABILIDAD FINANCIERA PARA LA ACTIVIDAD
PRESUPUESTADA EN LAS FUERZAS ARMADAS REVOLUCIONARIAS**

Módulo Cobros-Pagos.

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

Autores: Keilyn Guerra Rojas.
Yisel Barberena Fraga.

Tutor: Ing. Luis Laguna Oliva
Consultante: Ing. Yunei López Lugo

Ciudad de La Habana, Julio, 2008

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Facultad ____ de la Universidad de las Ciencias Informáticas (UCI) y al Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR) para que haga el uso que estime pertinente con este trabajo.

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

Keilyn Guerras Rojas

Yisel Barberena Fraga

Ing. Luis Laguna Oliva

Síntesis del Tutor: Ing. Luis Laguna Oliva

Profesión: Ingeniero informático

Años de graduado: 1

Síntesis del Consultante: Ing. Yunei López Lugo

Profesión: Ingeniero informático

Años de graduado: 4

PENSAMIENTO

“Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio consciente de recibir el premio en la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte...”

Che.

AGRADECIMIENTOS

Gracias a la Revolución, a Fidel y a Raúl por habernos dado la oportunidad de estudiar en este centro universitario...

... es muy difícil mencionar nombres cuando se trata de agradecer a las personas que nos han ayudado a llegar hasta aquí... gracias a aquellos que creyeron en nosotras cuando ni nosotras mismas creímos, a nuestros amigos que nos han apoyado y soportado durante estos años, a nuestra jefa de proyecto Yuneí, a nuestro tutor Luis, a nuestro inseparable compañero y amigo Lelo, sin ellos no hubiésemos podido llegar al final, a nuestros compañeros del proyecto, a Yaima y a Alahín por su ayuda incondicional... gracias a todos por el apoyo que nos dieron cuando más lo necesitamos, sepan que les estaremos eternamente agradecidas.

Keilyn y Yisel.

Quiero agradecer eternamente a mi mamá y a mi papá que sin su apoyo, ayuda, comprensión, educación y amor no sería hoy lo que soy, gracias a mi hermana por haberse preocupado tanto por mí, a mi familia por siempre confiar en que podía seguir adelante, a mi novio por haberme apoyado incondicionalmente y acompañado en este camino tan difícil, gracias a mi suegro y mi cuñada por haberme ayudado al estar tan lejos de mi casa y darme aliento cuando más lo necesitaba, a mis vecinos por la preocupación y por la ayuda brindada a mis padres. A todos, mi eterno agradecimiento con la seguridad de que siempre estarán en mi corazón.

Yisel.

Quiero agradecer a mi mamá, a mi abuela y a papa por el amor, la dedicación, la comprensión y el apoyo que me han brindado en estos 23 años, por ellos, soy hoy lo que soy... a Naldito por su ayuda y preocupación en todo momento... a Tata por ser como mi hermana mayor... a Danny por creer en mí, por estar conmigo en los momentos más difíciles, por ser mi amigo, por ser mi amor... a Conchí, a Joaquín, a Marieta y a Tere por darme aliento para seguir hacia adelante... a mi papá y a Marita... a mi familia... Gracias...!!!

Keilyn.

DEDICATORIA

A mi mamá, a mi papá y a mi hermana...

Yisel.

A mi mamá, a mi abuela y a papa...

Keilyn.

... por enseñarnos que en la vida, las grandes cosas se alcanzan a base de sacrificios y empeño. Por darnos la fuerza que hace falta en los momentos de debilidad. Por tendernos la mano firme en las caídas más profundas. Por abrirnos los ojos cuando el cansancio y la derrota nos lo cerraron. Por ser el ejemplo a seguir, y por encima de todo, por existir...

RESUMEN

El Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR) ha incorporado nuevas tecnologías a la realización de sus procesos en aras de hacerlos más eficientes y controlados. En consecuencia de esto, se ha propuesto construir un software de Planificación de Recursos Empresariales (ERP), el cual estará compuesto por varios subsistemas de gestión. Se propone realizar el módulo Cobros-Pagos perteneciente al Subsistema de Contabilidad Financiera para contribuir al desarrollo de este gran software de gestión. Actualmente existe un sistema contable que no es totalmente factible, pues carece de funcionalidades por lo que no satisface a plenitud la necesidad que existe de gestionar y controlar la contabilidad financiera, dicho sistema no tiene implementado módulos para lograr una perfecta gestión de esta contabilidad tales como Cobros-Pagos, además no cumple con las nuevas concepciones de informatización del MINFAR.

PALABRAS CLAVES

ERP

Cobros y Pagos

Contabilidad Financiera

ÍNDICE

PENSAMIENTO	I
AGRADECIMIENTOS.....	II
DEDICATORIA	III
RESUMEN	IV
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1.1 Introducción	4
1.2 ¿Qué es un ERP?	4
1.3 Sistemas Automatizados existentes vinculados a la Contabilidad Financiera	4
1.4 Tendencias y tecnologías actuales	6
1.4.1 Arquitectura Cliente – Servidor	7
1.4.2 Aplicaciones Web	7
1.4.3 Servicio Web	7
1.4.4 Lenguaje y Metodología para el desarrollo de Sistemas Informáticos	8
1.4.5 Herramienta CASE	9
1.4.6 Lenguajes de programación para la Web	10
1.4.6.1 Lenguajes del lado del Servidor	10
1.4.6.2 Lenguajes del lado del cliente	11
1.4.7 Servidor de aplicaciones Web	12
1.4.8 Sistemas de Gestión de Bases de Datos (SGBD)	13
1.4.9 Navegador	14
1.4.10 Herramientas de desarrollo	14
1.5 Conclusiones	14
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	15
2.1 Introducción	15
2.2 Objeto de Estudio	15
2.2.1 Problema y Situación Problemática	15
2.2.2 Procesos Objeto de Automatización	16
2.2.3 Sistemas automatizados existentes vinculados al campo de acción	16
2.3 Información que se maneja	17
2.4 Propuesta de Solución	17
2.6 Modelo de Negocio	17
2.6.1 Actores y Trabajadores del negocio	18
2.6.2 Diagrama de Casos de Uso del negocio	18
2.6.3 Descripción de Casos de Uso del Negocio	19
2.6.4 Diagramas de Actividades por cada Caso de Uso del Negocio	22
2.6.5 Modelo de Objetos	24
2.7 Especificación de los requisitos de software	24
2.7.1 Requerimientos funcionales	24

2.7.2 Requerimientos no Funcionales.....	26
2.8 <i>Modelo del Sistema</i>	27
2.8.1 Actores del sistema.....	28
2.8.2 Descripción de los casos de uso del sistema	28
2.8.3 Diagrama de casos de uso del Sistema	30
2.8.4 Descripción expandida de los Casos de uso.....	30
2.9 <i>Conclusiones</i>	42
 CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	 43
3.1 <i>Introducción</i>	43
3.2 <i>Modelo de Análisis</i>	43
3.2.1 Diagramas de clases del Análisis.....	44
3.3 <i>Arquitectura del sistema</i>	47
3.4 <i>Modelo de Diseño</i>	48
3.4.1 Diagramas de Secuencia.....	49
3.4.2 Diagrama de clases del Diseño.....	52
3.5 <i>Diseño de la Base Datos</i>	57
3.5.3 Descripción de las Tablas de la Base Datos	59
3.6 <i>Principios de diseño</i>	66
3.7 <i>Concepción de la Ayuda</i>	67
3.8 <i>Tratamiento de errores</i>	67
3.9 <i>Seguridad</i>	67
3.10 <i>Acceso a Dato</i>	69
3.12 <i>Conclusiones</i>	71
 CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.....	 72
4.1 <i>Introducción</i>	72
4.2 <i>Implementación</i>	72
4.2.1 Diagrama de Componente	72
4.3 <i>Modelo de prueba</i>	74
4.3.1 Casos de Prueba.....	74
4.4 <i>Conclusiones</i>	79
 CONCLUSIONES.....	 80
RECOMENDACIONES.....	81
REFERENCIAS BIBLIOGRÁFICAS.....	82
BIBLIOGRAFÍA.....	83
GLOSARIO DE TÉRMINOS.....	84
ANEXOS	86

INTRODUCCIÓN

La contabilidad es una ciencia aplicada de carácter social y de naturaleza económica que proporciona información financiera del ente económico para luego ser clasificada, presentada e interpretada con el propósito de ser empleada para controlar los recursos y tomar medidas oportunas para evitar una situación deficitaria que ponga en peligro la supervivencia de la entidad.

La función principal de la contabilidad financiera es llevar en forma histórica la vida económica de una empresa, los registros de cifras pasadas sirven para tomar decisiones que beneficien la entidad en el presente o en el futuro. También proporciona los estados contables o estados financieros que son sujetos al análisis e interpretación, informando a los administradores, a terceras personas y a entes estatales del desarrollo de las operaciones de la entidad.

Dadas sus prestaciones, la contabilidad es una técnica mundialmente utilizada y por supuesto, Cuba no está exenta de ello. A su vez, el Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR), es una de las instituciones cubanas que hace uso de la contabilidad para la actividad presupuestada y empresarial.

Aparejado a todo esto viene el acrecentado desarrollo que se ha ido experimentando en el campo de la informática. En la década de los sesenta, con la introducción de la computación en diversos renglones de la sociedad en el mundo, se generó una gran demanda de software para resolver disímiles problemas teóricos y prácticos que existían en aquel momento.

Cuba no se ha quedado atrás en esta era de avances tecnológicos, ejemplo de ello es el hecho de ir incorporando las nuevas tecnologías a disímiles ramas de la sociedad como son la salud, educación, economía, etc.

La presente investigación, constituye una prueba más de lo anteriormente expuesto, pues el MINFAR se ha propuesto realizar un software de Planificación de Recursos Empresariales (ERP) que gestione todos sus procesos (Recursos Humanos, Contabilidad Material, Contabilidad Financiera, Nóminas, entre otros), por tanto este trabajo se encargará de efectuar una pequeña parte de ese enorme software, pues realizará el módulo Cobros-Pagos del Subsistema de Contabilidad Financiera para la actividad presupuestada perteneciente al ERP.

En las Fuerzas Armadas Revolucionarias (FAR) actualmente existe un sistema contable implementado en el lenguaje de programación Visual FoxPro, dicho sistema no es totalmente factible, pues carece de funcionalidades por lo que no satisface a plenitud la necesidad que existe de gestionar y controlar la contabilidad financiera, dicho sistema no tiene implementado módulos para lograr una perfecta gestión de esta contabilidad tales como Cobros-Pagos, es decir que esta operación se lleva a cabo de forma manual recogiendo los datos y archivándose en papel, tablas de Excel entre otras, lo que indica que no existe homogeneidad en la contabilidad, lo cual constituye un riesgo para el correcto desempeño y control de la gestión contable. El registro de operaciones aumenta de forma considerable por lo que es necesario amenizarle el trabajo al contador, que en el día realiza muchos cálculos y en muchos de los casos están sujetos a cometer errores. Además es muy difícil realizar búsquedas en un corto período de tiempo.

Teniendo como base lo anteriormente expuesto, el problema reside en: el proceso de Cobros-Pagos no está implementado en el Sistema de Contabilidad Financiera existente en las FAR.

El objeto de estudio del presente trabajo está enfocado en los procesos de la Contabilidad Financiera en los órganos de las FAR, definiendo como campo de acción el estudio específico del proceso de Cobros-Pagos de forma que se pueda obtener un sistema que informaticice el mismo.

El objetivo general que se plantea la presente investigación es: desarrollar el módulo Cobros-Pagos perteneciente al Subsistema de Contabilidad Financiera para la actividad presupuestada en las FAR, teniendo en cuenta los requerimientos de los usuarios y permitiendo una gestión eficiente de los procesos que se llevan a cabo para realizar dichas tareas. Para darle cumplimiento al objetivo general fueron definidos los siguientes objetivos específicos:

- ✓ Realizar el estudio del proceso de Cobros-Pagos.
- ✓ Realizar el análisis y diseño el proceso de Cobros-Pagos.
- ✓ Implementar del módulo Cobros-Pagos.
- ✓ Integrar los subsistemas de Contabilidad Financiera y Contabilidad Material.

Para darle cumplimiento a los objetivos específicos, se plantean las siguientes tareas que guiarán la investigación:

- ✓ Realizar encuentros con el cliente para un mejor entendimiento del negocio.
- ✓ Identificar el proceso de Cobros-Pagos describiéndolo de forma detallada.

- ✓ Estudiar las tecnologías y tendencias actuales que serán aplicadas en el desarrollo de la aplicación con el fin de lograr un mayor rendimiento y mejores resultados.
- ✓ Aplicar la metodología de Análisis y Diseño de sistemas informáticos que facilite y garantice la creación con calidad del sistema.
- ✓ Implementar el módulo de Cobros-Pagos para la informatización de sus procesos.
- ✓ Integrar los subsistemas de Contabilidad Financiera y Contabilidad Material mediante un servicio web.

Para la realización del presente trabajo, se ha planteado la siguiente hipótesis: si se desarrolla el proceso de Cobros- Pagos entonces existirá un módulo que cumplirá con los requisitos del usuario.

Se realizará una aplicación Web utilizando los lenguajes de programación PHP y Javascript, como metodología de desarrollo de software RUP, como gestor de base de datos PostgreSql y será utilizado el Visual Paradigm como herramienta Case.

El presente documento consta de cuatro capítulos:

- En el capítulo 1, se enmarcará el entorno en el que se desarrolla el problema, así como el estado del arte del tema en cuestión incluyendo un análisis de las herramientas, técnicas, metodologías que serán usadas para construir la solución al problema planteado.
- El capítulo 2, se refiere a las características del sistema, describiendo los procesos de negocio y derivando de los mismos los requerimientos del software. Se modela además el sistema con sus casos de uso bien detallados.
- En el capítulo 3 se realiza el análisis y diseño del software, determinando así cómo el sistema será desarrollado a partir de las funcionalidades previstas, es donde se indica con precisión lo que se va a implementar.
- En el capítulo 4 se obtiene el modelo de implementación que incluye diagramas de componentes y de despliegue, además el modelo de pruebas, el cual se ocupa de buscar los defectos del software.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se expondrán algunos conceptos que ayudarán al entendimiento del módulo que se pretende desarrollar, se realizará además un análisis detallado justificando las tecnologías, herramientas y metodologías a utilizar para su desarrollo, incluyendo la información referente al subsistema que se pretende construir. Se abordará además información referente a algunos sistemas informáticos vinculados al campo de acción.

1.2 ¿Qué es un ERP?

El ERP (Planificación de Recursos Empresariales) es un sistema integral de gestión empresarial que está diseñado para modelar e informatizar la mayoría de los procesos en una entidad (área de finanzas, comercial, logística, producción). Su misión es facilitar la planificación de todos los recursos de la entidad.

Lo más destacable de un ERP es que agrupa y ordena toda la información de la entidad en un solo lugar, de este modo cualquier suceso queda a la vista de forma inmediata, posibilitando la toma de decisiones de forma más rápida y segura, acortando los ciclos productivos. [1]

Varios son los puntos de vista en cuanto a los diferentes beneficios que se esperan en una implementación de un ERP, así como los impactos que este tendrá en la organización.

La mayoría de los ERP tienen en común varios beneficios:

- Solo un sistema para manejar muchos de sus procesos comerciales.
- Integración entre las funciones de las aplicaciones.
- Reduce los costos de gerencia.
- Incrementa el retorno de inversión.
- Fuente de infraestructura abierta.

1.3 Sistemas Automatizados existentes vinculados a la Contabilidad Financiera

Con el acrecentado desarrollo de la informática a nivel mundial ha surgido la necesidad de informatizar muchas esferas de la economía, con el fin de no quedar atrás en la era tecnológica, mitigar errores que el ser humano comete sin darse cuenta y acelerar el trabajo en cualquier empresa o institución. Muchas empresas han desarrollado software que de una forma u otra agilizan los procesos que en ella

se realizan, el campo de la contabilidad financiera ha hecho uso de estas tecnologías en aras de agilizar sus procesos y hacerlos más eficientes. Ejemplo de ello son:

✓ Sistema de Contabilidad Financiera de las FAR

Fue realizado en el lenguaje Visual FoxPro. Tiene como objetivo principal la informatización de los procesos de trabajo de la Contabilidad Financiera. No posee implementado un módulo de Cobros-Pagos, y además no es auditable, no está orientado a roles y no se sustenta en herramientas de software libre. Está compuesto por las siguientes opciones:

- Contabilidad
- Estados financieros
- Clasificadores
- Salva
- Créditos
- Rx-Tx

✓ SABIC (Sistema Automatizado para la Banca Internacional de Comercio)

El SABIC es un sistema diseñado y desarrollado para satisfacer las necesidades de procesamiento de datos de bancos e instituciones financieras no bancarias, utilizando los medios técnicos de computación disponibles en el mercado. Este sistema ha sido adaptado a los requerimientos de las operaciones propias del Banco Central de Cuba (BCC) y ha sido enriquecido para que los empleados que hagan uso de él puedan tramitar sus operaciones y realizar sus consultas sin necesidad de acudir a los archivos, ni a la actividad manual. De esta forma, aumenta la confiabilidad de la información y la seguridad y eficiencia del trabajo. Este sistema permite contabilizar en tiempo real, lo cual se traduce en que mantiene actualizados los ficheros contables en todo momento. Soporta una contabilidad multimoneda que hace innecesario depender del tipo de cambio para registrar los activos y pasivos y evita las conversiones de monedas a la hora de contabilizar. Estas características permiten disponer en todo momento de la información, lo más exacta posible, sobre la posición financiera de la institución.

✓ Software de Gestión (Sistema Económico verSat SaraSola)

Este Software de origen cubano, le ofrece al usuario la posibilidad de contar con un instrumento seguro, rápido, eficaz y de fácil manejo para la planificación, el control y el análisis de la gestión económica. Ha sido diseñado para utilizarse en cualquier tipo de entidad, permite llevar el control de los registros contables individuales de todos los hechos económicos que se originan en las estructuras

internas de las mismas y obtener los estados financieros de toda la información económica y contable. Es una solución de software dirigida especialmente a la agroindustria azucarera. Se estructura en un grupo de subsistemas, en los cuales se procesan y contabilizan los documentos primarios donde se anotan los movimientos de los recursos materiales, laborales y financieros que se utilizan en una entidad.

Subsistemas que lo integran:

- Configuración
- Contabilidad general
- Control de inventarios
- Generador de reportes
- Control de activos fijos
- Costos y procesos
- Finanzas, caja y banco
- Contratación y facturación
- Planificación económico-productiva
- Análisis económico empresarial
- Paquetes de gestión
- Nóminas de salario

Después de analizar y ver las potencialidades de los sistemas informáticos anteriormente mencionados para la gestión y consolidación de la Contabilidad, se puede decir que no se ajustan a las necesidades actuales del módulo que se pretende desarrollar, ya que no cumple con las nuevas concepciones de informatización existentes en las FAR.

1.4 Tendencias y tecnologías actuales

Para desarrollar un software informático es necesario tener en cuenta un grupo de tecnologías, herramientas y metodologías disponibles para llevar a cabo dicha empresa. La decisión de las idóneas a utilizar depende de las características de la entidad donde se va a implantar el sistema resultante, así como de lo que se desea del producto final conjugado con lo que ofrece cada tecnología. Partiendo de esto, seguidamente se realizará un análisis justificando la selección de las herramientas y metodologías a utilizar para elaborar el producto final.

1.4.1 Arquitectura Cliente – Servidor

Cuando se habla de aplicaciones Web es preciso pensar en la arquitectura más óptima que permita el control e intercambio de información a través de la red. La Arquitectura Cliente-Servidor es una de las más importantes y usadas en este ámbito de enviar y recibir información, como su nombre lo indica está compuesta por un cliente y un servidor, donde el cliente realiza una petición de recursos, información o servicios al servidor, y este último se encarga de proporcionar al cliente la respuesta a las peticiones realizadas. Una de las ventajas de esta arquitectura es que el acceso a la información se realiza de forma más ágil y al estar almacenada en el servidor existe un mejor control de la seguridad. [2]

Ver Anexo 1

1.4.2 Aplicaciones Web

Las aplicaciones Web se desarrollan como una extensión de los Sistemas Web para agregar funcionalidad de negocio al proceso. En otros términos, una aplicación Web es un sistema Web que permite a los usuarios ejecutar lógica de negocio a través de un Navegador (Browser), o lo que es lo mismo: modificar el estado del negocio. Su arquitectura general es la de un sistema cliente/servidor. El protocolo principal de comunicación en una aplicación Web es HTTP, el cual funciona normalmente desconectado, es decir, el cliente hace una petición al servidor, este la procesa y le devuelve el resultado, terminando la comunicación entre estos. [3]

1.4.3 Servicio Web

Los servicios Web permiten que las aplicaciones compartan información y que además invoquen funciones de otras aplicaciones independientemente de cómo se hayan creado las aplicaciones, cuál sea el sistema operativo o la plataforma en que se ejecutan y cuáles los dispositivos utilizados para obtener acceso a ellas. Aunque son independientes entre sí, pueden vincularse y formar un grupo de colaboración para realizar una tarea determinada.

Existen varios protocolos para la implementación de los servicios web. Estos protocolos son usados como medio de intercambio de mensajes estándar. Otros pueden servir como medio de transporte cuando se envía un documento XML por la red.

Para el intercambio de información entre la contabilidad financiera y la contabilidad material se escogió SOAP (Protocolo de Acceso a Objetos Simple) que es un lenguaje de mensajería basada en XML, el cual especifica todas las reglas necesarias para ubicar servicios Web XML, integrarlos en aplicaciones y establecer la comunicación entre ellos. Está creado con idea de dar un soporte completo y minucioso de todo tipo de servicios web. [4]

1.4.4 Lenguaje y Metodología para el desarrollo de Sistemas Informáticos

Las metodologías imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente. Lo hacen desarrollando un proceso detallado con un fuerte énfasis en planificar inspirado por otras disciplinas de la ingeniería.

El uso de una metodología para la elaboración de un producto informático, garantiza determinadas características en el mismo, dentro de ellas la calidad, factor clave tanto para el cliente como para el productor.

El desarrollo del sistema se basará en los fundamentos de la metodología del Proceso Unificado de Rational (RUP) y se utilizará como lenguaje de modelado UML.

✓ RUP (Proceso Unificado de Desarrollo)

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, es una metodología de ingeniería de software que permite aumentar la productividad del equipo de desarrollo y proporciona mejores prácticas de desarrollo a todos los miembros del equipo, además organiza y divide en 4 fases el desarrollo del software:

- Inicio: El objetivo en esta etapa es determinar la visión del proyecto.
- Elaboración: En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción: En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- Transición: El objetivo es llegar a obtener una versión del proyecto.

Cada una de estas etapas se desarrolla mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes. Importante mencionar que el ciclo de vida que se desarrolla por cada iteración es llevado bajo dos disciplinas:

Disciplina de Desarrollo

- Ingeniería de Negocios: Entendiendo las necesidades del negocio.
- Requerimientos: Traslado de las necesidades del negocio a un sistema informatizado.
- Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.
- Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.

Disciplina de Soporte

- Configuración y administración del cambio: Guardando todas las versiones del proyecto.
- Administración del proyecto: Administrando horarios y recursos.
- Ambiente: Administrando el ambiente de desarrollo.
- Distribución: Hacer todo lo necesario para la salida del proyecto.

✓ **UML (Lenguaje Unificado de Modelado)**

Es el lenguaje de modelado de sistemas de software más conocido en la actualidad. Es un lenguaje gráfico de modelado orientado a objetos estándar para especificar, visualizar, construir y documentar los elementos de los sistemas de software.

A continuación se enumeran los 9 diagramas que forman la base de UML, y dictan la manera en que es diseñado un sistema:

- Caso de Uso
- Clases
- Objetos
- Secuencia
- Colaboración
- De estado
- Actividad
- Componentes
- Implementación

1.4.5 Herramienta CASE

Las herramientas CASE (Ingeniería de Software Asistida por Ordenador de sus siglas en inglés Computer Aided Software Engineering,) se pueden definir como el conjunto de métodos, utilidades y técnicas que facilitan el mejoramiento del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases. Estas herramientas son el conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un Software, además de ser la mejor base para el proceso de análisis y desarrollo. La introducción de las herramientas CASE para ayudar en este proceso ha permitido que los diagramas puedan ser fácilmente creados y modificados, mejorando la calidad de los diseños de software.

Como herramienta CASE en la construcción de la solución propuesta en este trabajo, será utilizado el Visual Paradigm.

✓ Visual Paradigm

Es una herramienta CASE que utiliza “UML” como lenguaje de modelado. Esta útil herramienta apoya el ciclo de vida completo de desarrollo del software, análisis, diseño, implementación y prueba. Ofrece un diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad, además presenta disponibilidad de múltiples versiones, para cada necesidad y disponibilidad en múltiples plataformas. Permite exportar imágenes JPG, PNG entre otras [5]. Genera código y permite convertir código fuente a 10 lenguajes de programación, entre los más importantes están:

- JAVA
- XML
- PHP
- C++

1.4.6 Lenguajes de programación para la Web

Los lenguajes de programación para la Web son aquellos que admiten que las aplicaciones sean dinámicas, permitiendo la interacción con el usuario y la personalización de la información, pueden ser del lado del servidor o del lado del cliente.

1.4.6.1 Lenguajes del lado del Servidor

Los lenguajes del lado del Servidor se caracterizan por desarrollar la lógica del negocio dentro del servidor, además de ser los encargados del acceso a Bases de Datos y tratamiento de la información. Por el auge que han tenido, los más sobresalientes son: PERL (Lenguaje Práctico para la Extracción e Informe), ASP (Servidor de paginas Activas de sus siglas en ingles Active Server Pages), PHP, JSP (Servidor de Paginas en Java de sus siglas en ingles JavaServer Pages).

Como lenguaje para la programación de los procesos a informatizar se ha elegido el PHP.

✓ PHP (Procesador de Hipertexto)

PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas Web dinámicas, similar al ASP de Microsoft o el JSP de Sun, embebido en páginas HTML y ejecutado en el servidor. [6]

La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. No es un lenguaje de marcas como podría ser HTML o XML.

Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas web dinámicas:

- Soporte para una gran cantidad de bases de datos: MySQL, PostgreSQL, Oracle, Sybase, entre otras.
- Integración con varias bibliotecas externas, permite generar documentos en PDF (documentos de Acrobat Reader), analizar código XML.
- Perceptiblemente más fácil de mantener y poner al día que el código desarrollado en otros lenguajes.
- Soportado por una gran comunidad de desarrolladores, como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.

1.4.6.2 Lenguajes del lado del cliente

Los lenguajes del lado del cliente son aquellos que pueden ser directamente comprendidos por el navegador y no necesitan un pre tratamiento, son totalmente independientes del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio. Dentro de los lenguajes del lado del cliente se encuentran principalmente el JavaScript (JScript) y el Visual Basic Script (VBScript), que son los encargados de aportar dinamismo a la aplicación en los navegadores. En el caso del VBScript éste es prácticamente usado a la hora de programar en ASP del lado del servidor, ya que su mayor desventaja radica en que solo es soportado por el navegador Web de su fabricante, Microsoft. Por otro lado JScript es soportado por la mayoría de los navegadores existentes actualmente. Para la programación del lado del cliente se utilizará Javascript.

✓ **Javascript**

Es un lenguaje interpretado, es decir, que no requiere compilación. Gracias a su afinidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado, permite de forma eficiente validar formularios, crear cookies, detectar navegadores y mejorar el diseño. Puede interactuar con el código HTML, permitiendo a los programadores Web utilizar contenido dinámico. [7]

✓ **HTML**

HyperText Markup Language (Lenguaje de marcado de Hipertexto) es un lenguaje de marcas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las

páginas Web. Gracias a Internet y a navegadores como Explorer o Netscape, el HTML se ha convertido en uno de los formatos más populares que existen para la construcción de documentos. [8]

✓ XML

XML es una tecnología que tiene a su alrededor otras tecnologías que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. También conocida como un lenguaje universal de marcado para documentos estructurados y datos en la Web, más amplio, más rico y dinámico que HTML. No solo es un lenguaje de marcado, sino también un metalenguaje que permite describir otros lenguajes de marcado, además permite que los diseñadores creen sus propias etiquetas, permitiendo la definición, transmisión, validación e interpretación de datos entre aplicaciones y entre organizaciones. [7]

✓ AJAX

No es exactamente un lenguaje su nombre viene dado por las siglas de Asynchronous JavaScript y XML, es un término que describe un nuevo acercamiento a usar un conjunto de tecnologías existentes juntas, incluyendo las siguientes: HTML o XHTML, hojas de estilo (Cascading Style Sheets o css), Javascript, el DOM (Document Object Model), XML, y el objeto XMLHttpRequest que nos permite realizar una conexión al servidor, enviarle una petición y recibir la respuesta que procesaremos en nuestro código Javascript. Cuando se combinan estas tecnologías en el modelo Ajax, las aplicaciones funcionan mucho más rápido, ya que las interfaces de usuario se pueden actualizar por partes sin tener que actualizar toda la página completa. [9]

1.4.7 Servidor de aplicaciones Web

✓ Apache

Apache es un software libre de código abierto que funciona sobre cualquier plataforma, como por ejemplo, Unix, Windows, Macintosh y otras. Entre sus características presenta mensajes de error altamente configurables y base de datos de autenticación y negociado de contenidos. Es utilizado comúnmente para sitios con páginas estáticas. [10]

Ventajas:

- Modular (módulos cargados dinámicamente)
- Funciona sobre muchas plataformas
- Extensible
- Popular

- Gratuito.
- Apoyo fuerte para proveedores de Servicios de Internet (ISP's).
- Amplias librerías disponibles.
- Código fuente seguro

1.4.8 Sistemas de Gestión de Bases de Datos (SGBD)

Los Sistemas Gestores de Bases de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan, los cuales permiten incorporar una serie de funciones que posibilitan la definición de los registros, sus campos, sus relaciones, insertar, eliminar, modificar y consultar los datos. El propósito general de los sistemas de gestión de base de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de información.

Entre los SGBD más utilizados en el mundo se encuentran Oracle, MySQL, Microsoft SQL Server, PostgreSQL, InterBase, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional.

Estos sistemas presentan disímiles ventajas, entre las que se encuentran:

- Facilidad de manejo de grandes volumen de información.
- Gran velocidad en muy poco tiempo.
- Independencia del tratamiento de información.
- Seguridad de la información (acceso a usuarios autorizados), protección de información, de modificaciones, inclusiones, consulta.
- No hay duplicidad de información, comprobación de información en el momento de introducir la misma.

✓ PostgreSQL

Es un servidor de base de datos relacional, libre. Ofrece soporte total para transacciones, disparadores, vistas, procedimientos almacenados, almacenamiento de objetos de gran tamaño. Se destaca en ejecutar consultas complejas, consultas sobre vistas, subconsultas de gran tamaño. Permite la definición de tipos de datos personalizados e incluye un modelo de seguridad completo.

Como toda herramienta de software libre PostgreSQL ofrece entre otras ventajas la de contar con una gran comunidad de desarrollo en Internet, su código fuente está disponible sin costo alguno y algo muy importante es que dicha herramienta es multiplataforma, está disponible en casi cualquier sistema operativo. Además de presentar gran estabilidad y confiabilidad legendarias, en contraste a muchos

sistemas de bases de datos comerciales, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. [11]

1.4.9 Navegador

Un navegador web o explorador web (del inglés, navigator o browser) es una aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores web.

Se decidió utilizar como navegador para conectarse a la aplicación el Mozilla Firefox 1.5 o superior, el mismo presenta múltiples ventajas con respecto a otros navegadores existentes, pues ofrece mayor seguridad al usuario, es multiplataforma, cuenta con soporte multilingüe, es extensible, adaptable y provee diversos métodos para cambiar la funcionalidad y la apariencia.

1.4.10 Herramientas de desarrollo

✓ Dreamweaver

Es una herramienta de diseño web avanzada. Soporta gran cantidad de tecnologías muy fáciles de usar, por ejemplo: hojas de estilo y capas, javascript para crear efectos e interactividades, inserción de archivos multimedia. Dispone de muy buena documentación. A pesar de no ser libre, se utilizará por las ventajas que ofrece, entre ellas y como una de las principales, el trabajo en equipo. [12]

✓ EMS PostgreSQL Manager

Es una herramienta gráfica de gran alcance para la administración y el desarrollo del servidor de BD PostgreSQL. Permite no sólo ejecutar las consultas y scripts SQL, sino importar y exportar datos de otros formatos, manejar usuarios y privilegios, crear y editar bases de datos y tablas.

1.5 Conclusiones

Con la culminación del primer capítulo del trabajo se puede afirmar que se han abordado los conocimientos necesarios que han sido de gran ayuda a la hora de decidir las herramientas y tecnologías a utilizar para el desarrollo y culminación exitosa del subsistema. Luego de un estudio detallado se decidió utilizar como lenguaje de programación PHP, JavaScript para la implementación del lado del cliente además de AJAX, como gestor de bases de datos el PostgreSql, el navegador Mozilla Firefox, RUP como metodología de desarrollo del software, UML como lenguaje de modelado y como herramienta case fue elegido el Visual Paradigm.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En el desarrollo de este capítulo se comenzará la primera fase del desarrollo del software, es decir, se concibe hacer un análisis de los procesos que se llevan a cabo actualmente en Cobro-Pagos, de forma tal que se pueda delimitar un modelo de negocio para obtener un listado de requerimientos del sistema que serán objeto de informatización así como el modelamiento del sistema con una descripción detallada de cada caso de uso registrado.

2.2 Objeto de Estudio

El MINFAR, es la principal institución cubana encargada de la defensa nacional. Sus funciones económicas se desarrollan como una empresa más del país, pero con características diferentes ya que no tiene integrada la contabilidad material a la financiera.

Como se ha planteado inicialmente, el objeto de estudio del presente trabajo está enfocado en los procesos de la Contabilidad Financiera en los órganos de las FAR.

Estos procesos son:

- Contabilidad
- Caja y Banco
- Cobros y Pagos
- Estados Financieros
- Nóminas

2.2.1 Problema y Situación Problemática

En las FAR actualmente existe un sistema contable implementado en el lenguaje de programación Visual FoxPro, dicho sistema no es totalmente factible, pues carece de funcionalidades por lo que no satisface a plenitud la necesidad que existe de gestionar y controlar la contabilidad financiera, dicho sistema no tiene implementado módulos para lograr una perfecta gestión de esta contabilidad tales como Cobros-Pagos, es decir que esta operación se lleva a cabo de forma manual recogiéndose los datos y archivándose en papel duro o en tablas de Excel como en otras unidades, lo cual indica que no hay homogeneidad en la contabilidad, lo cual constituye un riesgo para el correcto desempeño y control de la gestión contable. El registro de operaciones aumenta de forma considerable por lo que es necesario amenizarle el trabajo al contador, que en el día realiza muchos cálculos y en muchos de los casos están sujetos a cometer errores de cálculo. Aparejado a esto es muy difícil realizar búsquedas en un corto

2.2.2 Procesos Objeto de Automatización

El proceso de Cobros-Pagos garantiza el recibo de varios documentos, Factura (SCM-110) e Informe de Recepción (SCM-109) provenientes de la Contabilidad Material para luego ser cobrados o pagados. El contador recibe n cantidad de documentos al día de una o varias especialidades, en el caso de un pago: se obtienen los SCM-109 y la factura de la empresa civil, se contabiliza en una cuenta por pagar, se conforma un comprobante de operaciones para luego emitir el correspondiente instrumento de pago, en el caso de un cobro: se obtiene a Factura si proviene de una empresa civil o el SCM-110 si proviene de una unidad militar, se contabiliza en una cuenta por cobrar y cuando llega el instrumento de pago se salda la cuenta.

De este control se tendrá el submayor de cuentas por especialidad y clientes, las deudas por especialidad y por clientes.

Los procesos de Cobros-Pagos que se realizan y que necesitan ser automatizados son los siguientes:

Obtener Documentos: Tiene lugar cuando el contador recibe los documentos pertinentes para ser cobrados o emitir un instrumento de pago.

Rechazar Documento: Tiene lugar cuando el contador rechaza los documentos por tener errores.

Contabilizar: Tiene lugar cuando el contador decide contabilizar unos o varios documentos.

Actualizar Pagos/Cobros de Documentos: Tiene lugar cuando el contador decide realizar el pago o cobro de uno o varios documentos.

2.2.3 Sistemas automatizados existentes vinculados al campo de acción

Como se ha explicado anteriormente, en el MINFAR existe un sistema de gestión de la contabilidad financiera, el cual no se utiliza en todas las unidades pertenecientes a dicha institución sino en solo una pequeña parte de éstas, el mismo no tiene implementado el módulo de Cobros-Pagos, además de presentar diversas dificultades:

- No está concebido para ser multiplataforma.
- No permite el trabajo de múltiples usuarios.
- No cuenta con un entorno visual usuario-sistema. Al ser implementado en un software de programación bastante obsoleto, la calidad de la interfaz con el usuario no es la mejor y brinda muy pocas facilidades, lo que hace más engorroso el trabajo.

2.3 Información que se maneja

En la realización del proceso de Cobros-Pagos se manejan una serie de documentos legalmente certificados, los cuales permiten registrar y documentar todas las operaciones que se llevan a cabo en la misma, éstos son:

- ✓ DOCUMENTO DE COBRO O PAGO: FACTURA (SCM-110), INFORME DE RECEPCIÓN (SCM-109).
- ✓ INFORME DE DIFERENCIA (SCM-112).
- ✓ CHEQUE.
- ✓ TRANSFERENCIA BANCARIA.
- ✓ COMPROBANTE DE OPERACIONES (SCM-56)

2.4 Propuesta de Solución

Para la realización del módulo Cobros-Pagos es necesario obtener los datos de una Factura (SCM-110) y de un Informe de Recepción (SCM-109), para esto contaremos con un servicio web para recibir los documentos previamente elaborados por el subsistema de Contabilidad Material, que nos brindará la posibilidad de aceptarlos si no tienen errores o rechazarlos en caso de tenerlos. En los lugares que no exista conexión contaremos con la posibilidad de crear el documento de cobro o pago, además de poder eliminar y modificar los datos de los mismos. Luego se pasarán uno o varios documentos a contabilizar dependiendo de cuántos estén pendientes o cuántos se deseen procesar en ese momento, agrupando los mismos por diferentes criterios y registrando la suma de los importes en un comprobante de operaciones. El proceso termina, cuando se debita o acredita el importe del instrumento de pago al órgano de finanza correspondiente. El sistema proporcionará la posibilidad de mostrar todas las deudas que posee uno o varios clientes seleccionados por el usuario y las deudas por especialidad que presenta una o varias especialidades seleccionadas, además de mostrar el submayor por cliente.

2.6 Modelo de Negocio

El modelado del negocio es una técnica para comprender los procesos del negocio de la organización. De acuerdo a los objetivos propuestos y a lo que se desea alcanzar como producto final, teniendo en cuenta la metodología de desarrollo utilizada, se ha efectuado un estudio de los procesos de negocio como paso necesario en el desarrollo del software, en consecuencia se ha elaborado el modelo del negocio.

Los propósitos que se persiguen al realizarse el modelado del negocio son:

- Entender la estructura y la dinámica de la organización.
- Comprender los problemas actuales e identificar mejoras potenciales.
- Asegurarse que los clientes, usuarios finales y desarrolladores tienen una idea común de la organización.
- Derivar los requerimientos del sistema a partir del modelo de negocio que se obtenga.

2.6.1 Actores y Trabajadores del negocio

Luego de haber realizado un análisis del flujo de los procesos que se realizan en Cobros-Pagos se concluye que los actores y trabajadores del negocio son los que se listan a continuación:

Actores del negocio

Actores	Descripción
Especialista	Persona que entrega los documentos de cobro o pagos en los órganos de finanzas.

Trabajadores del negocio

Trabajadores	Descripción
Contador	Persona que recibe los documentos de cobros o pagos en los órganos de finanzas.
Persona autorizada	Persona que revisa y autoriza el instrumento de pago.

2.6.2 Diagrama de Casos de Uso del negocio

Contabilizar cuentas por pagar: el contador recibe los documentos que le entrega el especialista para luego emitir el instrumento de pago correspondiente.

Contabilizar cuentas por cobrar: el contador recibe el documento que le entrega el especialista y espera que llegue el instrumento de pago que indique el pago de la misma.

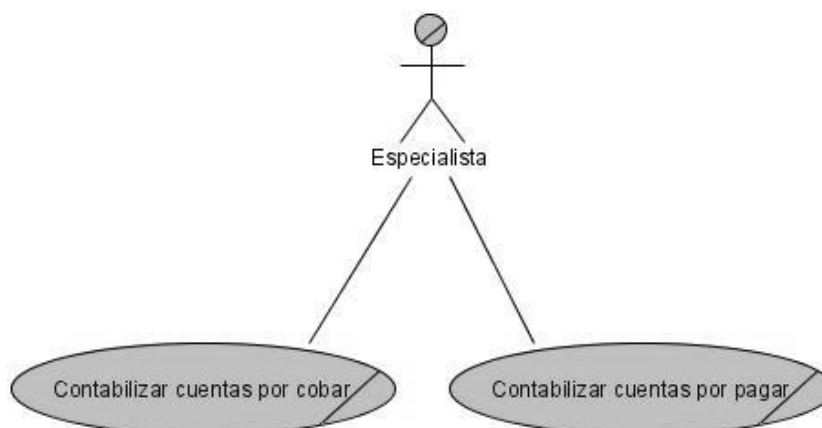


Fig.1 Diagrama de Casos de uso del Negocio, Módulo Cobros-Pagos.

2.6.3 Descripción de Casos de Uso del Negocio

Caso de Uso:	Contabilizar cuentas por cobrar	
Actores:	Especialista	
Trabajadores:	Contador	
Resumen:	El caso de uso inicia cuando el especialista entrega la factura (SCM-110) en el órgano de finanzas. Termina el caso de uso cuando el contador registra las operaciones contables después de haber liquidado la factura (SCM-110).	
Precondiciones:	Recibir factura (SCM-110) para emitir instrumento de pago.	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	1. El especialista entrega la factura (SCM-110) en el órgano de finanzas.	2. El contador recibe la factura (SCM-110). 3. El contador chequea que la factura (SCM-110) no tenga errores. 4. El contador archiva la factura (SCM-110) en el legajo correspondiente. 5. Registra las opresiones contables. 6. Verifica que exista el instrumento de pago. 7. Liquida la factura (SCM-110). 8. Registra las operaciones contables y se termina el caso de uso.
Flujos Alternos		

Capítulo 2: Características del Sistema

Acción 3: En caso de tener algún error devuelve la factura (SCM-110) al especialista.	
Acción 6: Si no existe el instrumento de pago verifica constantemente la existencia del mismo para cuando llegue liquidar la factura (SCM-110) correspondiente.	
Poscondiciones	
Mejoras	Los datos de la factura (SCM-110) llegarán a la entidad financiera a través de un servicio web desde el subsistema de Contabilidad Material. En caso de no haber conexión, se entregan a la entidad financiera y se introducen los datos a través de una aplicación web.
Prioridad	Crítica

Caso de Uso:	Contabilizar cuentas por pagar
Actores:	Especialista
Trabajadores:	Contador
Resumen:	El caso de uso inicia cuando el especialista entrega los documentos en el órgano de finanzas (factura e informe de recepción (SCM-109)). Termina el caso de uso cuando la persona autorizada revisa, firma el instrumento de pago y lo entrega al contador para éste entregarlo al especialista.
Precondiciones:	Recibir factura e informe de recepción (SCM-109) para emitir instrumento de pago.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. El especialista entrega los documentos en el órgano de finanzas.	2. El contador recibe los documentos. - Pago por Servicio (ver sección Pagar por Servicio). - Pago por Compra (ver sección Pagar por Compra). 3. Registra las operaciones contables. 4. El contador entrega el instrumento de pago a la persona autorizada para que lo firme. 5. La persona autorizada chequea el instrumento de pago. 6. La persona autorizada firma el instrumento de pago y lo entrega al contador. 7. El contador liquida el informe de recepción y registra las operaciones contables. 8. Registra el instrumento de pago y lo entrega al especialista.

Capítulo 2: Características del Sistema

9. El especialista recibe el instrumento de pago y se termina el caso de uso.	
Sección Pagar por Servicio	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El contador chequea los datos de la factura recibida. 2. El contador archiva la factura en el legajo correspondiente (ver flujo normal de eventos, acción 5).
Sección Pagar por Compra	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El contador chequea los datos de la factura e informe de recepción (SCM-109) recibidos. 2. El contador archiva la factura e informe de recepción (SCM-109) en el legajo correspondiente (ver flujo normal de eventos, acción 5).
Flujos Alternos	
Acción 5: Si existe error entrega el instrumento de pago al contador para que lo arregle.	
Sección Pagar por Servicio	
Acción 1: En caso de tener algún error devuelve la factura al especialista.	
Sección Pagar por Servicio	
Acción 1: En caso de error entre los importes de la factura e informe de recepción (SCM-109), verifica que exista un informe de diferencia, en caso de no existir devuelve los documentos al especialista al igual que si existe otro tipo de error.	
Poscondiciones	
Mejoras	Los datos de la factura (SCM-110) e informe de recepción (SCM-109) llegarán a la entidad financiera a través de un servicio web desde el subsistema de Contabilidad Material. En caso de no haber conexión, se entregan a la entidad financiera y se introducen los datos a través de una aplicación web.
Prioridad	Crítica

2.6.4 Diagramas de Actividades por cada Caso de Uso del Negocio

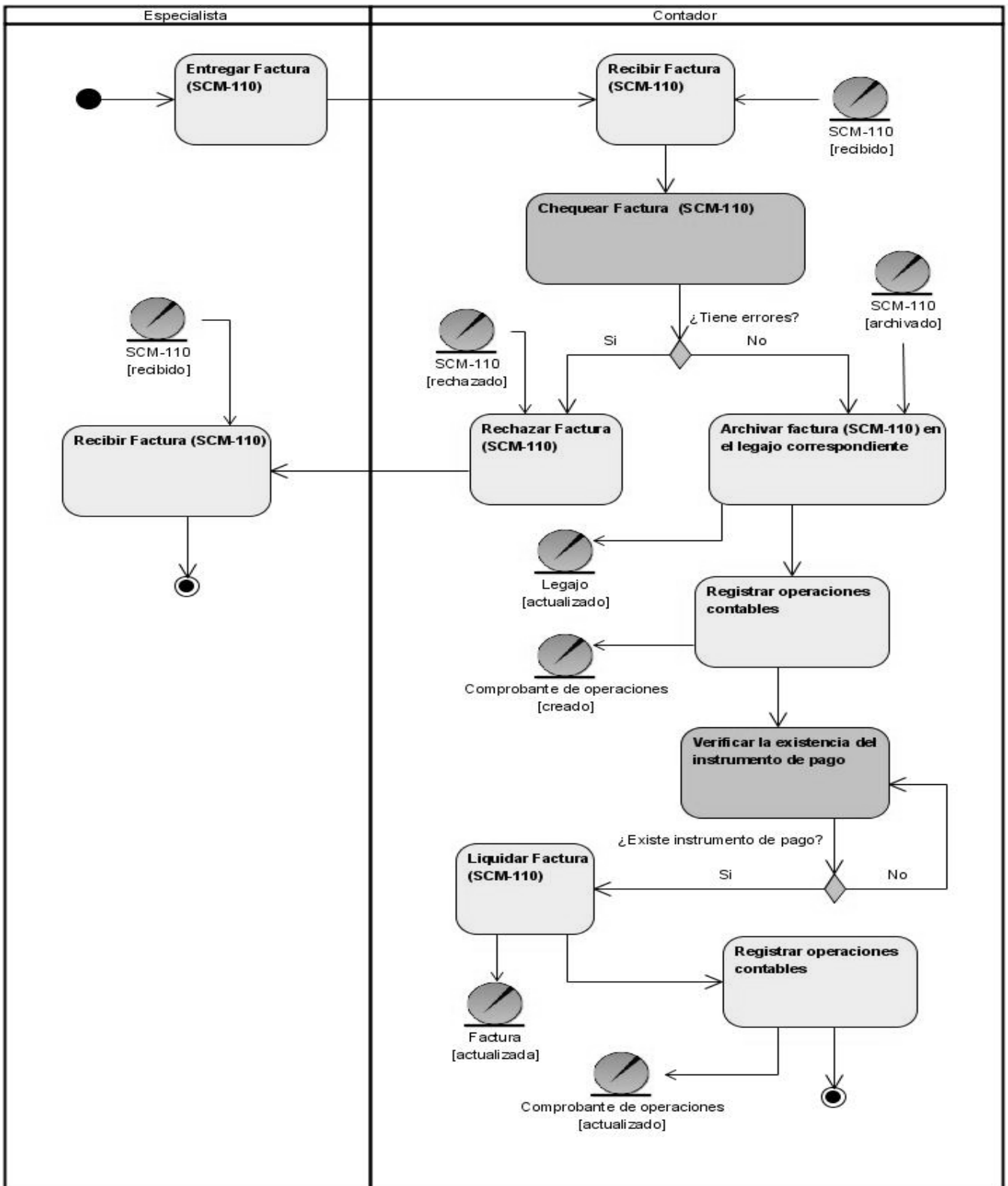


Fig.2 Diagrama de Actividades. CUN Contabilizar Cuentas por Cobrar.

Capítulo 2: Características del Sistema

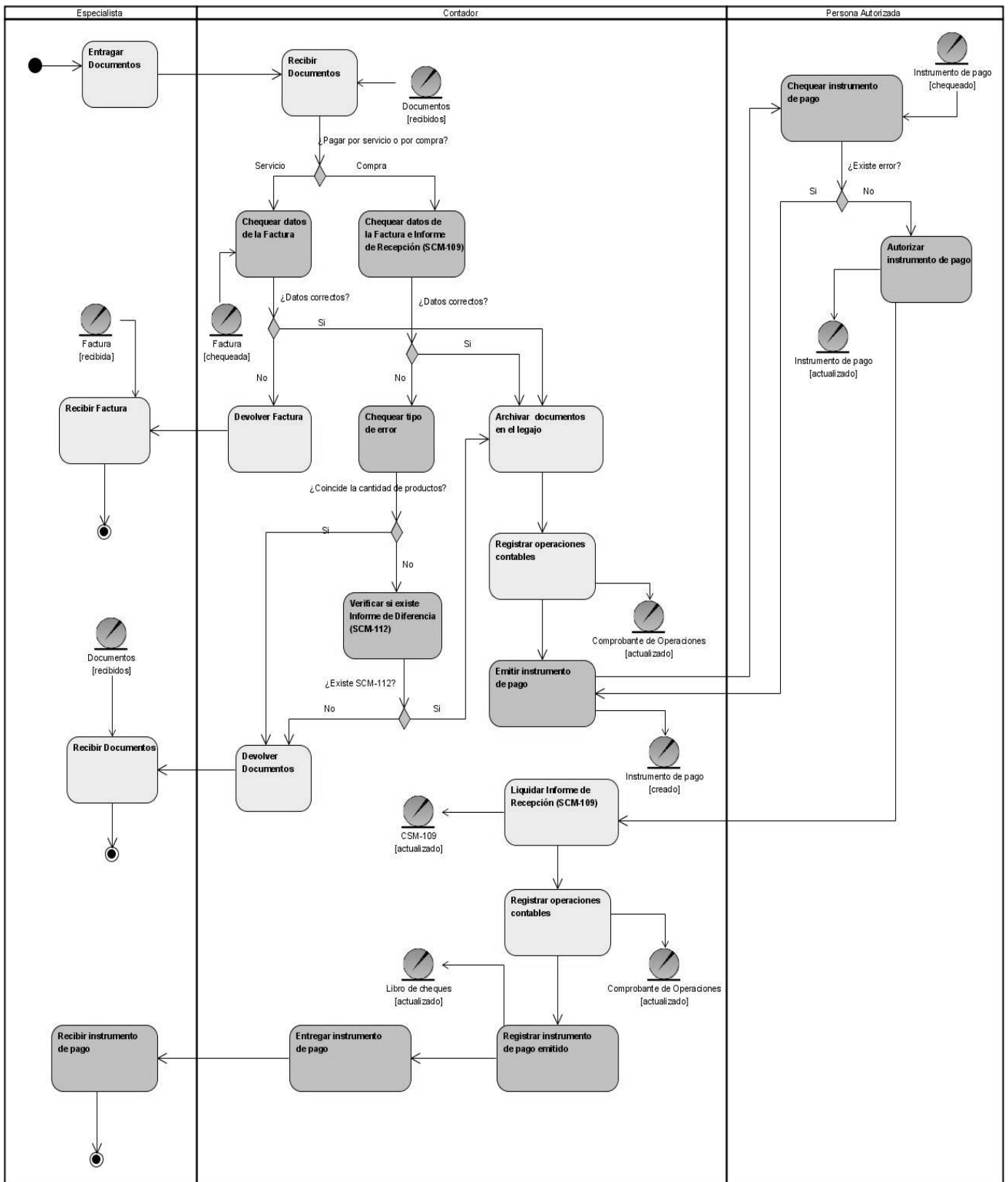


Fig.3 Diagrama de Actividades.CUN Contabilizar Cuentas por Pagar.

2.6.5 Modelo de Objetos

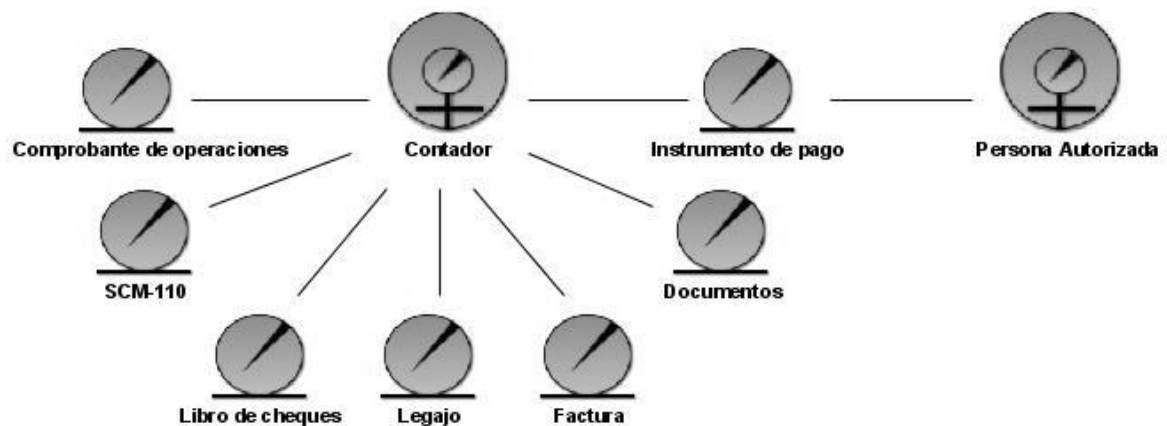


Fig.4 Modelo de Objetos del Negocio.

2.7 Especificación de los requisitos de software

Como próximo paso, corresponde realizar el levantamiento de requisitos, los mismos se dividen en funcionales y no funcionales.

Los requisitos funcionales constituyen las condiciones o capacidades que tiene que cumplir el sistema para satisfacer un contrato o documento formal, o sea, para satisfacer lo que el cliente ha solicitado, son los que describen el qué debe hacer el software, no el cómo. Los requisitos no funcionales son propiedades o cualidades que el producto debe tener porque representan las características del producto, conforman un grupo de categorías que pueden ser de software, hardware, de usabilidad, de seguridad, etc.; son características que debe cumplir la aplicación informática como complemento de los requisitos funcionales que son los que van a describir qué debe hacer el sistema.

2.7.1 Requerimientos funcionales

- ❖ R1 Obtener Documentos.
 - ✓ R1.1 Mostrar listados con documentos por Cobrar y por Pagar.
 - ✓ R1.2 Permitir seleccionar un documento e introducir la Cuenta al Debe y la Cuenta al Haber.
 - ✓ R1.3 Permitir rechazar un documento seleccionado.
- ❖ R2 Crear Documento
 - ✓ R2.1 Mostrar listado de los documentos no contabilizados.
 - ✓ R2.2 Permitir seleccionar o insertar el tipo de operación (cobro o pago), el concepto, la unidad militar, la especialidad, el número de la factura, la fecha de la factura, la cuenta al debe, el importe al debe, la cuenta al haber, el importe al haber, el cliente, el grupo presupuestario y el gasto.

- ✓ R2.3 Permitir insertar un cliente a partir del tipo de cliente seleccionado.
- ✓ R2.4 Permitir modificar los datos de un documento.
- ✓ R2.5 Permitir eliminar un documento.
- ❖ R3 Actualizar Pago/Cobro de Documentos
 - ✓ R3.1 Permitir seleccionar el concepto y el tipo de operación.
 - ✓ R3.2 Mostrar listado de los documentos no pagados.
 - ✓ R3.3 Permitir buscar en el listado por unidad militar, por registro, por cliente y por especialidad.
 - ✓ R3.4 Permitir seleccionar los documentos a pagar o cobrar.
 - ✓ R3.5 Permitir insertar la fecha, el número de cheque y la cuenta de banco.
- ❖ R4 Mostrar Deudas por Cliente
 - ✓ R4.1 Permitir seleccionar los datos: tipo de cliente (civil o militar), criterio (otros gastos, fondo de operaciones e inversiones (investigación y desarrollo, equipos y muebles, constructivas)).
 - ✓ R4.2 Permitir seleccionar los clientes a partir del tipo de cliente seleccionado.
 - ✓ R4.3 Permitir buscar por el código del cliente.
 - ✓ R4.4 Mostrar reporte de acuerdo a los criterios seleccionados.
- ❖ R5 Mostrar Deudas por Especialidad
 - ✓ R5.1 Permitir seleccionar los datos: especialidad, código de la especialidad, criterio (otros gastos, fondo de operaciones e inversiones (investigación y desarrollo, equipos y muebles, constructivas)).
 - ✓ R5.2 Permitir seleccionar las especialidades.
 - ✓ R5.3 Permitir buscar por el código de la especialidad.
 - ✓ R5.4 Mostrar reporte de acuerdo a los criterios seleccionados.
- ❖ R6 Mostrar Submayor por Cliente
 - ✓ R6.1 Permitir seleccionar los datos: tipo de cliente (civil o militar), criterio (otros gastos, fondo de operaciones e inversiones), especialidad, fecha inicial, fecha final, suma acumulada y suma desglosada.
 - ✓ R6.2 Permitir seleccionar los clientes a partir del tipo de cliente seleccionado.
 - ✓ R6.3 Permitir buscar por el código del cliente.
 - ✓ R6.4 Mostrar reporte de acuerdo a los criterios seleccionados.
- ❖ R7 Contabilizar
 - ✓ R7.1 Permitir seleccionar el tipo de operación (cobro o pago), concepto (otros gastos, fondo de operaciones e inversiones).
 - ✓ R7.2 Mostrar listado de los documentos no contabilizados.
 - ✓ R7.3 Permitir buscar por la unidad militar, el código de la especialidad y el código del cliente.

- ✓ R7.4 Permitir seleccionar los documentos a contabilizar.

2.7.2 Requerimientos no Funcionales

Apariencia o interfaz externa:

- ✓ El sistema debe tener una interfaz fácil de usar y amigable para que pueda ser utilizada sin mucho entrenamiento por el usuario.
- ✓ Empleo de imágenes identificadas con el negocio donde se implantará el sistema y colores agradables a la vista, siendo éstos claros.

Usabilidad:

- ✓ El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora.

Rendimiento:

- ✓ Los tiempos de respuesta y velocidad de procesamiento de la información serán rápidos, no mayores de 5 segundos para las actualizaciones y 20 para las recuperaciones.

Portabilidad:

- ✓ El sistema debe ser multiplataforma, importante el correcto funcionamiento en Linux y Windows.

Seguridad:

- ✓ Autenticación (Contraseña de acceso).
- ✓ Autorización (Atribución a los usuarios respecto a sus funciones de trabajo).
- ✓ Implementación de auditoria (Registrar la confirmación de cada operación efectuada por el usuario que afecte los registros contables).
- ✓ La atención al sistema incluyendo, el mantenimiento de las bases de datos así como la salva de la información se realizarán de forma centralizada por el administrador.

Políticos culturales:

- ✓ El sistema solo podrá ser utilizado en territorio cubano y por las entidades autorizadas por el Ministerio de las FAR.
- ✓ El producto no debe contener palabras en otros idiomas.
- ✓ El producto debe respetar los términos empleados normalmente por los especialistas en el tema de la esfera que se automatiza.

Legales:

El sistema está avalado por los tres documentos rectores emitidos en el país para la certificación y validación de los sistemas contables:

- ✓ La Resolución Conjunta de los ministerios de Finanzas y Precios de fecha 8.04.04.
- ✓ La Resolución 340 del Ministerio de Finanzas y Precios de fecha 8.12.04.

- ✓ La Resolución No. 12 del Ministerio de la Informática y las Comunicaciones de fecha 24.01.05.

Software:

Para el cliente:

- ✓ Navegador Mozilla Firefox.
- ✓ Sistema operativo Linux, Windows 98 o superior.

Para el servidor:

- ✓ Sistema operativo Windows Advancer Server (2000 o superior) o Linux en cualquiera de sus distribuciones.
- ✓ Un servidor Apache 2.0 o superior con módulo PHP 5.0 disponible, este debe estar configurado con la extensión "pgsql" incluida.
- ✓ Un servidor de base de datos PostgreSQL 8.0 o superior.

Hardware:

Para el servidor:

- ✓ Requerimientos mínimos: Procesador Pentium III a 1GHz de velocidad de procesamiento y 1Gb de memoria RAM (Memoria de Acceso Aleatorio de sus siglas en ingles Random Access Memory).
- ✓ Al menos 40Gb de espacio libre en disco duro.
- ✓ Tarjeta de red.

Para el cliente:

- ✓ Requerimientos mínimos: Procesador Pentium II a 133Mhz con 128 Mb de memoria RAM.
- ✓ Tarjeta de red.

Restricciones para el diseño e implementación:

- ✓ Emplear como servidores Web y de bases de datos Apache y PostgreSQL respectivamente.
- ✓ Utilizar como lenguaje del lado del servidor al PHP 5.0 o superior y del lado del cliente el JavaScript.

2.8 Modelo del Sistema

Como parte del flujo de trabajo de levantamiento de requisitos, se debe modelar la solución, para lo cual es preciso construir el modelo del sistema, para ello se debe definir, al igual que en el modelo del negocio actores del sistema y casos de uso del sistema para poder construir entonces los diagramas de casos de uso del sistema además de las descripciones textuales de cada uno de estos casos de uso.

2.8.1 Actores del sistema

Actores	Justificación
Contador	Persona que realiza los procesos de Cobros-Pagos en los órganos de finanzas.

2.8.2 Descripción de los casos de uso del sistema

Los casos de uso son de gran significación, ya que son fragmentos de funcionalidad que ofrece el sistema y tiene un resultado tangible para los actores. De aquí que requiera un análisis detallado.

CU -1	Obtener Documentos
Actores:	Contador (Inicia)
Descripción: Mediante este proceso el contador consulta el servicio web por el cual se intercambia información entre los subsistemas de Contabilidad Financiera y Contabilidad Material. El actor tiene la opción de revisar cada uno de los documentos recibidos (que representan un cobro o un pago), aceptarlos o rechazarlos para que pasen al estado de no contabilizados. El caso de uso concluye cuando se acepten o rechacen los documentos que han llegado por el servicio web y se hayan seleccionado.	
Referencias: R1, R1.1, R1.2, R1.3	

CU -2	Crear Documento
Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador tiene las opciones de Insertar, Modificar o Eliminar los datos de un documento. Estas operaciones se realizan en unidades donde no exista conexión de red o los documentos sean llevados al órgano de finanzas por mediación del especialista. El caso de uso concluye cuando el contador realiza alguna de estas acciones.	
Referencias: R2, R2.1, R2.2, R2.3, R2.4, R2.5	

CU -3	Actualizar Pago/Cobro de Documentos
Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador tiene la opción de pagar o cobrar uno o varios documentos. Debe seleccionar el tipo de operación y el concepto por el cual quiere filtrar la información para luego escoger él o los documentos. Tiene la opción de buscarlo por la unidad militar y por el registro donde se encuentra. Se le asigna comprobante de operaciones, con una fecha y una	

Capítulo 2: Características del Sistema

cuenta bancaria donde se va a pagar o cobrar. Se le emite un cheque y el documento se da como pagado o cobrado. El caso de uso concluye cuando se ha pagado o cobrado el documento.
Referencias: R3, R3.1, R3.2, R3.3, R3.4, R3.5

CU -4	Mostrar Deudas por Cliente
Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador puede conocer todas las deudas que tiene un o varios clientes de acuerdo a varios criterios que puede o no seleccionar. Se debe escoger el tipo de operación, tipo de cliente, el criterio por el cual se quiere filtrar, además tiene la opción de buscar el cliente por su código. Se emite un reporte con las deudas para los criterios seleccionados. El caso de uso concluye cuando el contador revisa los datos del reporte.	
Referencias: R4, R4.1, R4.2, R4.3, R4.4	

CU -5	Mostrar Deudas por Especialidad
Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador puede conocer todas las deudas que tiene una o varias especialidades de acuerdo a varios criterios que puede o no seleccionar. Se debe escoger el tipo de operación, la especialidad, el criterio por el cual se quiere filtrar, además tiene la opción de buscar la especialidad por su código. Se emite un reporte con las deudas para los criterios seleccionados. El caso de uso concluye cuando el contador revisa los datos del reporte.	
Referencias: R5, R5.1, R5.2, R5.3, R5.4	

CU -6	Mostrar Submayor por Cliente
Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador puede conocer el submayor de cuentas por pagar de uno o varios clientes. Tiene la opción de escoger el tipo de cliente, los criterios, un rango de fecha, y si quiere el reporte por suma acumulada o por suma desglosada. Se emite un reporte de acuerdo a los datos seleccionados. El caso de uso concluye cuando el contador revisa los datos del reporte.	
Referencias: R6, R6.1, R6.2, R6.3, R6.4	

CU -7	Contabilizar
-------	--------------

Actores:	Contador (Inicia)
Descripción: Mediante este caso de uso el contador pasa a contabilizar uno o varios documentos. Debe seleccionar el tipo de operación, el concepto, y escoger él o los documentos que desea contabilizar. Se le da la opción además de buscarlos por varios criterios como la unidad militar, el código de la especialidad y el código del cliente. El caso de uso concluye cuando el contador contabiliza los documentos seleccionados.	
Referencias: R7, R7.1, R7.2, R7.3, R7.4	

2.8.3 Diagrama de casos de uso del Sistema

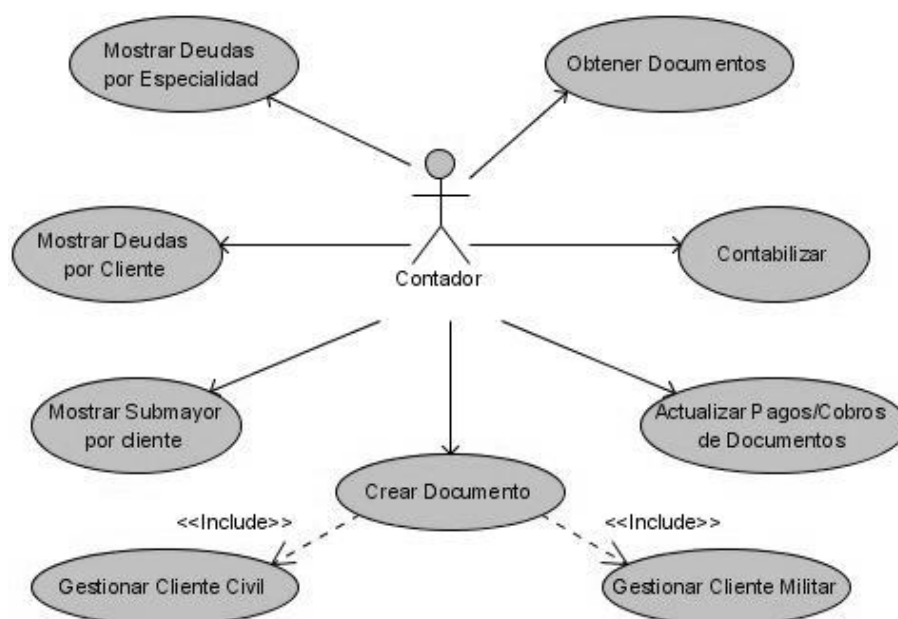


Fig.5 Diagrama de Casos de Uso del Sistema, Módulo Cobros-Pagos.

2.8.4 Descripción expandida de los Casos de uso.

Caso de Uso	Crear Documento
Actores	Contador
Propósito	Insertar, Modificar o Eliminar los datos de una factura.
Resumen	El contador inicia el caso de uso cuando decide Insertar, Modificar o Eliminar los datos de un documento. El caso de uso termina cuando el Contador realiza la acción deseada.
Responsabilidades	R2, R2.1, R2.2, R2.3, R2.4, R2.5
CU asociados	Gestionar Cliente Civil (extendido).

Capítulo 2: Características del Sistema

	Gestionar Cliente Militar (extendido).
Precondiciones	
Requisitos especiales	
Descripción	
Interfaz I.	

http://10.12.162.222 - Contabilidad - Mozilla Firefox

erp FAR Sistema Automatizado de Contabilidad Financiera de las FAR.
Contabilidad

Actualizar Deudas Submayor Contabilizar Cerrar Sesión 13/06/2008

Creación

Tipo de Operación: Cobro Cobro

Criterio * Seleccione

Nro de Factura

Fecha de Factura *

U. Militares * Seleccione

Especialidad Seleccione

Grupo Presupuestario Seleccione

Gasto Seleccione

Cliente * Civil Militar

Cuenta Debe *

Importe Debe *

Cuenta Haber *

Importe Haber *

(*) Campos obligatorios

ACEPTAR CANCELAR

Listado de Documentos por Contabilizar

U.Militar	Especialidad	Cliente	Factura	Fecha Factura	Cuenta Debe	Importe Debe	Cuenta Haber	Importe Haber	Grupo	Gasto
1390	14	ETECSA	2000	31/05/2008	135.2	12.30	135.2	2.30	711145	202C
1390	07	ETECSA	202	31/05/2008	135.2	100.00	135.2	100.00	170151	208C
1390	04	MICONS	808	31/05/2008	135.2	35.00	135.2	35.00	711145	103C
1390	05	MICONS	2222	31/05/2008	135.2	50.00	135.2	22.00	1112223	202C
1390	07	ECOI	98890	31/05/2008	412	88.00	412	88.00	711145	208C
1390	07	ECOI	887799	31/05/2008	137	55.00	137	55.00		202C
1390	08	minbas	11225588	31/05/2008	410	77.00	410	77.00		103C
1390	09	ECOI	774411	31/05/2008	425	22.00	425	22.00	711145	202C
1390	09	ECOI	115599	31/05/2008	410	66.00	410	66.00	1649491	208C
1390	08	hjppppp	884433	31/05/2008	410	55.00	410	55.00		208C
1390	07	2222	662266	31/05/2008	410	55.00	410	55.00	711145	208C
1390	07	2222	881199	31/05/2008	410	99.00	410	99.00		202C

Listo

- Control de aceptar una acción.
Nombre: Aceptar. Tipo: Botón.
- Control de cancelar una acción.
Nombre: Cancelar. Tipo: Botón.
- Control para limpiar los campos.
Nombre: Nuevo Documento. Tipo: Icono.
- Control para eliminar un documento seleccionado.
Nombre: Eliminar Documento. Tipo: Icono.
- Control para seleccionar el tipo de operación por la cual se desea filtrar.
Nombre: Tipo de Operación. Tipo: Radiobutton.
- Control que aparece visible de acuerdo al tipo de operación que se selecciona.

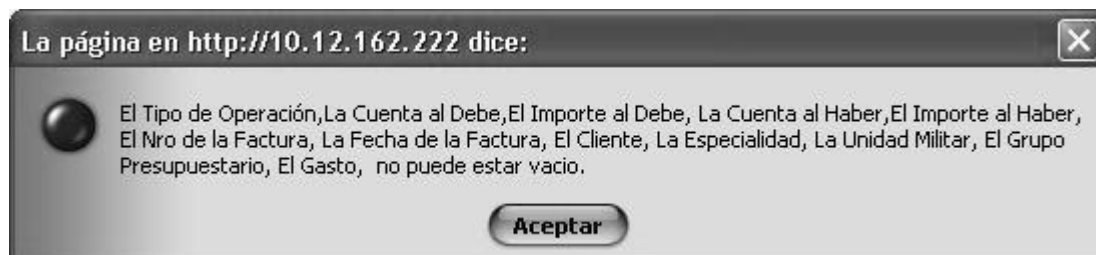
Nombre: Concepto. Tipo: Combo box.	
7. Control para introducir el número de la factura.	
Nombre: No Factura. Tipo: Text Box.	
8. Control para seleccionar la fecha de la factura.	
Nombre: Fecha Factura. Tipo: Text Box.	
9. Control para seleccionar la unidad militar a la cual está asociado el órgano financiero.	
Nombre: U Militares. Tipo: Combo Box.	
10. Control para seleccionar la especialidad que registró la factura.	
Nombre: Especialidad. Tipo: Combo Box.	
11. Control para seleccionar el grupo presupuestario asociado a la factura.	
Nombre: Grupo Presupuestario. Tipo: Combo Box.	
12. Control para seleccionar el gasto asociado a la factura.	
Nombre: Gasto. Tipo: Combo Box.	
13. Control para seleccionar el cliente.	
Nombre: Tipo de Cliente. Tipo: Radiobutton.	
14. Control para mostrar el o los clientes seleccionados.	
Nombre: Cliente. Tipo: Chek Box.	
15. Control que se carga a partir del concepto que se selecciona, muestra la o las cuentas asociadas al concepto seleccionado.	
Nombre: Cuenta Debe. Tipo: Combo Box.	
16. Control para introducir el saldo de la cuenta al debe.	
Nombre: Importe Debe. Tipo: Text Box.	
17. Control que se carga a partir del concepto que se selecciona, muestra la o las cuentas asociadas al concepto seleccionado.	
Nombre: Cuenta Haber. Tipo: Combo Box.	
18. Control para introducir el saldo de la cuenta al haber.	
Nombre: Importe Haber. Tipo: Text Box.	
19. Control para mostrar los datos de la o las facturas pendientes a contabilizar.	
Nombre: Grid. Tipo: Grid.	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona en el menú principal Actualizar. Luego	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas, además muestra en el Grid (19) los

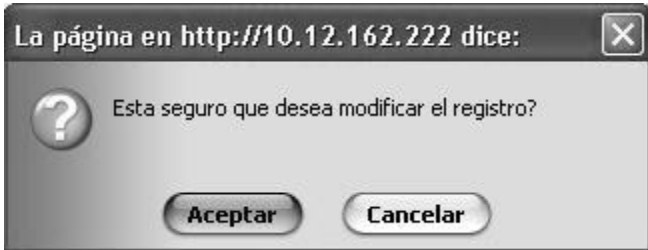
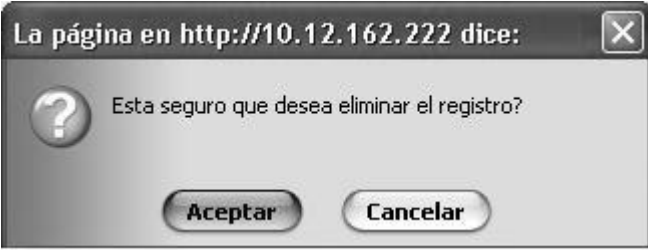
Capítulo 2: Características del Sistema

selecciona Crear Documento.	datos de los documentos que están pendientes a contabilizar.
<p>3. El contador decide: Crear Documento (Flujo Normal). Modificar Documento (1). Eliminar Documento (4).</p> <p>El contador selecciona el tipo de operación (5) que desea filtrar, si es un Cobro o un Pago.</p>	4. El sistema muestra los conceptos (6) de acuerdo al Tipo de Operación seleccionada.
<p>5. El contador selecciona el concepto (6), introduce el número del documento (7), selecciona la fecha del documento (8), selecciona la unidad militar (9), selecciona la especialidad (10), selecciona el grupo presupuestario (11), selecciona el gasto asociado (12), selecciona el tipo de cliente (13).</p>	6. El sistema ejecuta el caso de uso Gestionar Cliente Civil o Cliente Militar dependiendo del Tipo de Cliente seleccionado, muestra en el campo (14) el o los clientes seleccionados.
<p>7. El contador selecciona la cuenta al debe (15), introduce el importe al debe (16), selecciona la cuenta al haber (17), introduce el importe al haber (18) y presiona la opción Aceptar (1).</p>	8. El sistema muestra en el Grid (19) el documento creado que esta pendiente a contabilizar en conjunto con las que aún no se han contabilizado.

Cursos alternos

Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



<p>Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.</p> <p>Acción 6: Si no selecciona ningún cliente el sistema no realiza ninguna acción.</p>	
Sección Modificar	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona en el Grid (19) la factura que desea Modificar.	2. El sistema activa todos los controles cargándolos con los datos del documento seleccionado.
3. El contador modifica los datos que desee en los controles respectivos y presiona Aceptar (1).	4. El sistema pide confirmación de la acción. 
5. El contador presiona Aceptar.	6. El sistema modifica los datos y se termina el caso de uso.
Cursos alternos	
<p>Acción 4: Si el contador cierra la ventana, el sistema no ejecuta ninguna acción y se termina el caso de uso.</p>	
Sección Eliminar	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona en el Grid (19) la factura que desea Eliminar y presiona en la barra de Controles (4).	2. El sistema pide confirmación de la acción: 
3. El contador presiona Aceptar.	4. El sistema elimina la factura y se termina el caso de uso.
Cursos alternos	
<p>Acción 4: Si el contador presiona la opción Cancelar, el sistema no ejecuta ninguna acción y se termina el caso de uso.</p>	
Requerimientos no Funcionales	

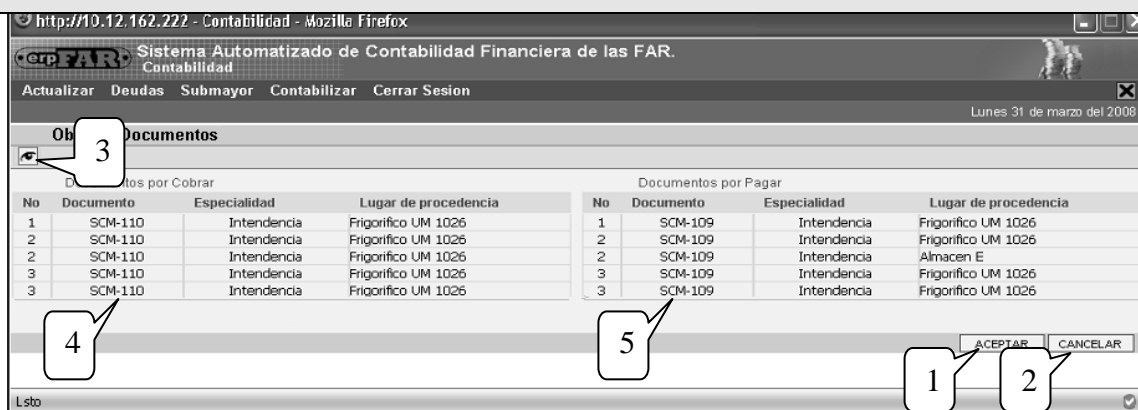
Capítulo 2: Características del Sistema

Postcondiciones	
Prioridad	Crítica

Caso de Uso	Obtener Documentos
Actores	Contador
Propósito	Aceptar o Rechazar los documentos que vienen de la Contabilidad Material.
Resumen	El caso de uso se inicia cuando el Contador decide consultar el servicio web. El actor tiene la opción de revisar cada uno de los documentos, aceptarlos o rechazarlos. Termina el caso de uso cuando el actor realiza alguna acción sobre los documentos.
Responsabilidades	R1, R1.1, R1.2, R1.3
CU asociados	
Precondiciones	
Requisitos especiales	

Descripción

Interfaz I.



1. Control de aceptar una acción.
Nombre: Aceptar. Tipo: Botón.
2. Control de cancelar una acción.
Nombre: Cancelar. Tipo: Botón.
3. Control para mostrar los datos de un documento seleccionado.
Nombre: Mostrar Documento. Tipo: Icono.
4. Control para mostrar los documentos por cobrar.
Nombre: Documentos por Cobrar. Tipo: Grid.

Capítulo 2: Características del Sistema

5. Control para mostrar los documentos por pagar. Nombre: Documentos por Pagar. Tipo: Grid.	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Contador selecciona del menú principal la opción Actualizar y luego escoge Obtener Documentos.	2. EL sistema muestra la Interfaz I con los documentos por cobrar y por pagar pendientes a aceptar o rechazar y las opciones Aceptar (1) y Cancelar (2) activas.
3. El Contador selecciona el documento que desea pasar al área de Contabilidad Financiera, y luego en la barra de herramientas presiona el icono para visualizar los datos (3).	4. El sistema muestra la Interfaz II o III (dependiendo si es un cobro o un pago) con los datos del documento seleccionado que viene de la Contabilidad Material. Activa los botones Rechazar (1), Aceptar (4) y Cancelar (5).
Interfaz II.	

Capítulo 2: Características del Sistema

http://10.12.162.222 - Contabilidad - Mozilla Firefox

MINFAR UM:		FACTURA SCM-110				No.		FECHA:	
Plan De Distribución No: FECHA:					Orden De Entrega No: FECHA:				
SUMINISTRADOR		NOMBRE MINFAR FONDO DE OPERACIONES: UM:		Código		Dirección			
		MUNICIPIO		AGENCIA BANCARIA No.		CUENTA BANCARIA No.		CONTRATO No.	
COMPRADOR		NOMBRE		CÓDIGO		Dirección			
		MUNICIPIO		AGENCIA BANCARIA No.		CUENTA BANCARIA No.			
		EFECTUAR ENTREGA EN:		INFORME RECEPCIÓN No.		FECHA		INFORME DE DIFERENCIAS No.	
								FECHA	
CANTIDAD DE BULTOS		TRANSPORTADOR		CARTA DE PORTE No.		CHAPA No.		No. DEL CARNÉ DE IDENTIDAD	
CÓDIGO		DESCRIPCIÓN		U/M		CANTIDAD		PRECIO	
								REC/DESC.	
								PRECIO U. TOTAL	
								IMPORTE	
								SALDO	
								IMPORTE TOTAL	
								.00	
OBSERVACIONES:									
ENTREGADO SUMINISTRADOR		RECIBIDO -- TRANSPORTADOR				RECIBIDO - CLIENTE		ANOTADO POR:	
NOMBRES Y APELLIDOS		NOMBRES Y APELLIDOS				NOMBRES Y APELLIDOS		T. ESTIBA	
								S. INV.	
CARGO		CARGO				CARGO		CONTABILIZADO POR:	
FIRMA		D		M		A		FIRMA	
								D	
								M	
								A	
								D	
								M	
								A	
Rechazar									
Actualizar Factura									
Cuenta al Debe		Seleccione		Cuenta al Haber		Seleccione			
								Aceptar	
								Cancelar	

1. Control para rechazar los documentos en caso de tener error.

Nombre: Rechazar. Tipo: Botón.

2. Control para seleccionar la cuenta al debe.

Nombre: Cuenta al Debe. Tipo: Botón.

3. Control para seleccionar la cuenta al haber.

Nombre: Cuenta al Haber. Tipo: Botón.

4. Control de aceptar una acción.

Nombre: Aceptar. Tipo: Botón.

5. Control de cancelar una acción.

Nombre: Cancelar. Tipo: Botón.

Interfaz III.

MINFAR UM: No: FECHA D M A

ENTIDAD SUMINISTRADORA CONDUCE FACTURA ALMACEN

TRANSPORTADOR NOMBRE Y APELLIDOS CARNET DE IDENTIDAD CHAPA No INFORME DE DIFERENCIA

BULTOS CARTA PORT CONOCIMIENTO DE EMBARQUE MANIFIESTO PARTIDA CONTRATO No CASILLA

Nro	CODIGO	Nro DE PIEZA	DENOMINACION	UM	CANTIDAD	PRECIO UNITARIO	RECARGO O DESCUENTO	PRECIO UNITARIO TOTAL	IMPORTE	SALDO EN EXISTENCIAS
IMPORTE DE LA RECEPCION										

OBSERVACIONES

RECEPCIONADO POR REVISADO POR ANOTADO POR CONTABILIZADO POR

NOMBRE Y APELLIDOS NOMBRE Y APELLIDOS TARJETA ESTIBA SUBMAYOR INVENTARIO

CARGO CARGO

FIRMA FIRMA

Rechazar

Actualizar Factura

Cuenta al Debe Seleccione Cuenta al Haber Seleccione

Aceptar Cancelar

1. Control para rechazar los documentos en caso de tener error.

Nombre: Rechazar. Tipo: Botón.

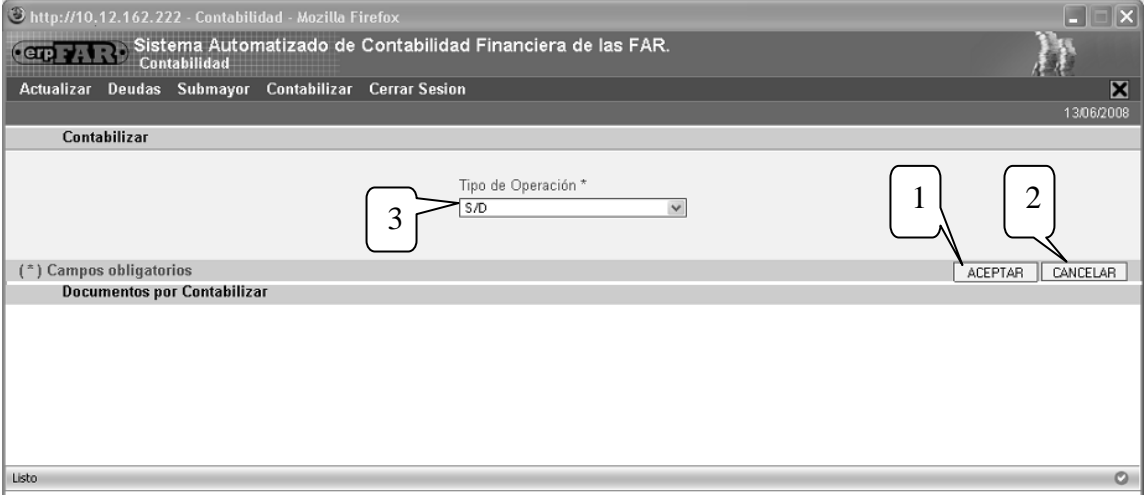
2. Control para seleccionar la cuenta al debe.

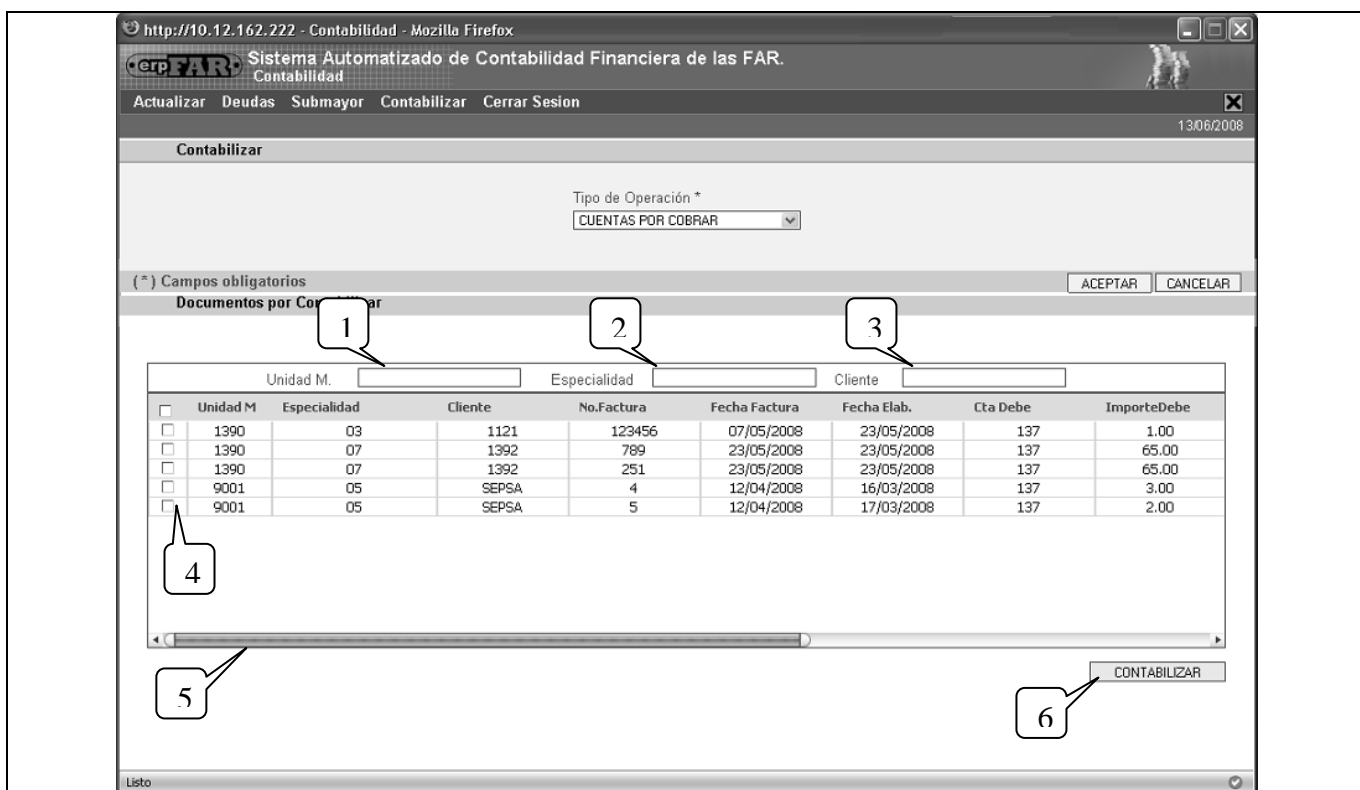
Nombre: Cuenta al Debe. Tipo: Botón.

Capítulo 2: Características del Sistema

<p>3. Control para seleccionar la cuenta al haber. Nombre: Cuenta al Haber. Tipo: Botón.</p> <p>4. Control de aceptar una acción. Nombre: Aceptar. Tipo: Botón.</p> <p>5. Control de cancelar una acción. Nombre: Cancelar. Tipo: Botón.</p>	
<p>7. El Contador revisa que los datos estén correctos. Selecciona el Código de la Cuenta (por el Debe (2) y por el Haber (3)) y presiona la opción Aceptar (4).</p>	<p>8. El sistema guarda los datos y se termina el caso de uso.</p>
Cursos alternos	
<p>Acción 2: Si el Contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.</p> <p>Acción 4: Si el Contador no selecciona el documento, no se realiza ninguna acción.</p> <p>Acción 5: Si los datos están incorrectos el Contador presiona la opción Rechazar (1) y se envía un mensaje de rechazo a la Contabilidad Material. Se termina el caso uso.</p> <p>Acción 5: Si el Contador selecciona la opción Cancelar (5) se cierra la ventana.</p> <p>Acción 6: Si el Contador no selecciona el código de la cuenta por algún criterio el sistema muestra un mensaje de error. El Contador presiona Aceptar.</p>	
Requerimientos no Funcionales	
Postcondiciones	
Prioridad	Crítica

Caso de Uso	Contabilizar
Actores	Contador
Propósito	Pasar a contabilizar los documentos que han llegado al órgano financiero.
Resumen	El Contador inicia el caso de uso cuando decide contabilizar uno o varios documentos. Termina el caso de uso cuando pasa a contabilizar los documentos seleccionados.
Responsabilidades	R7, R7.1, R7.2, R7.3, R7.4
CU asociados	

Precondiciones	
Requisitos especiales	
Descripción	
Interfaz I.	
	
<p>1. Control de aceptar una acción. Nombre: Aceptar. Tipo: Botón.</p> <p>2. Control de cancelar una acción. Nombre: Cancelar. Tipo: Botón.</p> <p>3. Control para seleccionar el tipo de operación por la cual que se desea filtrar. Nombre: Tipo de Operación. Tipo: Combo box.</p>	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona del menú principal Contabilizar.	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas.
3. El contador selecciona el tipo de operación que desea filtrar (3).	4. El sistema muestra la Interfaz II con la opción Seleccionar (6) activa.
Interfaz II.	



1. Control para buscar el documento por la unidad militar.

Nombre: Unidad M. Tipo: Text box.

2. Control para buscar el documento por la especialidad.

Nombre: Especialidad. Tipo: Text box.

3. Control para buscar el documento por el cliente.

Nombre: Cliente. Tipo: Text box.

4. Control para marcar el o los documentos, se pueden seleccionar todos además.

Nombre: Chek box. Tipo: Chek box.

5. Control para mostrar los documentos pendientes a contabilizar.

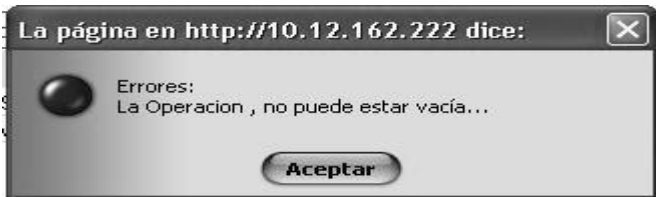
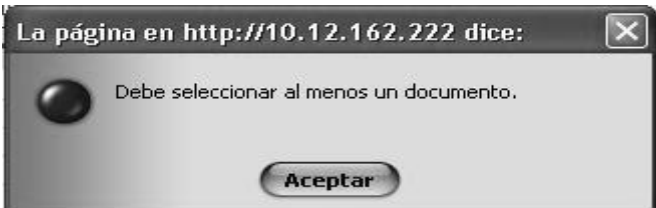
Nombre: Grid. Tipo: Grid.

6. Control para contabilizar los documentos seleccionados.

Nombre: Contabilizar. Tipo: Botón.

7. El contador busca él o los documentos por Unidad Militar (1), Especialidad (2) y/o Cliente (3), o también puede seleccionar directamente las facturas en el

8. El sistema contabiliza los documentos y se termina el caso de uso.

Grid (5) sin buscarlas. Presiona la opción Contabilizar (7).	
Cursos alternos	
<p>Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.</p>	
	
<p>Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.</p>	
<p>Acción 8: Si el Contador no selecciona ningún documento el sistema muestra el siguiente mensaje. El contador presiona Aceptar.</p>	
	
Requerimientos no Funcionales	
Postcondiciones	
Prioridad	Crítica

El resto de las descripciones expandidas de los casos de uso del sistema se encuentran en el *Anexo 2*.

2.9 Conclusiones

En este capítulo se han tomado importantes decisiones pues se ha comprendido bien el negocio, obteniendo de forma correcta los requisitos necesarios que el software debe cumplir además se han obtenido importantes resultados para los flujos de trabajo que siguen: Análisis, Diseño, Implementación y Pruebas, se ha definido los diagramas de casos de uso para cada módulo, que captura de forma íntegra los requisitos funcionales descritos, ya que son los casos de uso los que van a dirigir el desarrollo de la aplicación a lo largo de estos flujos y se obtuvo una descripción detallada de cada caso de uso así como los prototipos de las interfaces.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

En el presente capítulo, se pretende comenzar con el siguiente flujo de ingeniería después de la captura de requisitos, en el cual se obtuvo una vista externa del sistema puntualizando de forma concreta y detallada los procesos que soporta el mismo. El siguiente flujo de trabajo pertenece fundamentalmente a la fase de elaboración y constituye el análisis y diseño del sistema; se centra principalmente en obtener una vista interna del mismo, profundizando en cómo realizar cada uno de los procesos identificados como casos de uso, especificándolos de una forma más detallada y definiendo las clases necesarias para realizar las funcionalidades previstas. Serán analizados además una serie de patrones que influirán en la construcción de la arquitectura y en los mecanismos de diseño de la misma.

3.2 Modelo de Análisis

El Modelo de análisis se realiza para obtener una visión del sistema sobre los requisitos funcionales expresados ya en un lenguaje técnico, constituye una primera aproximación al modelo de diseño. El mismo ofrece ventajas tales como: suavizar la transición al diseño, apoyar el cambio a otra plataforma de programación, servir para obtener una visión general de la propuesta del sistema así como para planificar y dividir el diseño e implementación en pequeños módulos, apoya la aplicación de reingeniería a aplicaciones existentes (al ser en un lenguaje menos técnico ayuda a entender mejor la propuesta de solución). Está orientado a analizar cómo el sistema va a cumplir sus funcionalidades. El análisis debe, pues, capturar los requisitos de usuario sin adoptar prematuramente decisiones de implementación, es decir, omitiendo detalles dependientes de la tecnología, y utilizando conceptos extraídos únicamente del dominio del problema.

A continuación se representan los modelos de clases del análisis por cada uno de los casos de uso definidos.

3.2.1 Diagramas de clases del Análisis

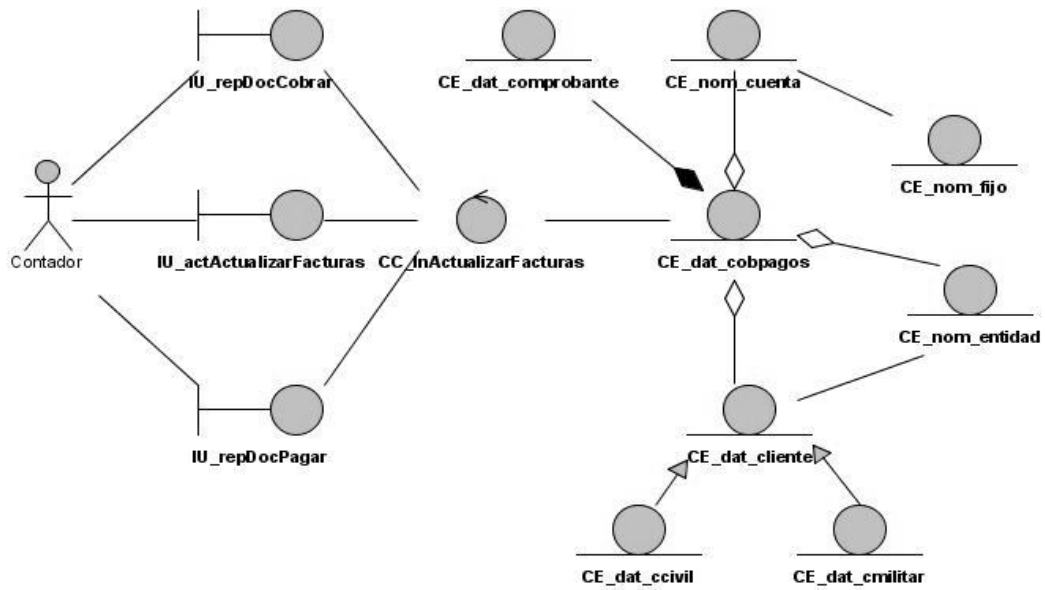


Fig.6 Diagrama de clases del análisis. CU Obtener Documentos.

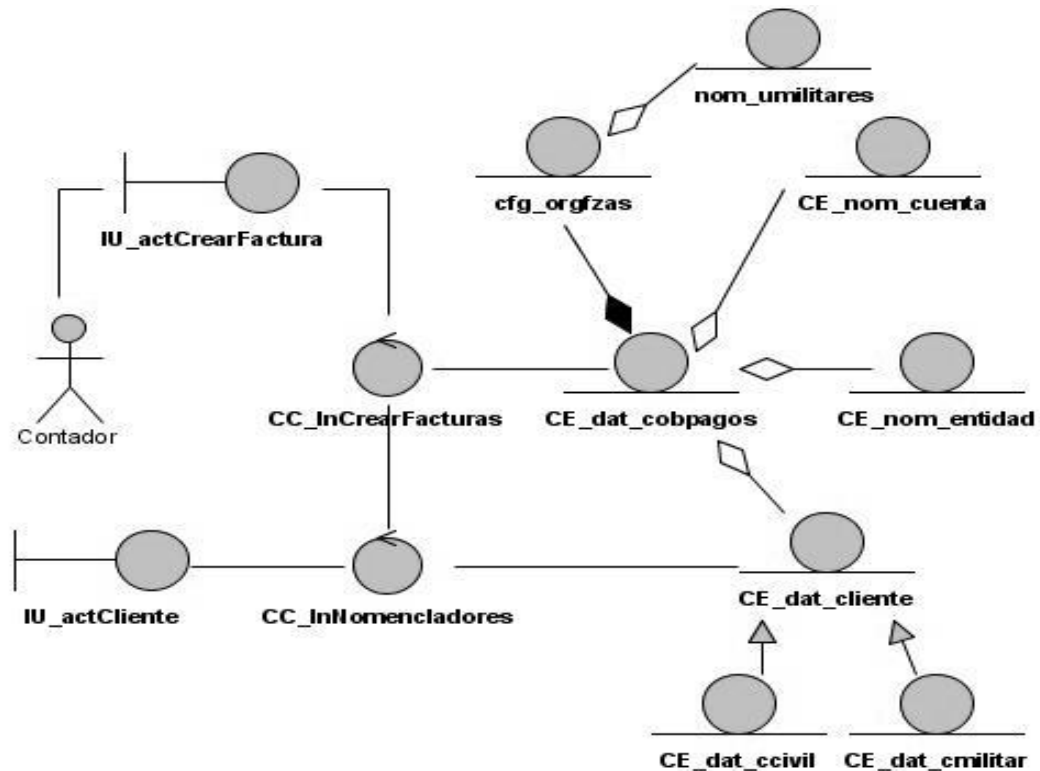


Fig.7 Diagrama de clases del análisis. CU Crear Documento.

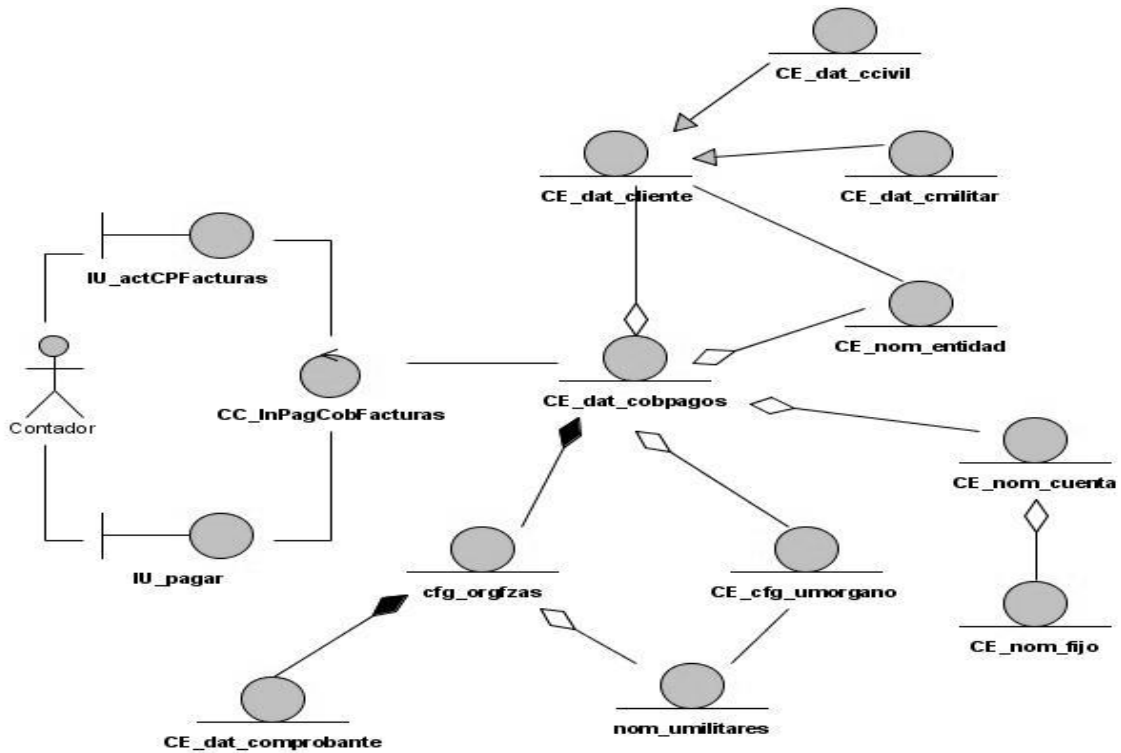


Fig.8 Diagrama de clases del análisis. CU Actualizar Pago/Cobro de Documentos.

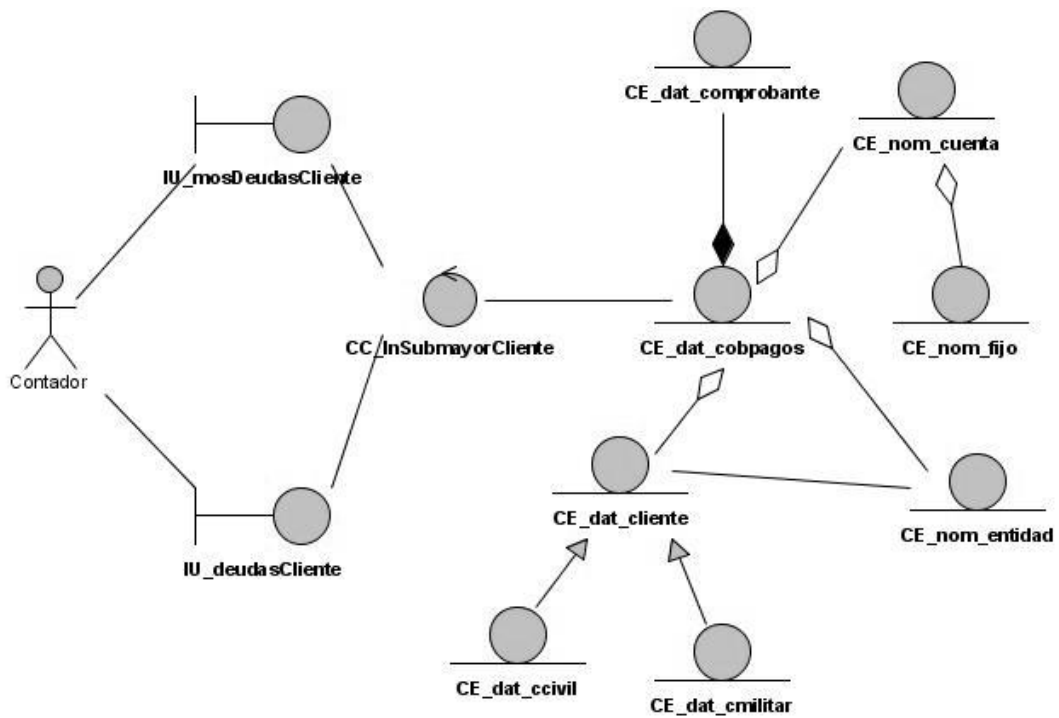


Fig.9 Diagrama de clases del análisis. CU Mostrar Deudas por Cliente.

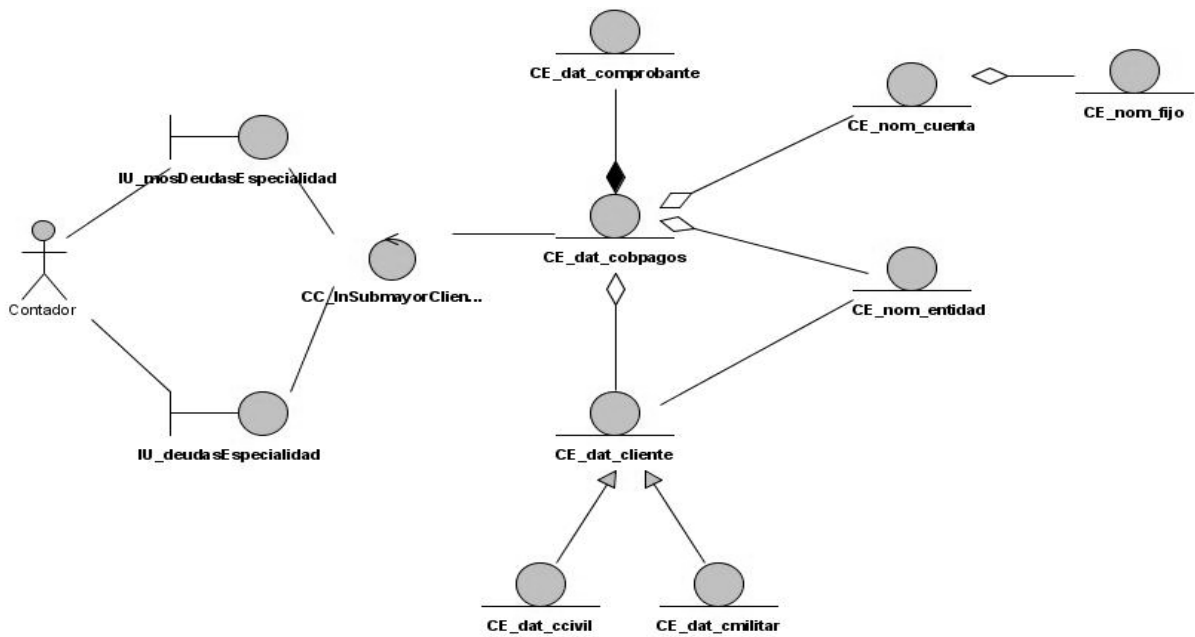


Fig.10 Diagrama de clases del análisis. CU Mostar Deudas por Especialidad.

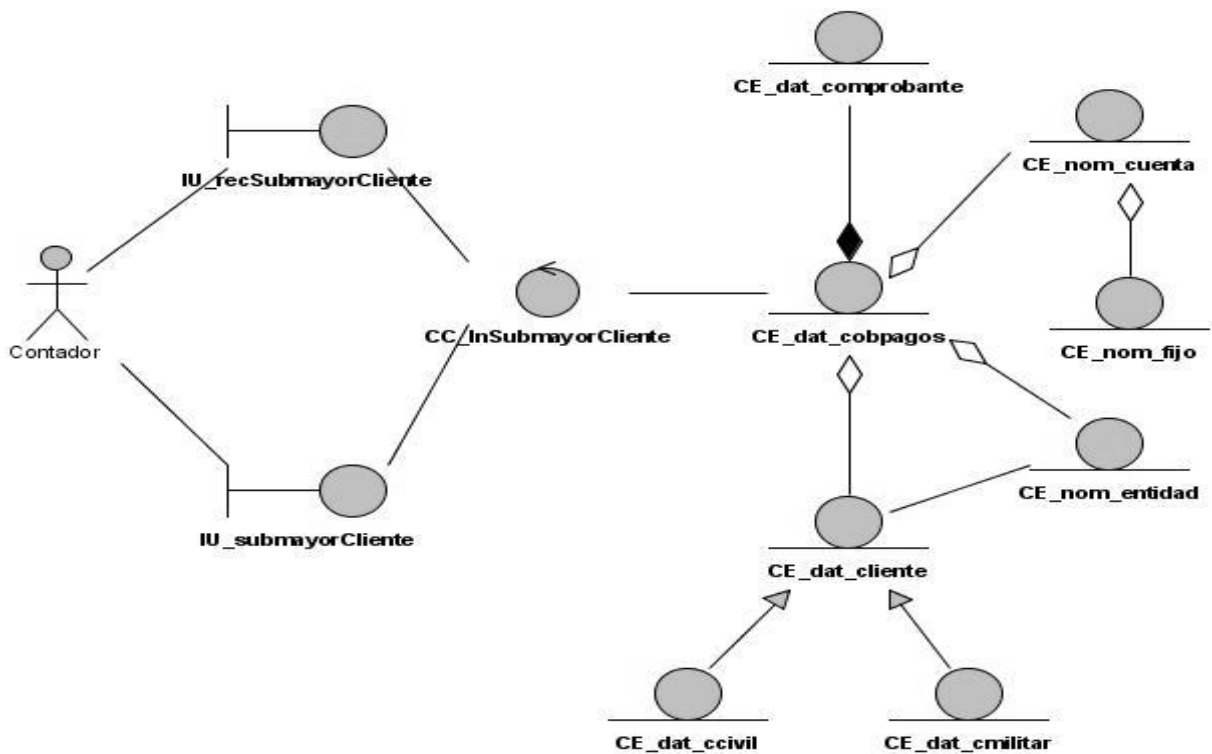


Fig.11 Diagrama de clases del análisis. CU Mostrar Submayor por Cliente.

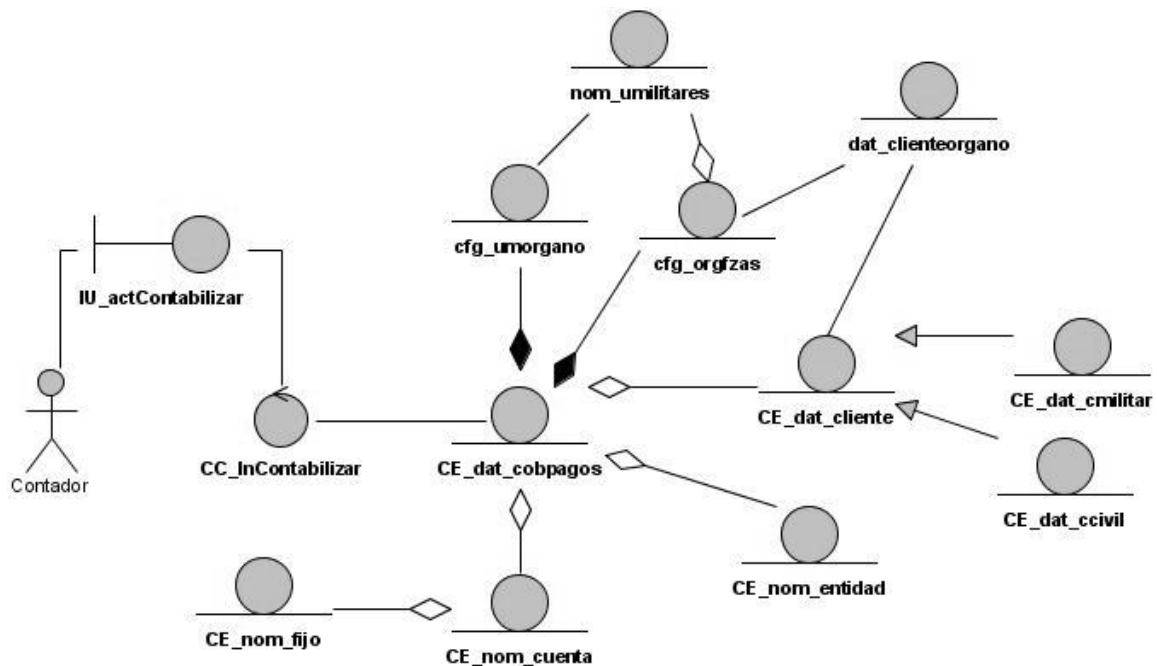


Fig.12 Diagrama de clases del análisis. CU Contabilizar.

3.3 Arquitectura del sistema

Con el objetivo de comprender mejor el sistema y organizar el desarrollo se hace necesario describir la arquitectura definida por el proyecto anteriormente, esta se mantendrá durante el diseño y la implementación.

La arquitectura que se empleará para el desarrollo de esta aplicación responde a la Arquitectura en capas la cual proporciona una buena organización y estructuración entre los distintos niveles de abstracción, donde un cambio en uno de estos niveles no debe proporcionar cambios en los restantes.

El diseño más empleado en la actualidad es el de *tres capas*, siendo estas:

Capa de presentación: es la capa mediante la cual el sistema le brinda la posibilidad al usuario de interactuar con él, se encarga de mostrar la información y captura la misma con el objetivo de procesarla, además en ella se realiza una primera validación de la información con el objetivo de evitar posible errores. Esta capa solo se comunica con la capa lógica del negocio.

Capa Lógica de negocio: es una de las capas más importante, se encarga de recibir las peticiones de los usuarios y enviar las respuestas, aquí es donde se establecen todas las reglas que deben cumplirse y se manipula toda la información. Esta capa se comunica con la capa de presentación, para

recibir las solicitudes y mostrar los resultados, y con la capa de acceso a datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él, para el acceso a datos esta capa se relaciona con una clase que implementa la interfaz del modelo de persistencia que responde a un patrón llamado Factory la cual es una puerta de enlace entre la capa de Acceso a Datos y la capa de Lógica de Negocio.

Capa de Acceso a datos: Es la capa encargada de manejar todo el flujo de información que entra y sale de la fuente de datos, así como la conexión a la misma. Como se explicaba en la capa anterior aquí se implementa una clase interfaz que se encarga de establecer un enlace con la capa lógica del negocio. Ya que es la capa encargada de establecer la conexión con la base de datos ella implementa el patrón de diseño Singleton cuyo principal objetivo y para el cual está diseñado es restringir la creación de objetos pertenecientes a una clase, logrando que una clase sólo tenga una instancia.

Ver Anexo 3

3.4 Modelo de Diseño

Al contrario del análisis, el diseño debe definir una solución software que satisfaga de modo efectivo y eficiente los requisitos especificados en el análisis, y al hacer esto el modelo incorporará nuevos artefactos (nuevas clases, nuevos atributos y operaciones para las clases, etc.) y tendrá en cuenta la plataforma tecnológica concreta sobre la que debe construirse el sistema informático. De hecho, el diseño debe proporcionar una solución creativa para el problema especificado en el análisis.

A continuación se representan las realizaciones de los principales escenarios de los casos de uso críticos, como pieza indispensable del modelo de diseño. El resto de los diagramas se encuentran en el *anexo 4*.

3.4.1 Diagramas de Secuencia



Fig.13 Diagrama de secuencia. CU Crear Documento. Escenario Insertar.

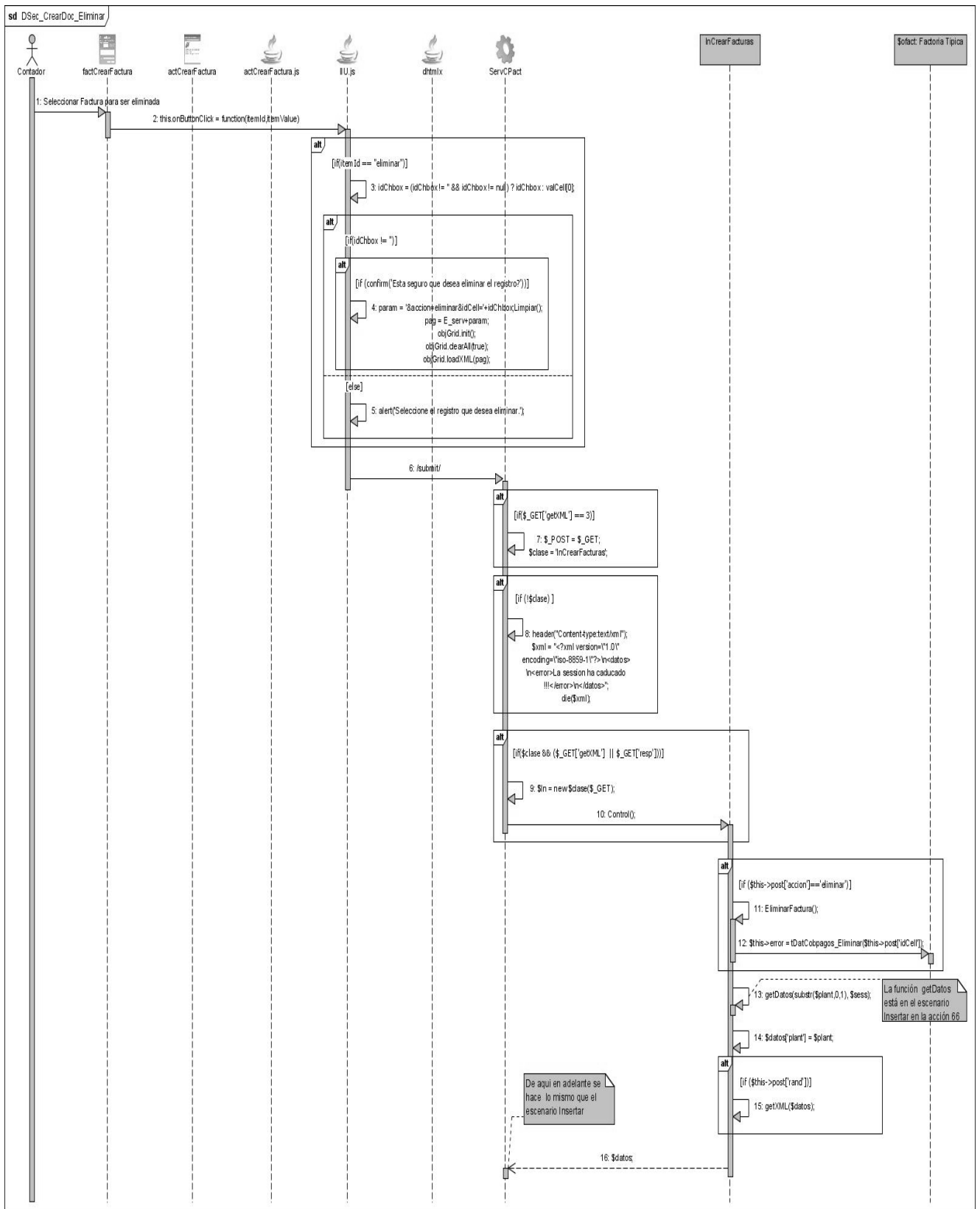


Fig.15 Diagrama de secuencia. CU Crear Documento. Escenario Eliminar.

3.4.2 Diagrama de clases del Diseño

Un diagrama de clases de diseño muestra la especificación para las clases software de una aplicación. En la siguiente figura, se representa una parte genérica a todos los diagramas de clases, además se realizó un diagrama de clases de diseño por caso de uso.

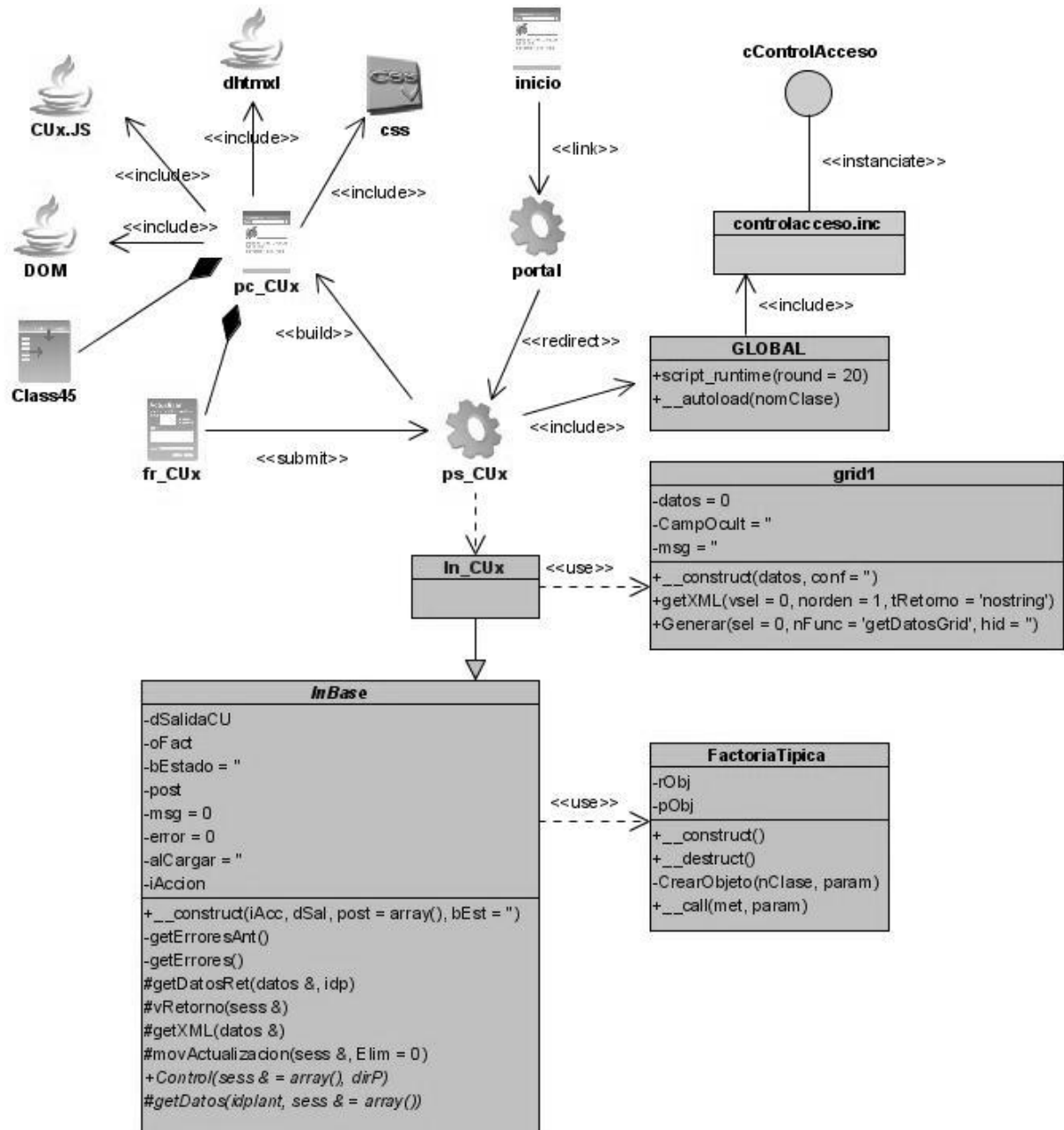


Fig.16 Diagrama de clases Genérico del Diseño.

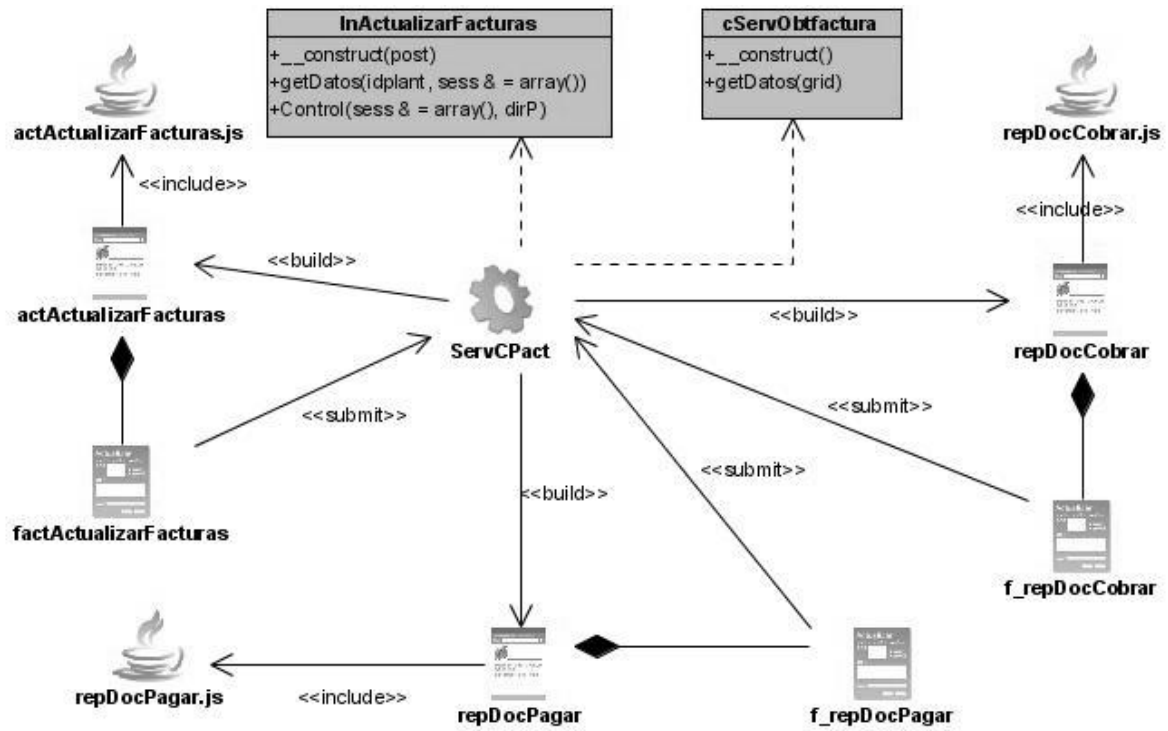


Fig.17 Diagrama de clases del Diseño, CU Obtener Documentos

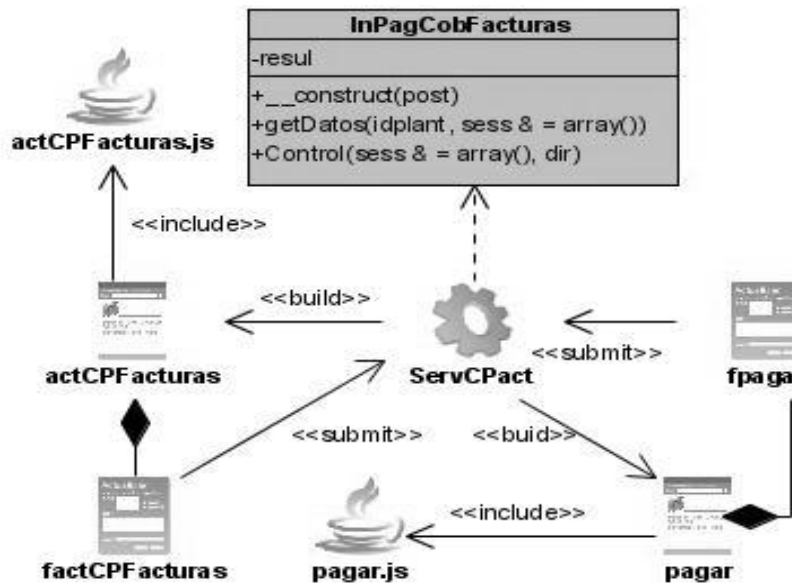


Fig.18 Diagrama de clases del Diseño, CU Actualizar Pago/Cobro de Documentos

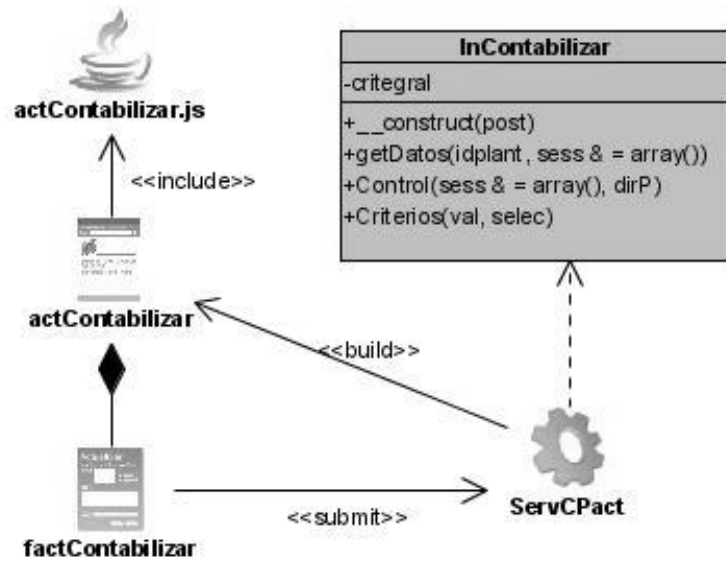


Fig.19 Diagrama de clases del Diseño, CU Contabilizar

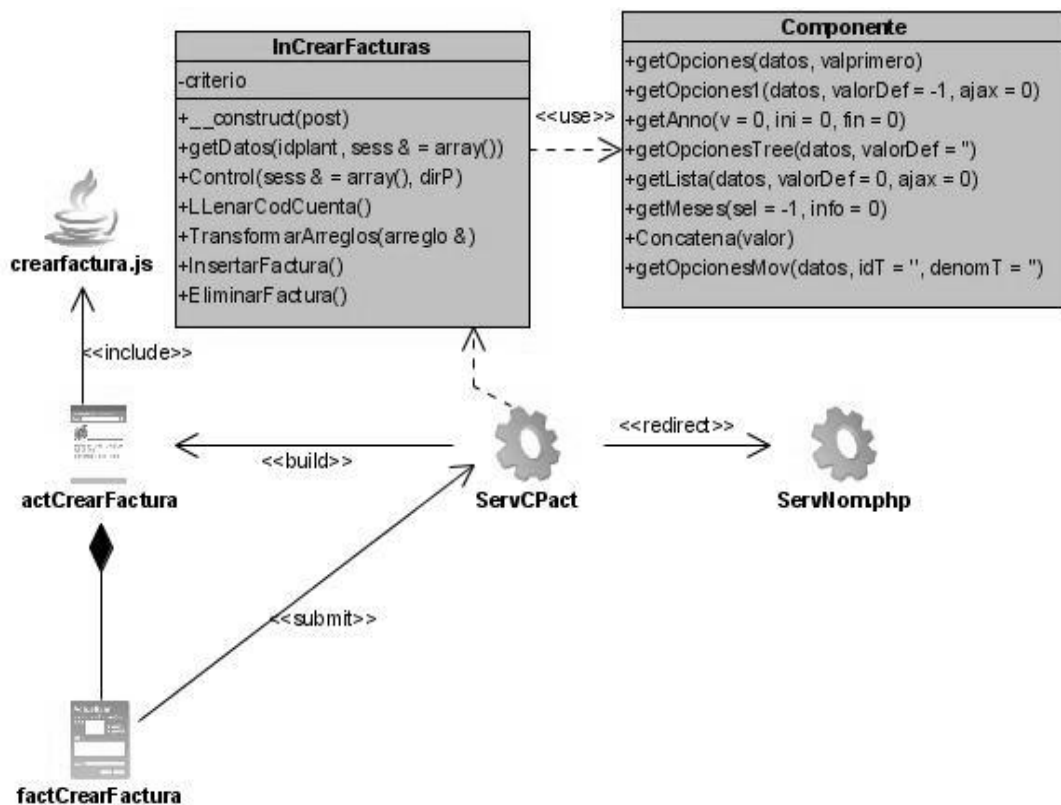


Fig.20 Diagrama de clases del Diseño, CU Crear Documento.

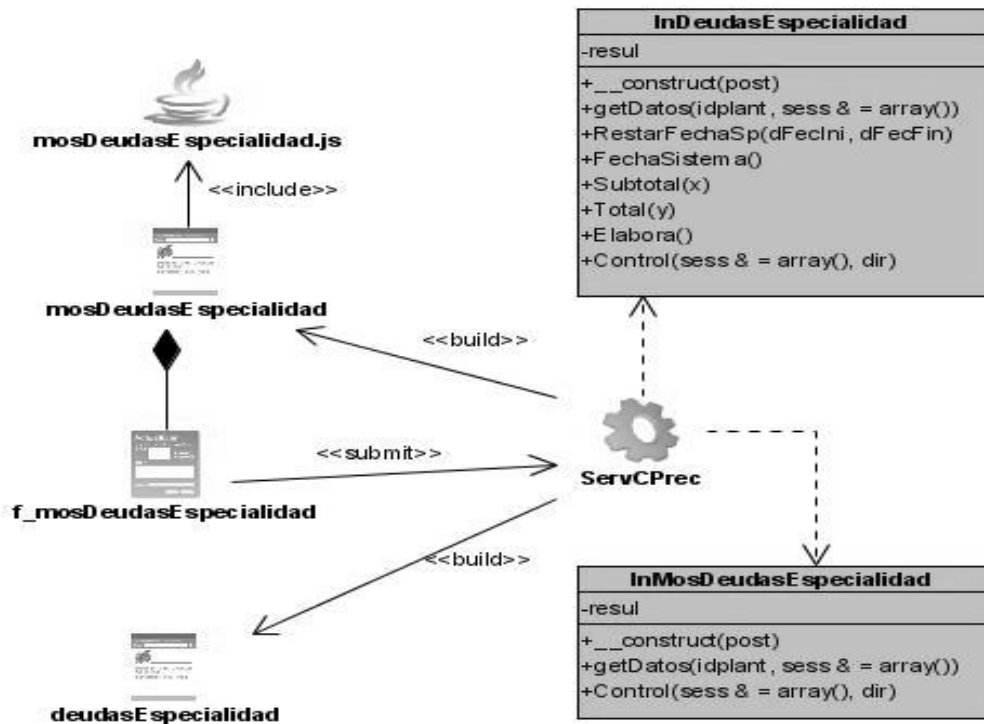


Fig.21 Diagrama de clases del Diseño, CU Mostrar Deudas por Especialidad.

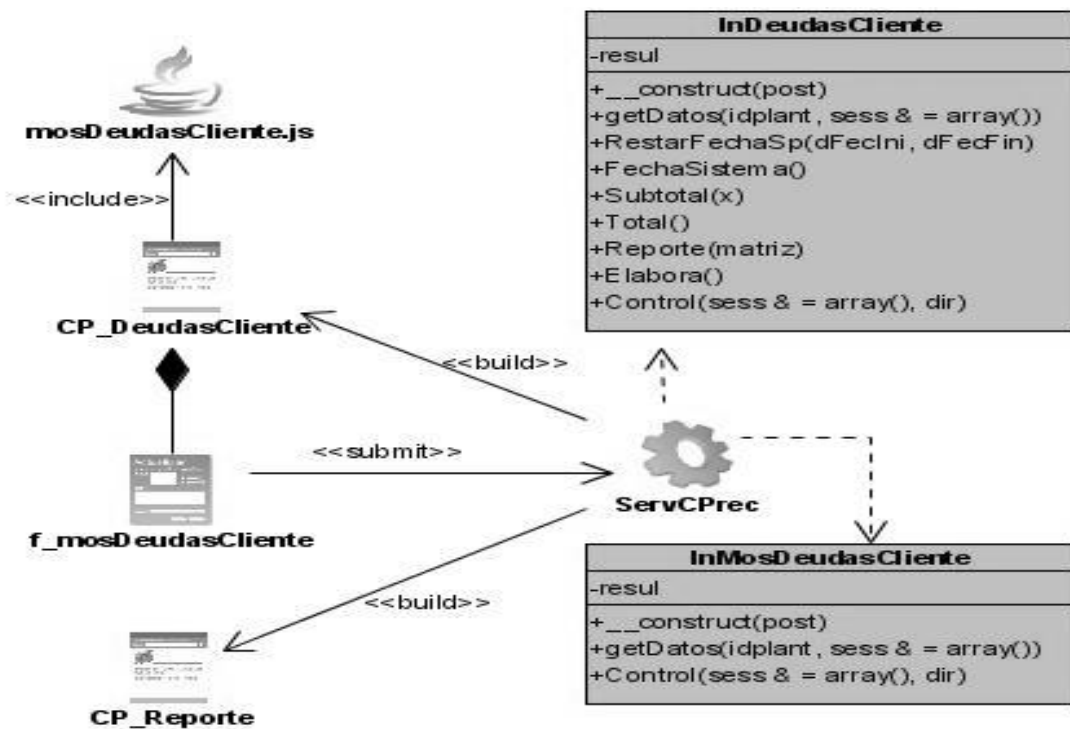


Fig.22 Diagrama de clases del Diseño, CU Mostrar Deudas por Cliente.

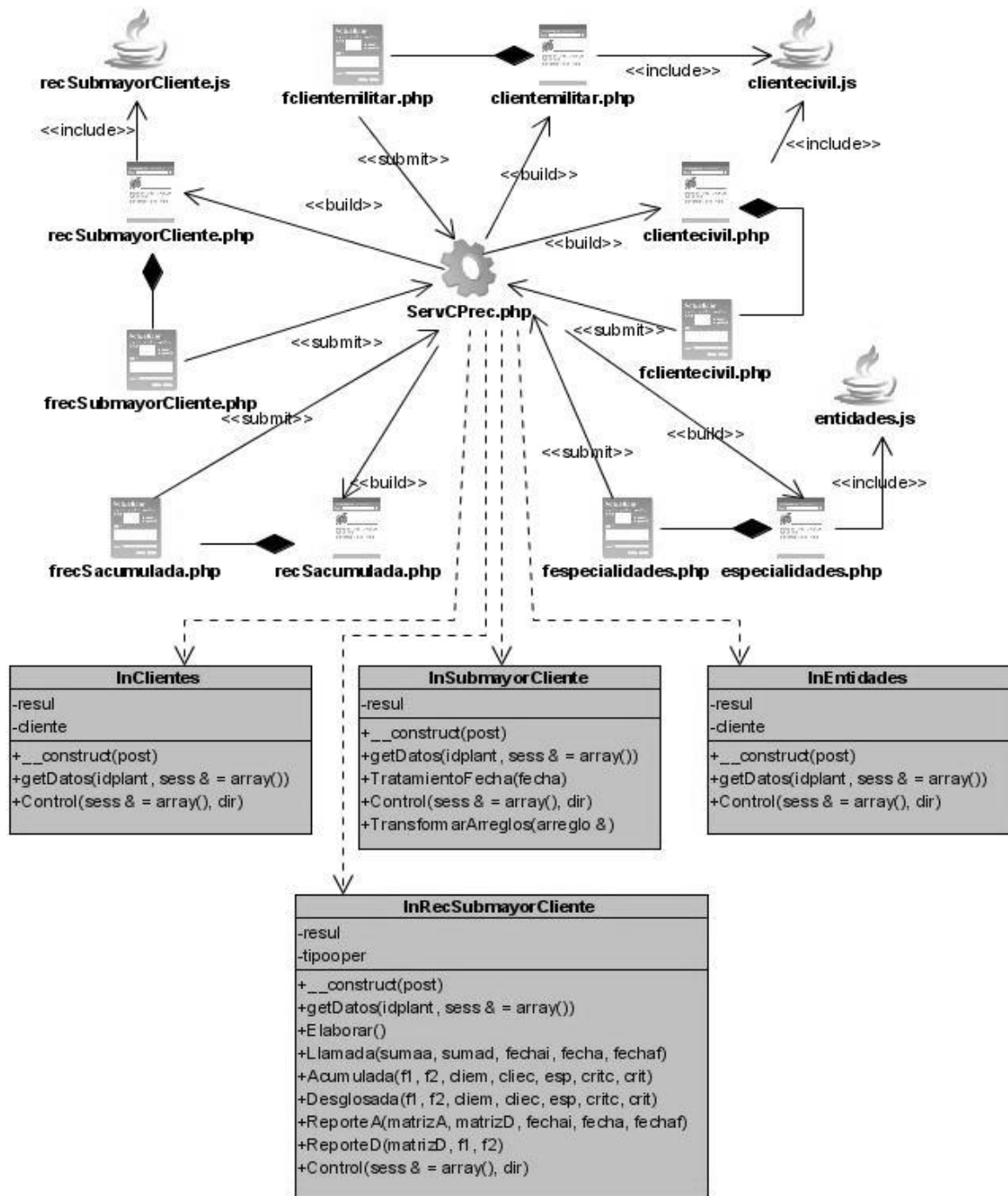


Fig.23 Diagrama de clases del Diseño, CU Mostrar Submayor por Clientes.

✓ Modelo Físico de Datos

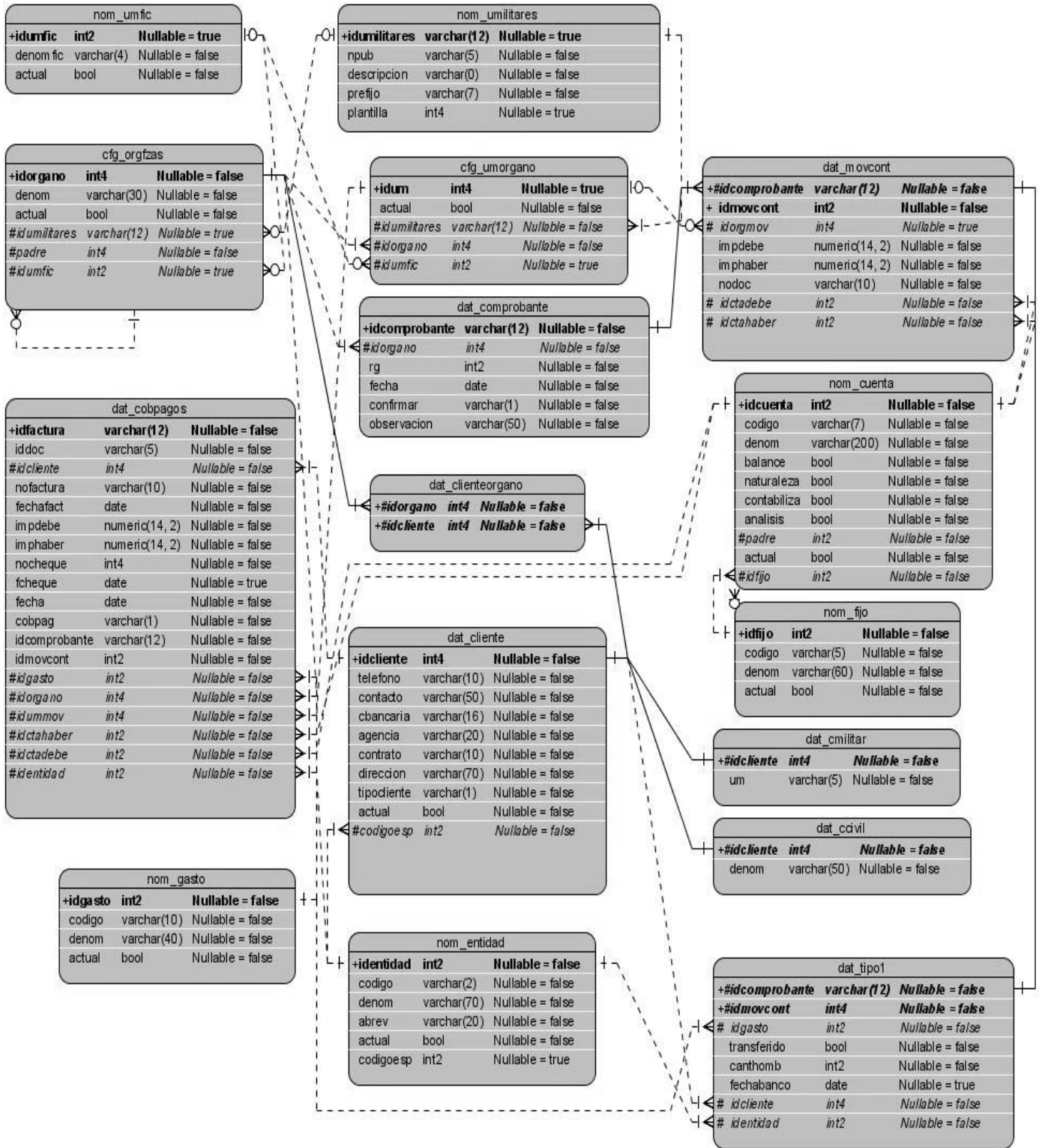


Fig.25 Modelo Entidad Relación de la Base Datos.

3.5.3 Descripción de las Tablas de la Base Datos

En esta sección se relacionan las tablas principales con la que se trabajarán para el manejo de la información, y se muestra la descripción y el tipo de atributo de cada uno de los campos de las tablas.

Nombre: dat_cobpagos		
Descripción: tabla elaborada para guardar los datos de los documentos por cobrar y cobrados, por pagar y pagados.		
Atributo	Tipo	Descripción
idfactura	varchar(12)	Se forma concatenando los valores (año+org+000+el último número aumentado en uno) Ej. 200700010001.
idorgano	integer	Viene de cfg_orgfzas, es el identificador del órgano financiero.
identidad	integer	Viene de nom_entidad, el identificador de la entidad.
idcliente	integer	Viene de dat_cliente, identificador del cliente.
idctadebe	smallint	Id de la cuenta, viene de nom_cuenta.
idctahaber	smallint	Id de la cuenta, viene de nom_cuenta.
idummov	integer	Viene de cfg_umorgano cuando es un órgano consumidor, cuando es un órgano administrador viene de cfg_orgfzas.
iddoc	varchar(5)	Viene de Contabilidad Material por el servicio web, se forma concatenando los valores iddoc+nrodoc+codigoL.
nofactura	varchar(10)	Número de la factura si es un servicio: iddoc+nrodoc sino es el número de la factura civil.
fechafact	date	Fecha de la factura.
impdebe	numeric(14,2)	Importe de la factura por el debe.
imphaber	numeric(14,2)	Importe de la factura por el haber.
nocheque	integer	Número de cheque se llena cuando se pague o cobre un documento.
fcheque	date	Fecha de cheque se llena cuando se introduzca el no cheque.

fecha	date	Fecha en que se recibe la factura en el órgano de finanzas (el año es el valor que se concatena para el idfactura).
cobpag	varchar(1)	Estado del documento: por cobrar 0, por pagar 1.
idcomprobante	varchar(12)	Id del comprobante, viene de dat_comprobante.
idmovcont	smallint	Viene de dat_movcont, identificador del movimiento contable.
idpresup	smallint	Viene de nom_presup, identificador del grupo presupuestario.
idgasto	smallint	Viene de nom_gasto, identificador del gasto o partida.

Nombre: dat_cliente		
Descripción: Tabla que guarda los datos comunes de los clientes militares y civiles.		
Atributo	Tipo	Descripción
Idcliente	integer	Identificador del cliente.
idprovincia	smallint	Viene de nom_provincia, identificador de la provincia.
idmunic	smallint	Municipio al que pertenece el cliente.
telefono	varchar(10)	Teléfono del cliente.
contacto	varchar(50)	A quien se puede localizar en ese lugar.
cbancaria	varchar(16)	Cuenta bancaria.
agencia	varchar(20)	Agencia a la que pertenece este cliente.
contrato	varchar(10)	Número del contrato.
direccion	varchar(70)	Dirección particular del cliente.
codigoesp	smallint	Código de la especialidad.
tipocliente	varchar(1)	Tipo de cliente P-Permanente, E-Eventual.
idmando	smallint	Viene de nom_mando, identificador del mando.
actual	bit(1)	Indicador de uso.

Nombre: dat_ccivil		
Descripción: Tabla que contiene clientes civiles acreedores o proveedores.		

Atributo	Tipo	Descripción
idcliente	integer	Viene de dat_cliente, identificador del cliente.
denom	varchar(50)	Denominación del cliente civil.

Nombre: dat_cmilitar		
Descripción: Tabla que contiene clientes militares de la Contabilidad Presupuestada y Empresarial.		
Atributo	Tipo	Descripción
idcliente	integer	Viene de dat_cliente, identificador del cliente.
denom	varchar(5)	Denominación de la unidad militar.

Nombre: dat_comprobante		
Descripción: Tabla que guarda para un registro y el órgano a que pertenece, la fecha y el tipo de operación.		
Atributo	Tipo	Descripción
idcomprobante	varchar(12)	Identificador del comprobante se forma concatenando los valores fecha (año)+idorgano+rg (Ej. 200700030001).
idorgano	integer	Órgano propio, viene del cfg_orgfzas, identificador del órgano financiero.
idoperacion	smallint	Viene de nom_operacion, identificador de la operación general.
idmodulo	smallint	Viene de nom_modulo, identificador del módulo.
rg	smallint	Número del comprobante se inicializa por año en 1.
fecha	date	Fecha de realización del comprobante (menor o igual a la fecha actual).
confirmar	varchar(1)	Marca de estado del comprobante 1: E. Confirmado, no se puede realizar ninguna modificación 0: E. En elaboración, permite modificar. 2: E. Terminado, se puede realizar modificación mientras no exista CO con fecha posterior.

observacion	varchar(50)	Observación asociada al comprobante de operaciones.
-------------	-------------	---

Nombre: dat_movcont		
Descripción: Tabla donde se registran los movimientos contables asociados a un registro y a un órgano financiero.		
Atributo	Tipo	Descripción
idcomprobante	varchar(12)	Viene de dat_comprobante, identificador del comprobante se forma concatenando los valores fecha (año)+idorgano+rg (Ej. 200700030001).
idmovcont	smallint	Identificador del movimiento contable.
idorgmov	integer	Identificador del movimiento contable por órgano financiero.
idtipooper	smallint	Viene de nom_tipooper, identificador de tipos de operaciones por cada operación general.
idctadebe	smallint	Viene de nom_cuenta.
idctahaber	smallint	Viene de nom_cuenta.
impdebe	numeric(14,2)	Importe de la cuenta al debe.
imphaber	numeric(14,2)	Importe de la cuenta al haber.
nodoc	varchar(10)	Número del documento: solo # hasta 10, puede empezar con 0.

Nombre: dat_tipo1		
Descripción: Tabla donde se guardan los detalles de los movimientos contables asociados a un comprobante.		
Atributo	Tipo	Descripción
idcomprobante	varchar(12)	Identificador del comprobante, viene de dat_movcont.
idmovcont	integer	Identificador del movimiento contable, viene de dat_movcont.
idpresup	smallint	Identificador del grupo presupuestario, viene del nom_presup.
identidad	smallint	Identificador de la entidad o especialidad, viene del

		nom_entidad.
idgasto	smallint	Identificador del gasto, viene del nom_gasto.
idingreso	smallint	Identificador de ingresos al presupuesto del estado, viene del nom_ingreso.
transferido	bit (1)	Marcar si es transferido o no el comprobante.
canthomb	smallint	Cantidad de personas que se le paga por nóminas.
idcliente	smallint	Identificador de los clientes militares y civiles, viene del dat_cliente.
idpersona	smallint	Identificador de las personas que tiene retención al banco, viene de la tabla persona.
idretencion	smallint	Identificador de retención bancaria, viene de nom_retencion.
fechabanco	date	Se guarda la fecha de banco.

Nombre: dat_clienteorgano		
Descripción: Tabla para relacionar los clientes que tiene un órgano financiero.		
Atributo	Tipo	Descripción
idorgano	integer	Viene de nom_NOorgano, identificador del órgano.
idcliente	integer	Viene de dat_cliente, identificador del cliente.

Nombre: nom_umilitares		
Descripción: Tabla que almacena las unidades militares que se necesitan para trabajar con la información.		
Atributo	Tipo	Descripción
idumilitares	varchar(12)	Este campo se calcula a través de la concatenación de los campos prefijo + plantilla, es el que identifica a las unidades militares.
prefijo	varchar(7)	Prefijo para trabajar con el nomenclador de plantilla, viene de nom_plantilla.
npub	varchar(5)	Viene de nom_plantilla.
descripcion	varchar	Número público de la UM, viene de plantilla.
plantilla	varchar(5)	Número de plantilla de la unidad militar.

Nombre: nom_entidad		
Descripción: Nomenclador de Entidad.		
Atributo	Tipo	Descripción
identidad	smallint	Identificador de entidad.
codigo	varchar(2)	Código de la entidad.
denom	varchar(70)	Denominación de la entidad.
abrev	varchar(20)	Abreviatura de la entidad.
actual	bit(1)	Indicador de uso.
codigoesp	smallint	Código de la especialidad, viene del nomenclador de especialidad.

Nombre: nom_cuenta		
Descripción: Nomenclador de cuentas contables.		
Atributo	Tipo	Descripción
idcuenta	smallint	Identificador de cuenta (auto numérico).
codigo	varchar(7)	El Código de la cuenta se forma con números y puntos (###.#.#).
denom	varchar(200)	Denominación de la cuenta.
balance	bit(1)	Para saber si es una cuenta de patrimonio (1) o de orden (0).
naturaleza	bit(1)	(0) si es una cuenta deudora (D-H). (1) si es una cuenta acreedora (H-D).
contabiliza	bit(1)	Si por esa cuenta se contabiliza (1), no se contabiliza por ella (0).
analisis	bit(1)	Si esa cuenta lleva análisis (1) (grupo, entidad, gasto, ingreso, costo, retenciones etc., no lleva análisis (0).
idgrupo	smallint	Identificador del grupo de cuentas, viene de nom_grupo.
padre	smallint	Código del padre de la cuenta.
idfijo	smallint	Viene de nom_fijo, identificador de categoría de las cuentas por grupos.
actual	bit(1)	Indicador de uso.

Nombre: nom_fijo		
Descripción: Nomenclador de categorías de las cuentas por grupos.		
Atributo	Tipo	Descripción
idfijo	smallint	Identificador de categoría de las cuentas por grupos.
codigo	varchar(5)	Código de la categoría de la cuenta.
denom	varchar(60)	Denominación de la categoría de la cuenta.
actual	bit(1)	Indicador de uso.

Nombre: nom_gasto		
Descripción: Nomenclador de Gastos en las FAR.		
Atributo	Tipo	Descripción
idgasto	smallint	Identificador del gasto o partida.
codigo	varchar(10)	Código del gasto.
denom	varchar(40)	Denominación del gasto.
actual	bit (1)	Indicador de uso.

Nombre: nom_umfic		
Descripción: Nomenclador de las unidades militares ficticias.		
Atributo	Tipo	Descripción
idumfic	smallint	Identificador de la unidad militar ficticia.
denomfic	varchar(4)	Denominación o número publico de la unidad militar ficticia.
actual	bit (1)	Indicador de uso.
denom	varchar(40)	Denominación de la unidad militar.

Nombre: cfg_orgfzas		
Descripción: Tabla para definir los órganos financieros y su nivel.		
Atributo	Tipo	Descripción
idorgano	integer	Identificador del órgano financiero.
idnivel	smallint	Viene de nom_nivel, identificador del nivel del órgano financiero.

denom	varchar(30)	Viene de nom_especialidad.
padre	integer	Determina el órgano administrador.
actual	bit(1)	Indicador de uso.
idumilitares	varchar(12)	Viene de nom_umilitares, este campo se calcula a través de la concatenación de los campos prefijo+plantilla, es el que identifica a las unidades militares.
idumfic	smallint	Identificador de la unidad militar ficticia, viene de nom_umfic.

Nombre: cfg_umorgano		
Descripción: Tabla para definir las unidades que atiende cada órgano de finanzas.		
Atributo	Tipo	Descripción
idum	integer	Identificador de la unidad militar perteneciente a un órgano financiero.
idumilitares	varchar(12)	Identificador de la unidad militar.
idorgano	integer	Viene de nom_orgfzas, identificador del órgano financiero.
actual	bit(1)	Indicador de uso.
idumfic	smallint	Identificador de la unidad militar ficticia, viene de nom_umfic.

3.6 Principios de diseño

El diseño del sistema está orientado directamente hacia los usuarios finales, por lo que uno de los puntos fundamentales que hay que tener en cuenta es el diseño de la interfaz, pues es la parte del sistema con que el usuario interactúa y que le facilita además el acceso a los recursos. Una aplicación con una interfaz bien diseñada debe tener, además de un buen diseño gráfico, una buena navegabilidad, usabilidad y distribución de los contenidos.

Para el diseño de la interfaz se tuvieron en cuenta los siguientes principios:

- ✓ Permitir su utilización desde el primer momento, por cualquier persona que tenga un mínimo dominio de la informática.
- ✓ Los elementos mostrados en la pantalla no serán numerosos.

- ✓ Cada elemento se diseñará siguiendo un patrón de tamaño, colores y formas.
- ✓ No se utilizarán colores brillantes.
- ✓ Los campos de texto con valores por defecto deben aparecer seleccionados, para que el usuario sólo tenga que teclear y no seleccionar todo, borrar y escribir.
- ✓ Los elementos que se repitan en las distintas páginas, se situarán en un mismo lugar.
- ✓ Permitir al usuario acceder solamente a las opciones que dado su rol, puede ejecutar.

3.7 Concepción de la Ayuda

Con el propósito de hacer más fácil para el usuario el trabajo con el software final se deriva la necesidad de brindarle al usuario el manual de usuarios para que se familiarice con el funcionamiento de cada una de las opciones de la aplicación antes de comenzar a trabajar con ella y la consulte cada vez que lo considere necesario, asegurando con esto que el uso del producto sea lo más eficiente posible.

3.8 Tratamiento de errores

En esta aplicación se trataron los errores de forma tal que las interacciones con la base de datos (inserción, eliminación, modificación, etc.) se realicen de forma correcta. Para ello se establecieron mecanismos de validación que comprueban la corrección de los datos a tratar, además en algunos formularios se insistió en que el usuario introduzca la menor cantidad posible de datos, evitando así incoherencias e incorrecciones en los mismos. En el caso de la entrada de datos por parte del usuario se implementó funciones que validan dicha entrada, de existir errores, se muestran mensajes que ilustran la incorrecta inserción, modificación o mala manipulación de datos en general y así le será posible al usuario rectificar el error y ejecutar nuevamente la acción de forma correcta.

3.9 Seguridad

De manera general, cuando se utiliza la Web para el modelado de sistemas se debe tener en cuenta que el acceso a los mismos debe ser de forma controlada, es decir que cada usuario tenga acceso sólo a la información que le concierne.

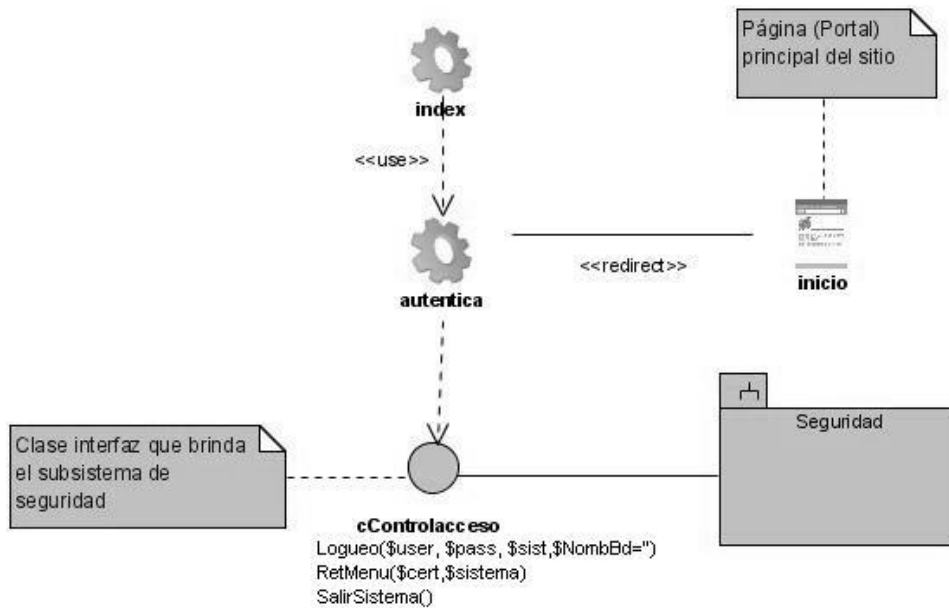


Fig.26 Mecanismo de diseño de seguridad.

El sistema en el cual está involucrada nuestra investigación, como ya se dijo anteriormente, es parte de un ERP y este está constituido por un conjunto de subsistemas dentro de los cuales se encuentra el de Seguridad, el cual proporciona una interfaz `cControlacceso` para acceder al servicio web que proporciona el subsistema para la autenticación de los usuarios en los diferentes módulos del ERP, esta interfaz contiene entre otros métodos uno público llamado logueo, el cual recibe como parámetro el usuario, la contraseña y el sistema al que desea acceder, como resultado se obtiene el acceso al sistema deseado o un mensaje de error en caso de existir algún problema, un método llamado RetMenu, el cual nos proporciona el menú de la aplicación con las opciones a las que puede acceder el usuario autenticado y un método llamado SalirSistema que su función no es más que salir del sistema.

3.10 Acceso a Datos

La figura 3.16 muestra el diagrama de clases de la solución propuesta para el acceso a datos del sistema. Se muestra la relación entre las clases así como sus responsabilidades.

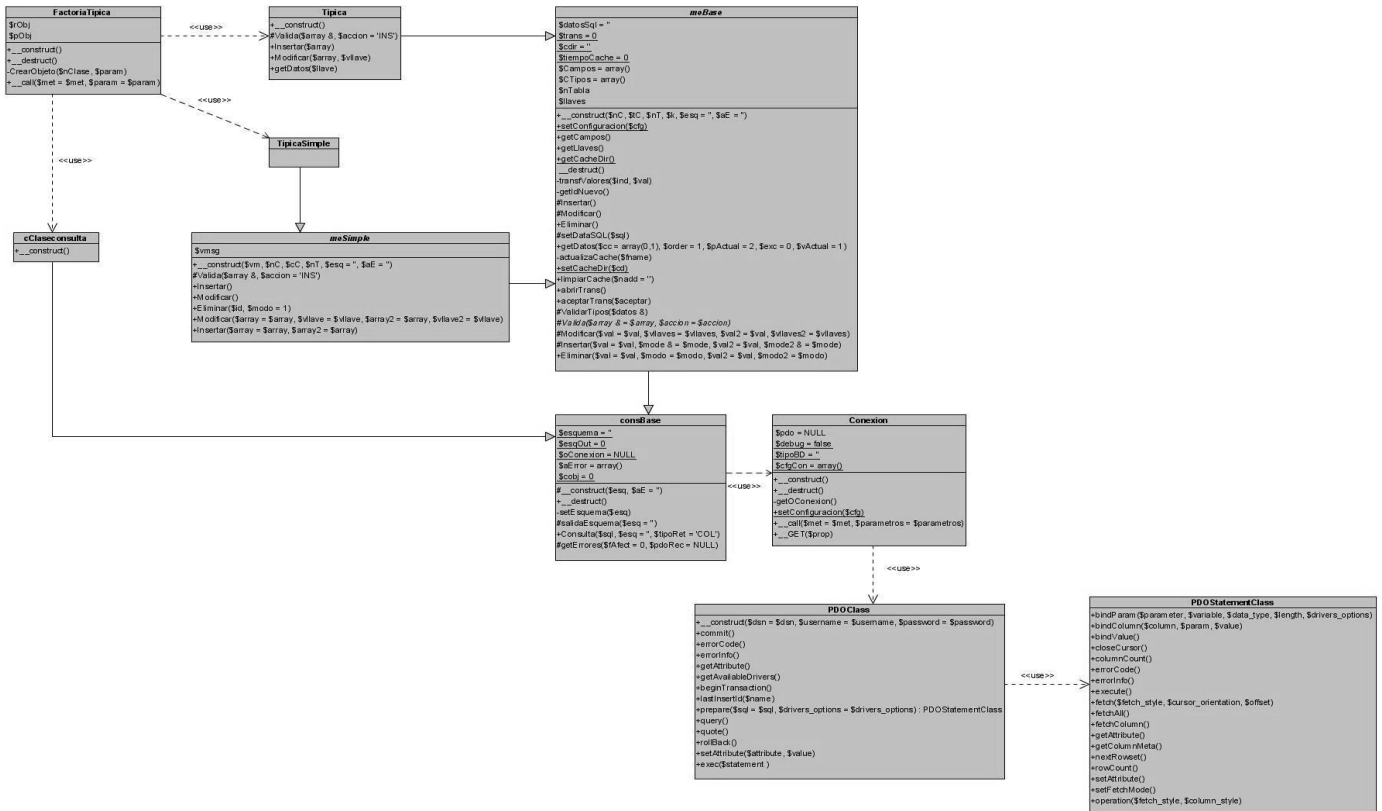


Fig.27 Mecanismo de diseño de acceso a datos.

Breve descripción de las clases

Factoría Típica: Clase que implementa la interfaz del modelo de persistencia con el resto de los subsistemas. A través de esta clase se crean y se manipulan los objetos de las típicas simples, los nomencladores y las demás típicas. Es una puerta entre la capa de Acceso a Datos y la capa de Lógica de Negocio.

Típicas: Clase que representa a las clases típicas de la aplicación, por cada entidad de la base de datos existe una clase típica que implementa generalmente un método de Validación, Insertar, Modificar y Eliminar. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway, que consiste en crear una instancia por cada tabla existente en la BD. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, insertar, modificar y eliminar. Hereda de la clase abstracta meBase.

Típica Simple: Es una clase que representa a las clases típicas (nomencladores simples) en general de la aplicación. Estas típicas son de una implementación muy sencilla, pues la mayoría de las líneas que normalmente habían que codificar quedaron encapsuladas en la clase base de las mismas. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway, que consiste en crear una instancia por cada tabla existente en la BD. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, insertar, modificar y eliminar. Hereda de la clase abstracta meSimple.

cClaseconsulta: Clase que representa a las clases de la aplicación. Existe una clase consulta para cada entidad de la base de datos, éstas heredan de la clase abstracta consBase.

meSimple: Clase abstracta heredera de meBase, y a la vez base para la implementación de las típicas que responderán a los nomencladores simples del modelo de persistencia dado. Redefine las operaciones básicas con la funcionalidad de Validación dada. Redefine las operaciones básicas que pudieran realizarse a una entidad (insertar, eliminar, modificar) para los nomencladores simples.

meBase: Clase abstracta que hereda de consBase, es la base para el resto de las que implementan funcionalidades para el trabajo con las entidades del sistema a implementar. Implementa las operaciones básicas que pudieran realizarse a una entidad (insertar, eliminar y modificar). Y hereda de consBase la operación de Consulta.

consBase: Esta clase es la base en toda la jerarquía de Acceso a Datos y es empleada para aportar contenido dinámico a las plantillas. Encapsula el objeto conexión. Implementa la operación de Consulta.

Conexión: Esta clase es la encargada de establecer la conexión con el servidor de la base de datos a través de un objeto PDO de la librería de PHP.

PDO: Es un modelo de acceso a datos para php que brinda una capa de abstracción para el acceso a Base de Datos desde PHP.

3.11 Diagrama de despliegue

En el mismo se sitúa el software en el hardware que lo contiene. Cada hardware se representa como un nodo. Un nodo se representa como un cubo, es un elemento donde se ejecutan los componentes, representan el despliegue físico de los mismos.

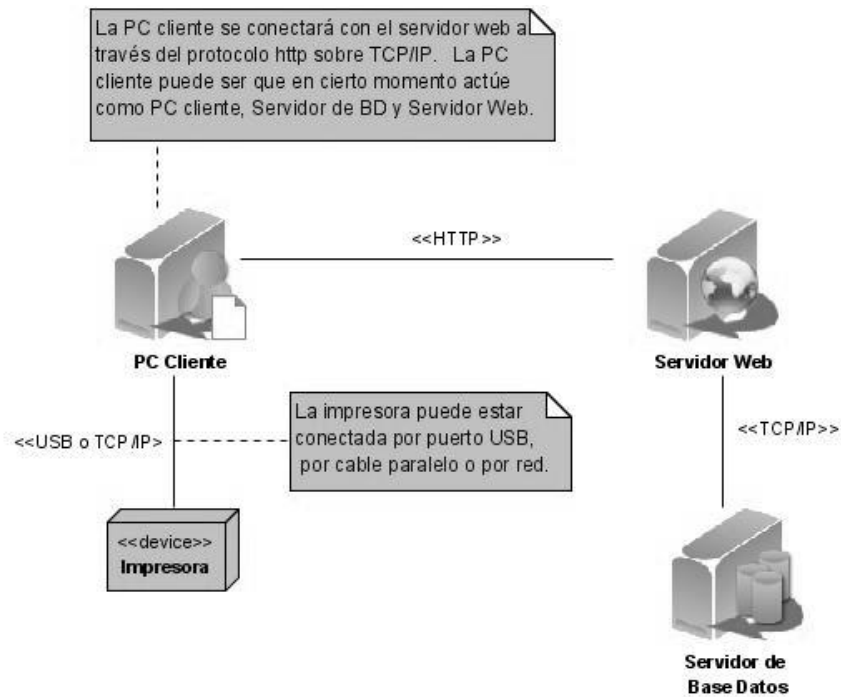


Fig.28 Diagrama de Despliegue.

3.12 Conclusiones

Con la culminación de este capítulo se ha dado un gran paso hacia la implementación del sistema, pues ya se definieron los mecanismos de diseño a utilizar para la elaboración del mismo y se tienen las clases persistentes de datos que se generan en todo este proceso, además mediante el diagrama de despliegue se conoce la situación física del software realizado.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.

4.1 Introducción

Con los resultados arrojados en el capítulo anterior se comenzará la implementación y las pruebas del sistema. Se mostrará el diagrama de componentes reflejando las clases del diseño en términos de componentes y la relación entre los distintos paquetes y subsistemas. Al final se definirán los diferentes casos de pruebas que guiarán los procedimientos para evaluar resultados de la implementación.

4.2 Implementación

El modelo de implementación describe cómo los elementos del modelo de diseño, se implementan en términos de componentes, ficheros de código fuente, ejecutables, etc. Este modelo describe además cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizado, y cómo dependen los componentes unos de otros.

4.2.1 Diagrama de Componente

Los diagramas de componentes describen la organización y dependencias lógicas entre componentes software, siendo éstos, componentes de código fuente, ejecutables, librerías, tablas, entre otros. Se utiliza para modelar la vista estática de un sistema. Muestra la organización y las dependencias lógicas entre un conjunto de componentes software.

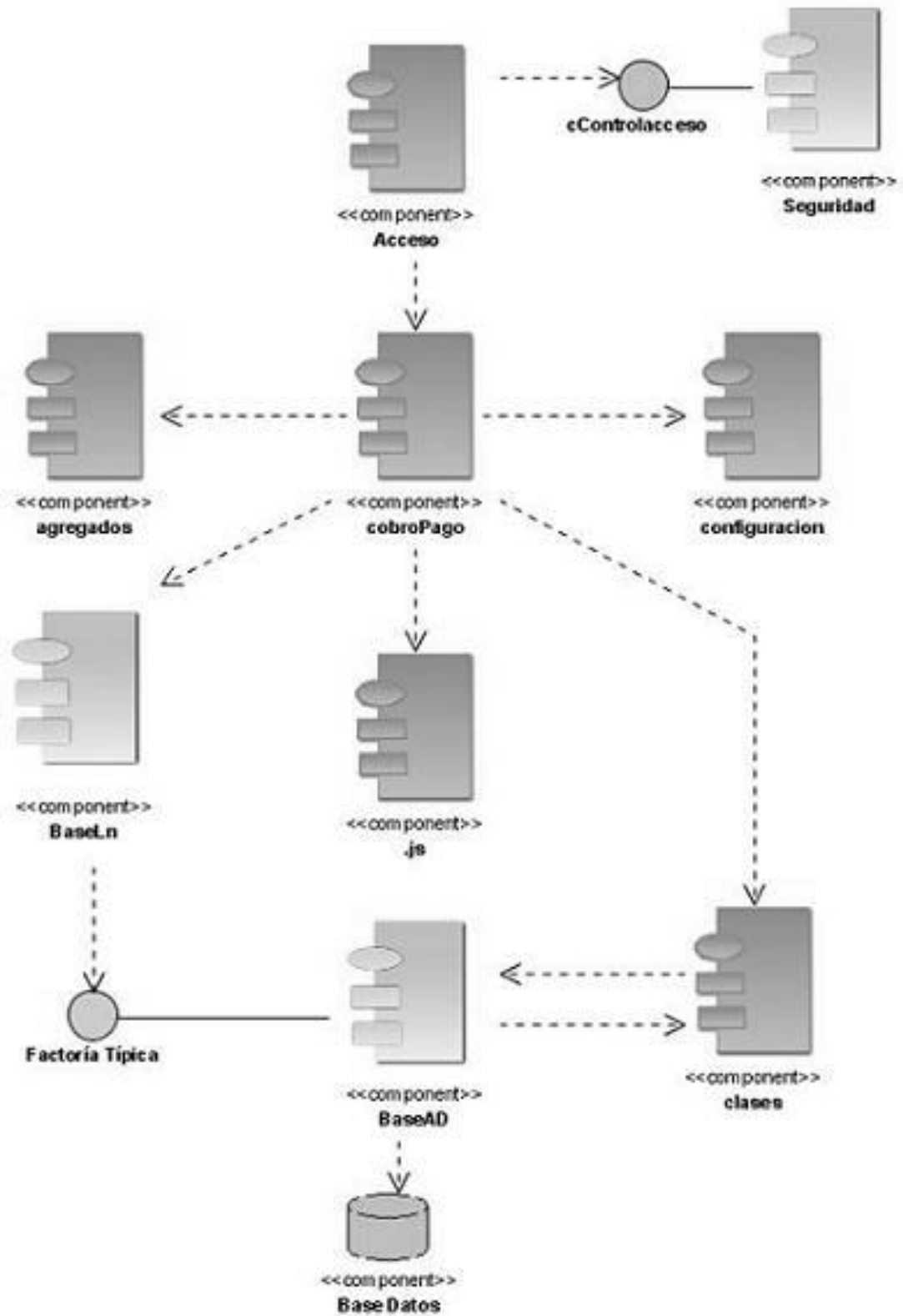


Fig.29 Diagrama de Componente del subsistema de Contabilidad Financiera con el módulo de Cobros-Pagos.

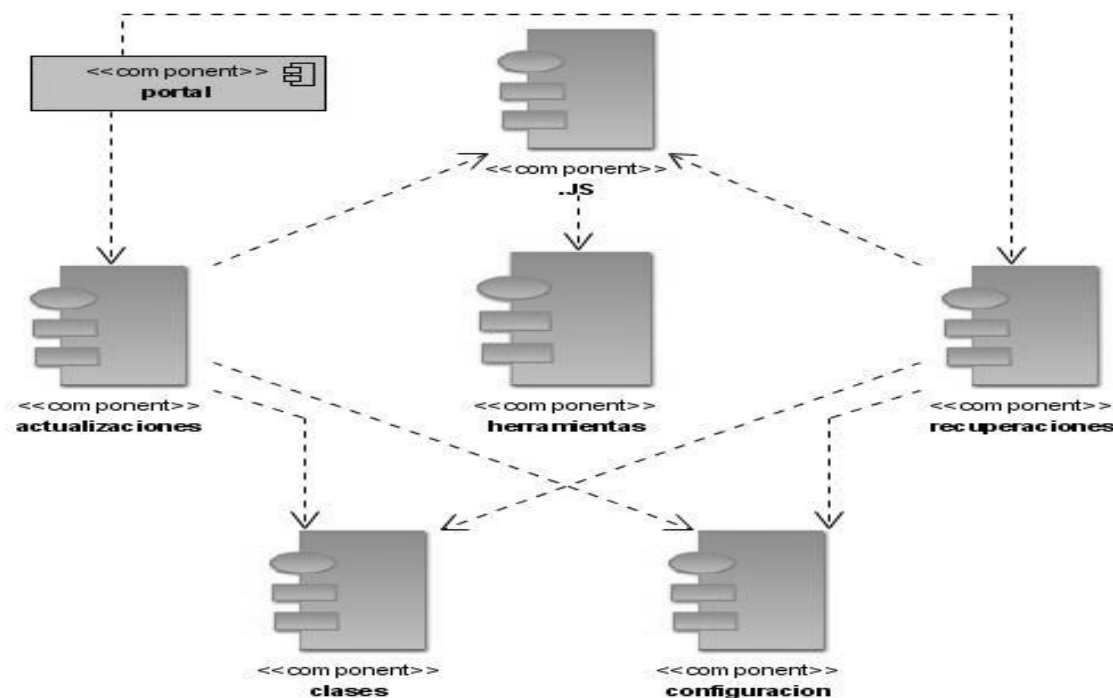


Fig.30 Diagrama de Componente módulo de Cobros-Pagos.

El resto de los diagramas de componente se encuentran en el *anexo 5*.

4.3 Modelo de prueba

Una de las últimas fases del ciclo de vida del desarrollo de un software es el flujo de trabajo de pruebas, dicha actividad se lleva a cabo cuando se ha obtenido un resultado después de la implementación y está encaminada a encontrar errores en el software.

Las pruebas se pueden hacer a distintos niveles, tales como unidad, integración, sistema y aceptación. La prueba de caja negra es una prueba de sistema y se centran en los requisitos funcionales del software, permitiendo obtener un conjunto de condiciones de entrada que incluyan completamente los requisitos funcionales del programa.

A continuación se desarrollan las pruebas de caja negra al software final, para esto fueron definidos diferentes casos de prueba por varios casos de uso del sistema.

4.3.1 Casos de Prueba

Nombre CU: Crear Documento

1. Introducir todos los valores para crear un documento.

2. Campos obligatorios sin datos.
3. Modificar un documento y no introducir los campos obligatorios.

CU Crear Documento		
Entrada	Resultados	Condiciones
	El sistema muestra la interfaz para crear un nuevo documento.	
-1- Tipo Operación: "Cobro" Concepto: "Otros Gastos" No Factura: "3000" Fecha de la Factura: "06/06/2008" U.Militares: "1390" Especialidad: "03 - c" Grupo Presupuestario: "3453 - SDFG" Gasto: "k2 - 2" Cliente: "1392" Cuenta Debe: "412" Importe al Debe: "8888" Cuenta Haber: "412" Importe al Haber: "7777"	El sistema actualiza el grid, y los select de la interfaz con los datos del nuevo documento, refresca la página para si el contador desea introducir un nuevo documento.	No existe ese documento en la aplicación.
-2- Tipo Operación: "" Concepto: "" No Factura: "" Fecha Factura: "" U.Militares: "" Especialidad: "" Cliente: "" Cuenta Debe: "" Importe al Debe: "" Cuenta Haber: ""	El sistema muestra un mensaje de alerta informando que se deben introducir los datos faltantes.	No se registran valores vacíos en campos obligatorios del documento.

Importe al Haber: ""		
-3- Factura: "2000" Tipo Operación: "" Concepto: "" No Factura: "" Fecha Factura: "" U.Militares: "" Especialidad: "" Cliente: "" Cuenta Debe: "" Importe al Debe: "" Cuenta Haber: "" Importe al Haber: ""	El sistema muestra un mensaje de alerta informando que se deben introducir los datos faltantes.	No se registran valores vacíos en campos obligatorios del documento.

Nombre CU: Actualizar Pago/Cobro de Documentos.

1. Introducir todos los valores para pagar o cobrar un documento.
2. Campos obligatorios sin datos.

CU Actualizar Pago/Cobro de Documentos		
Entrada	Resultados	Condiciones
	El sistema muestra la interfaz de Actualizar Cobros o Pagos de un documento.	
-1- Tipo de Operación: "Cobro" No Cheque: Fecha: Cuenta de Banco:	El sistema actualiza el grid.	Se debe seleccionar al menos un documento para pagar o cobrar.
-2- Tipo de Operación: "" Nro de Cheque: "" Fecha: ""	El sistema muestra un mensaje de alerta informando que se debe seleccionar el tipo de operación que se desea filtrar.	No se registran valores vacíos en campos obligatorios del documento.

Cuenta de Banco:""		
--------------------	--	--

Nombre CU: Mostrar Deudas por Cliente

1. Introducir todos los valores para mostrar el reporte Deudas por Cliente.
2. Campos obligatorios sin datos.

CU Mostrar Deudas por Cliente		
Entrada	Resultados	Condiciones
	El sistema muestra la interfaz de Mostrar Deudas por Cliente.	
-1- Tipo de Operación: "Cobro" Criterio: "Otros Gastos" Tipo Cliente: "Militar"	El sistema muestra el reporte con los datos que se filtraron.	Existen los datos en la Base Datos.
-2- Tipo de Operación:""	El sistema muestra un mensaje de alerta informando que se debe seleccionar el tipo de operación que se desea filtrar.	No se registran valores vacíos en campos obligatorios.

Nombre CU: Mostrar Deudas por Especialidad.

1. Introducir todos los valores para mostrar el reporte Mostrar Deudas por Especialidad.
2. Campos obligatorios sin datos.

CU Mostrar Deudas por Especialidad		
Entrada	Resultados	Condiciones
	El sistema muestra la interfaz Mostrar Deudas por Especialidad.	
-1- Tipo de Operación: "Cobro" Criterio: "Otros Gastos" Especialidad: "e"	El sistema muestra el reporte con los datos que se filtraron.	Existen los datos en la Base Datos.
-2-	El sistema muestra un mensaje de	No se registran valores vacíos

Capítulo 4: Implementación y Prueba

Tipo de Operación:""	alerta informando que se debe seleccionar el tipo de operación que se desea filtrar.	en campos obligatorios.
----------------------	--	-------------------------

Nombre CU: Mostrar Submayor por Cliente

1. Introducir todos los valores para mostrar el reporte Submayor por Cliente.
2. Campos obligatorios sin datos.

CU Mostrar Submayor por Cliente		
Entrada	Resultados	Condiciones
	El sistema muestra la interfaz Mostrar Submayor por Cliente.	
-1- Fecha Inicial: "01/06/2008" Fecha Final: "05/06/2008" Tipo de Operación: "Cobro" Criterio: "Otros Gastos" Tipo Cliente: "Militar" Especialidad: "3" "Suma Acumulada"	El sistema muestra el reporte con los datos que se filtraron.	Existen los datos en la Base Datos.
-2- Tipo de Operación:"" Cliente:"" Especialidad:"" Suma Acumulada y Suma Desglosada:""	El sistema muestra un mensaje de alerta informando que se debe seleccionar el tipo de operación que se desea filtrar.	No se registran valores vacíos en campos obligatorios.

Nombre CU: Contabilizar

1. Introducir todos los valores para Contabilizar.
2. Campos obligatorios sin datos.

CU Contabilizar		
Entrada	Resultados	Condiciones

	El sistema muestra la interfaz de Contabilizar.	
-1- Tipo de Operación: "Cobro" Concepto: "Fondo de Operaciones"	El sistema muestra un grid con los documentos pendientes a contabilizar.	Existen los datos en la Base Datos.
-2- Tipo de Operación:""	El sistema muestra un mensaje de alerta informando que se debe seleccionar el tipo de operación que se desea filtrar.	No se registran valores vacíos en los campos obligatorios.

4.4 Conclusiones

Con la culminación de este capítulo, mediante la realización del diagrama de componentes se conocen las relaciones de dependencias entre los componentes y las interfaces que estos soporten. Se hicieron además pruebas de sistema garantizando el correcto cumplimiento de los requerimientos funcionales que se debían cumplir.

CONCLUSIONES

Con la realización de este trabajo, se obtuvo un software funcional con las características apropiadas para ser desplegado en las unidades de las FAR e integrado al subsistema de Contabilidad Financiera que forma parte del ERP. Dicho software cumple con los requerimientos funcionales y no funcionales previstos en la fase de inicio. Con la aplicación propuesta se realizan los principales procesos de Cobros y Pagos de forma rápida y eficiente constituyendo la misma un importante apoyo para el correcto desempeño y calidad del trabajo de los contadores en las FAR.

El proceso de desarrollo del sistema estuvo regido por los procesos de RUP, se utilizó para la modelación UML, como lenguaje de programación PHP, PostgreSQL como sistema gestor de base de datos, y para el diseño del sitio se utilizó el Dreamweaver MX.

Este trabajo nos proporcionó la posibilidad de aplicar todos los conocimientos de Ingeniería de Software y Programación adquiridos durante los 5 años de carrera.

Por lo tanto afirmamos que todos los objetivos propuestos fueron alcanzados satisfactoriamente.

RECOMENDACIONES

- ✓ Desplegar el software en las condiciones especificadas.
- ✓ Modelar el proceso de traspaso de las facturas liquidadas a un histórico.
- ✓ Modelar e implementar los procesos de cobros y pagos anticipados.
- ✓ Realizar una segunda iteración de desarrollo para perfeccionar el software construido.

REFERENCIAS BIBLIOGRÁFICAS

Citadas

[1] ¿Qué es un ERP?

<http://www.adpime.com> (13/1/2008).

[9] *8 Lenguajes de Programación que deberías aprender*, 2007. Disponible en:

<http://www.tufuncion.com/diferentes-lenguajes-programacion> (11/03/2008)

Consultadas

[2] *Herramientas Web para la enseñanza de Protocolos de Comunicación. El modelo cliente – servidor.*

Disponible en:

<http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html> (21/01/2008)

[3] Introducción a las Aplicaciones Web. Disponible en:

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node11.html> (15/4/2008)

[4] Guía Breve de Servicios Web. Disponible en:

<http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>

[5] Página oficial. *Visual Paradigm*. Disponible en:

<http://www.visual-paradigm.com/product/vpuml/> (21/03/2008)

[6] Torres, Ariel. *Que podemos hacer con PHP?*, 2007. Disponible en:

http://www.articuloweb.com/category.php?cat_id=3 (10/02/2008)

[7] de la Torre, Aníbal. *Lenguajes del lado del Servidor o Cliente*, 2006. Disponible en:

http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html

(11/03/08)

[8] *Lenguajes de Programación para páginas Web HTML*, 2007. Disponible en:

<http://www.monografias.com/trabajos7/html/html.shtml> (11/03/2008)

[10] El servidor de web Apache: Introducción práctica. Disponible en:

<http://acsblog.es/articulos/trunk/LinuxActual/Apache/html/index.html>

[11] ¿Qué es Postgres?

<http://www.postgres-ql.com.ar/html/informacion.php?opcion=queespostgres> (19/3/2008).

[12] DesarrolloWeb. Disponible en: <http://www.desarrolloweb.com/articulos/332.php> (10/4/2008)

BIBLIOGRAFÍA

Manual de Contabilidad Financiera para la Actividad Presupuestada en las FAR.

Jacobson, Ivar. *El Proceso Unificado de Desarrollo del Software Volumen I*, La Habana, Félix Varela, 2004.

Jacobson, Ivar. *El Proceso Unificado de Desarrollo del Software Volumen II*, La Habana, Félix Varela, 2004.

Presuman, Roger. *Ingeniería de Software, Un enfoque práctico Parte 1*, La Habana, Félix Varela, 2005.

Booch, G., Rumbaugh, J., Jacobson, I. *El Lenguaje Unificado de Modelado*. Addison-Wesley. 1999.

Brocard Delfino, Yusniel, Frometa Moreno, Yoiler Joaquín, Suárez Corrales, Maikel. *Diseño y Servicios Web para el Registro de Población de la Atención Primaria del Sistema de Información para la Salud*. Universidad de las Ciencias Informáticas, 2007.

GLOSARIO DE TÉRMINOS

Unidades presupuestadas: Son las entidades que reciben del estado los recursos destinados a financiar los diferentes servicios: educación, salud, actividades socioculturales, científicas, administrativas, defensa y orden interior debiendo mantener el control de los mismos, son aquellas que se dedican a recibir, ejecutar y controlar los recursos financieros que el estado brinda para financiar sus actividades.

Cobros y Pagos: Pagos son las salidas de dinero y cobros las entradas que se produzcan en la tesorería de la empresa. Así, en principio, ante el alta o la baja de algún elemento patrimonial, la empresa paga o cobra dinero.

Cheque: Instrumento de pago que expide la entidad emisora y que constituye una orden para que, la Sucursal Bancaria con que ésta opera, realice la transferencia monetaria de su cuenta a la del beneficiario. Los cheques tienen diferentes clasificaciones si resulta necesaria su previa habilitación o certificación.

Factura: Es un documento que refleja la entrega de un producto o la provisión de un servicio, junto a la fecha de devengo, además de indicar la cantidad a pagar como contraprestación.

En la factura se encuentran los datos del expedidor y del destinatario, el detalle de los productos y servicios suministrados, los precios unitarios, los precios totales, los descuentos y los impuestos.

Cuenta: Registro que contiene las operaciones de igual naturaleza y la fecha de cada una de ellas en orden cronológico. Las anotaciones en una cuenta se expresan en débitos y créditos, evaluados en términos monetarios, y muestran el saldo actual en caso de existir.

Contabilidad Financiera: Es la técnica mediante la cual se recolecta, se clasifica, se registra, se resume y se informa de las operaciones cuantificables en dinero, realizadas por una entidad económica. También proporciona estados financieros que son sujetos al análisis e interpretación, informando a los administradores, a terceras personas y a oficinas gubernamentales del desarrollo de las operaciones de la empresa.

Metalinguaje: Es un lenguaje usado para hacer referencia a otros lenguajes. Los modelos formales de sintaxis para la descripción de la gramática, como por ejemplo, la gramática generativa, son tipos de metalinguaje. En un aspecto más general, puede referirse a cualquier terminología o lenguaje

usado para discutir acerca del mismo lenguaje - un texto sobre gramática, por ejemplo, o una discusión acerca del uso del lenguaje.

ANEXOS

Anexo 1. Arquitectura Cliente Servidor

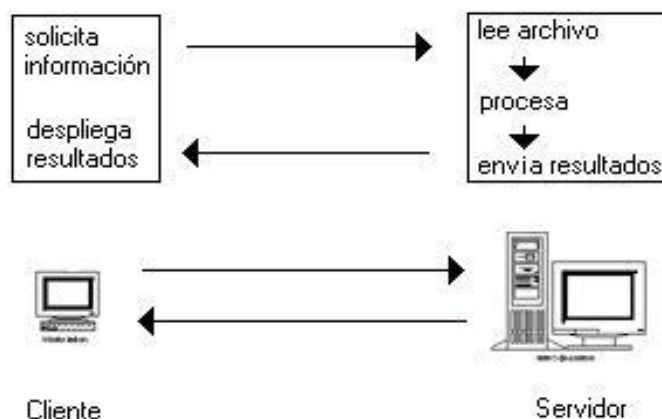
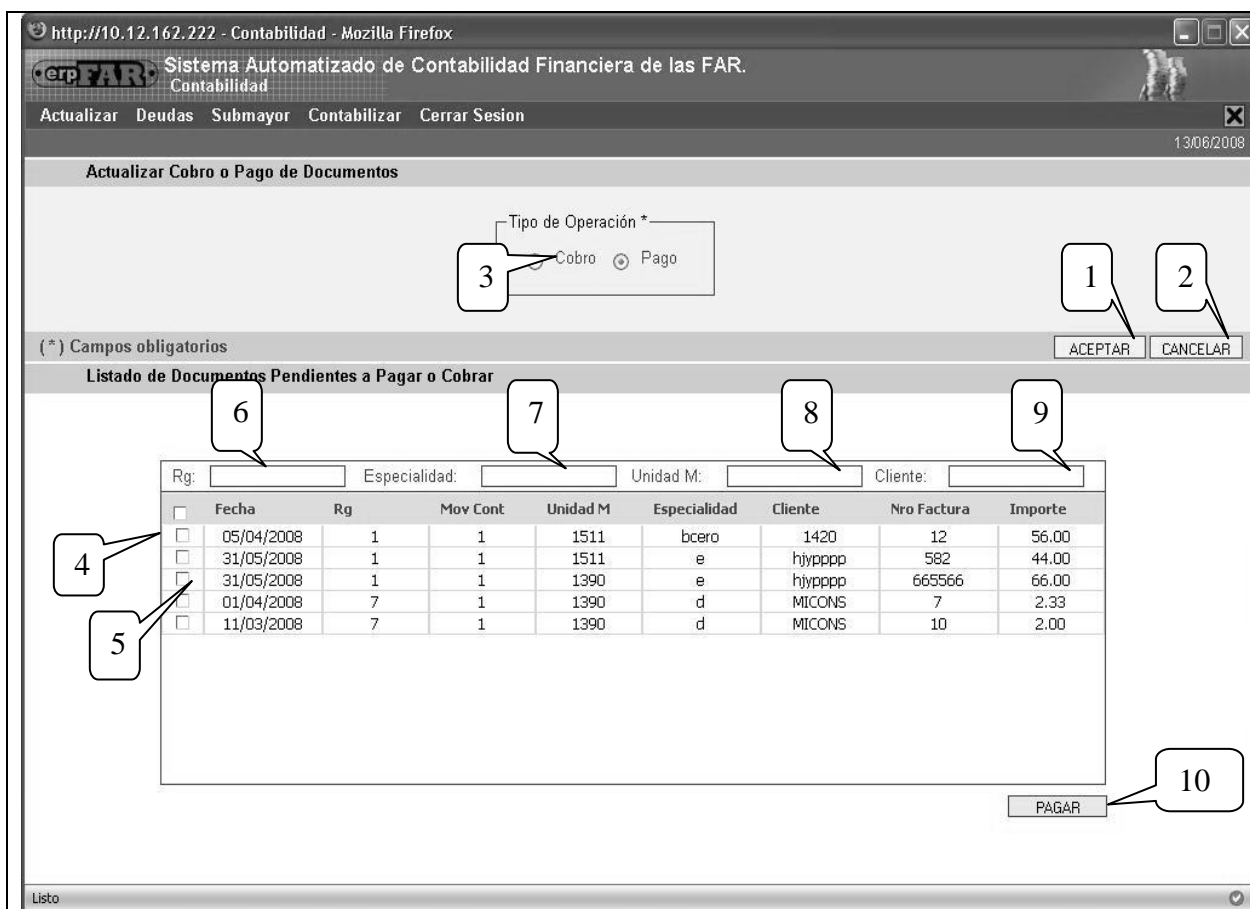


Fig.31 Arquitectura Cliente Servidor

Anexo 2. Descripciones de los casos de usos.

Caso de Uso	Actualizar Pago/Cobro de Documentos
Actores	Contador
Propósito	Actualizar el pago o cobro de uno o varios documentos asignándole un número de cheque y registrándolo en un comprobante de operaciones.
Resumen	El contador inicia el caso de uso cuando decide realizar el pago de uno o varios documentos. Se termina el caso de uso cuando se realice la acción.
Responsabilidades	R3, R3.1, R3.2, R3.3, R3.4, R3.5
CU asociados	
Precondiciones	
Requisitos especiales	
Descripción	
Interfaz I.	



1. Control de aceptar una acción.

Nombre: Aceptar. Tipo: Botón.

2. Control de cancelar una acción.

Nombre: Cancelar. Tipo: Botón.

3. Control para seleccionar el tipo de operación por la cual que se desea filtrar.

Nombre: Tipo de Operación. Tipo: Radiobutton.

4. Control para mostrar los documentos pendientes a pagar.

Nombre: Grid. Tipo: Grid.

5. Control para marcar el o los documentos, se pueden seleccionar todos además.

Nombre: Chek box. Tipo: Chek box.

6. Control para buscar el documento por el número del registro.

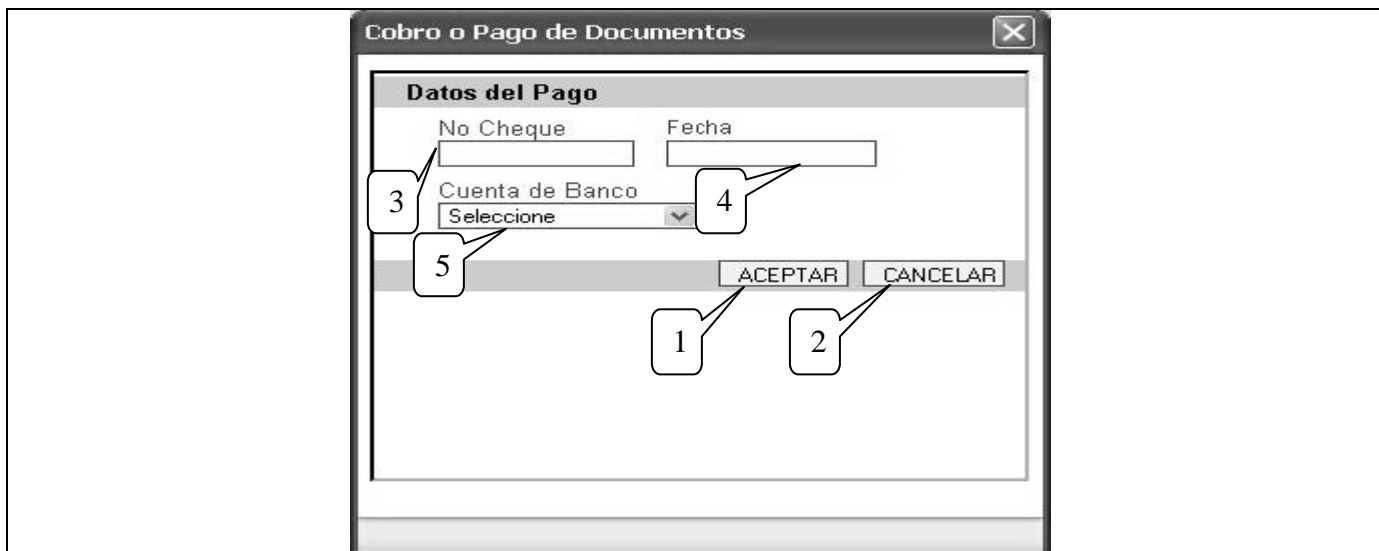
Nombre: Rg. Tipo: Text box.

7. Control para buscar el documento por la especialidad.

Nombre: Especialidad. Tipo: Text box.

8. Control para buscar el documento por la unidad militar.

<p>Nombre: Unidad M. Tipo: Text box.</p> <p>9. Control para buscar el documento por el cliente.</p> <p>Nombre: Cliente. Tipo: Text box.</p> <p>10. Control para que abre la ventana en la cual se cobra o se paga el documento, chequea que todos pertenezcan al mismo Rg, la misma Unidad Militar, Especialidad y Cliente.</p> <p>Nombre: Pagar Tipo: Botón.</p>	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona del menú principal Actualizar, luego Actualizar Pago/Cobro de Documentos	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas.
3. El contador selecciona el tipo de operación (3) que desea filtrar y presiona la opción Aceptar (1).	4. El sistema muestra el Grid (4) con los documentos que han sido contabilizadas y el botón Pagar (10) activo.
5. El contador selecciona los documentos que desea pagar o cobrar, deben tener todas el mismo Rg (6), la misma Especialidad (7), Unidad Militar (8) y Cliente (9), además se puede buscar por esos criterios. El contador presiona la opción Pagar (10).	6. El sistema muestra la Interfaz II con las opciones Aceptar (1) y Cancelar (2) activas.
Interfaz II.	



1. Control de aceptar una acción.

Nombre: Aceptar. Tipo: Botón.

2. Control de cancelar una acción.

Nombre: Cancelar. Tipo: Botón.

3. Control para introducir el número de cheque.

Nombre: Tipo de Operación. Tipo: Text box.

4. Control para seleccionar la fecha en la que se le asigna un cheque al documento.

Nombre: Fecha. Tipo: Text Box.

5. Control para seleccionar la cuenta de banco.

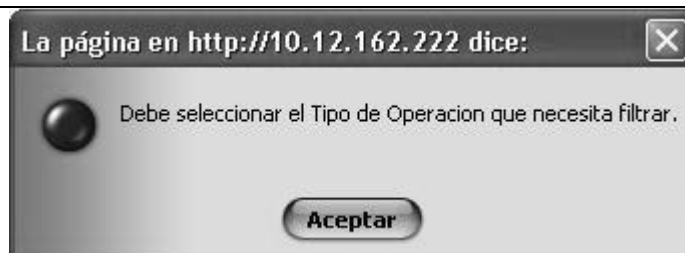
Nombre: cuenta de Banco. Tipo: Combo Box.

7. El contador introduce el número de cheque (3), selecciona la fecha (4) y la cuenta de banco (5). El contador presiona la opción Aceptar (1) y cierra la ventana.

8. El sistema pasa el o los documentos al estado de pagado o cobrado, actualiza el grid (4) y se termina el caso de uso.

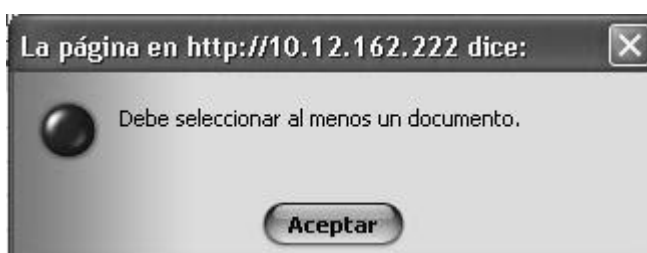
Cursos alternos

Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.

Acción 6: Si el contador no selecciona ningún documento en el grid el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 6: Si alguna factura difiere en los criterios de las demás, el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 8: Si el contador no introduce ningún dato, el sistema muestra el siguiente mensaje, el contador presiona Aceptar.

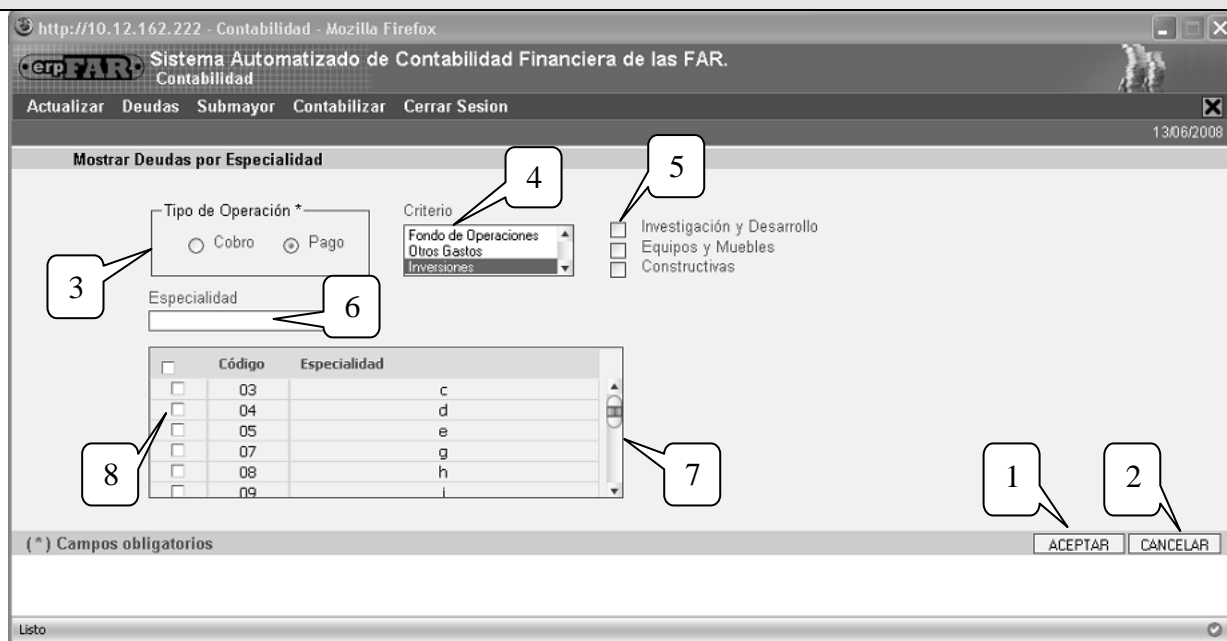


Requerimientos no Funcionales	
Postcondiciones	
Prioridad	Crítica

Caso de Uso	Mostrar Deudas por Especialidad
Actores	Contador
Propósito	Mostrar todas las deudas que tiene una o varias especialidades.
Resumen	El caso de uso comienza cuando el contador decide conocer todas las deudas que tiene una o varias especialidades de acuerdo a varios criterios. El caso de uso termina cuando el contador revisa los datos del reporte.
Responsabilidades	R5, R5.1, R5.2, R5.3, R5.4
CU asociados	
Precondiciones	
Requisitos especiales	

Descripción

Interfaz I.



1. Control de aceptar una acción.

Nombre: Aceptar. Tipo: Botón.

2. Control de cancelar una acción.

Nombre: Cancelar. Tipo: Botón.

3. Control para seleccionar el tipo de operación por la cual que se desea filtrar.

Nombre: Tipo de Operación. Tipo: Radiobutton.

<p>4. Control que aparece visible de acuerdo al tipo de operación que se selecciona. Nombre: Criterio. Tipo: Select Multiple.</p> <p>5. Control que aparece visible si se selecciona el criterio Inversiones. Nombre: Concepto. Tipo: Chek Box.</p> <p>6. Control para buscar la especialidad a partir de su código. Nombre: Especialidad. Tipo: Text box.</p> <p>7. Control para mostrar el código y la denominación de las especialidades. Nombre: Grid. Tipo: Grid.</p> <p>8. Control para marcar la o las especialidades, se pueden seleccionar todas además. Nombre: Chek box. Tipo: Chek box.</p>	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona del menú principal Deudas. Luego Mostrar Deudas por Especialidad.	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas.
3. El contador selecciona el tipo de operación que desea filtrar (3).	4. El sistema muestra los criterios de acuerdo al tipo de operación seleccionada (4).
5. El contador selecciona él o los criterios (4), selecciona la o las especialidades que desea filtrar (8) y presiona la opción Aceptar (1).	6. El sistema muestra la Interfaz II.
Interfaz II.	

Deudas por Especialidad

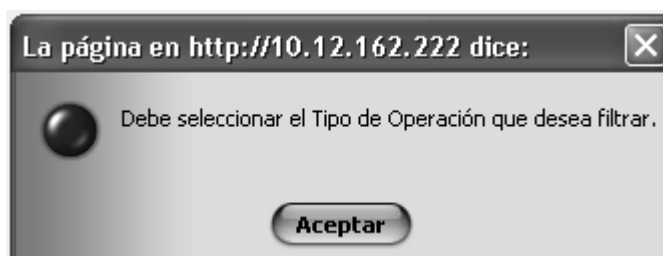
ESPECIALIDAD: acero					
RG	CUENTA	NO.FACTURA	CLIENTE	EDAD*DIAS	IMPORTE
1	265.2	1	1392	16	23.00
				Subtotal	23
				Total	23

ESPECIALIDAD: bzero					
RG	CUENTA	NO.FACTURA	CLIENTE	EDAD*DIAS	IMPORTE
2	265.1	2	1420	9	2.00
				Subtotal	2
				Total	2

7. El contador revisa los datos, cierra la ventana y se termina el caso de uso.

Cursos alternos

Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.

Acción 5: El Contador puede buscar la especialidad (6) por el código de la misma.

Acción 6: Si el Contador no selecciona ningún criterio el sistema asume que se escogieron todos.

Acción 6: Si el contador selecciona el criterio Inversiones el sistema muestra los conceptos asociados (5), si el contador no selecciona ningún concepto el sistema asume que se escogieron todos.

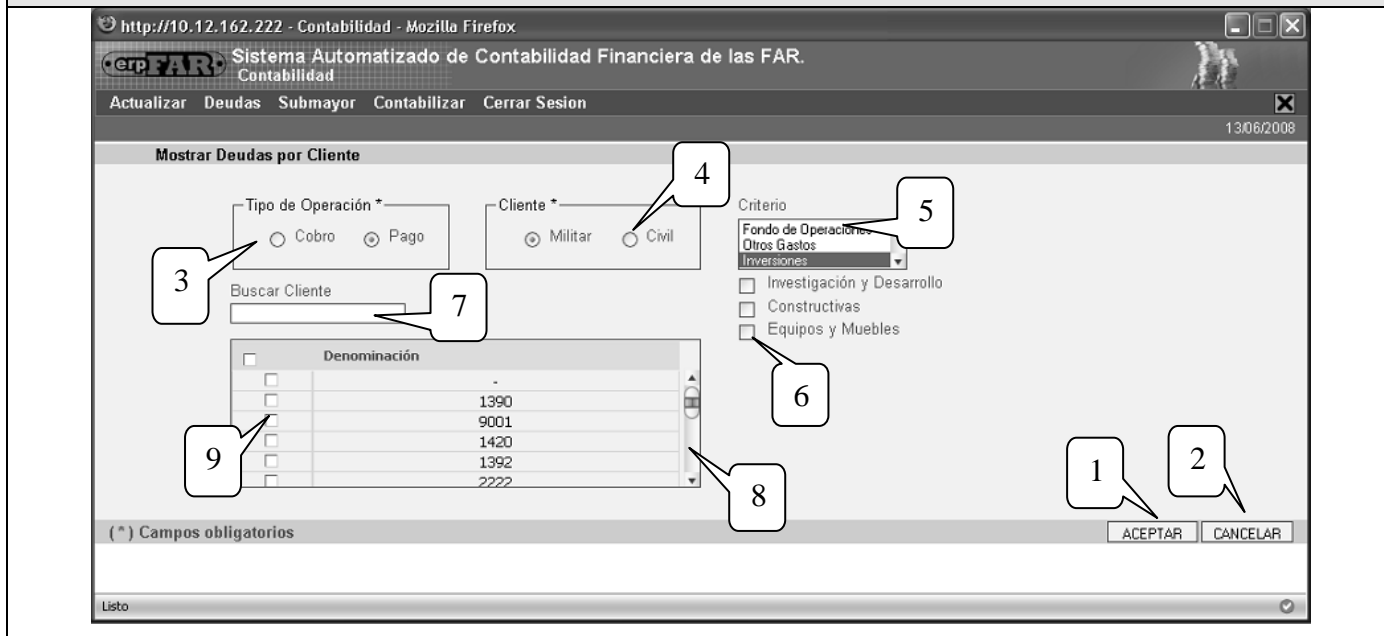
Acción 6: Si el Contador no selecciona ninguna especialidad el sistema asume que se escogieron

todas.	
Requerimientos no Funcionales	
Postcondiciones	
Prioridad	Secundaria

Caso de Uso	Mostrar Deudas por Cliente
Actores	Contador
Propósito	Mostrar todas las deudas que tiene uno o varios clientes
Resumen	El caso de uso comienza cuando el contador decide conocer todas las deudas que tiene uno o varios clientes de acuerdo a varios criterios. El caso de uso termina cuando el contador revisa los datos del reporte.
Responsabilidades	R4, R4.1, R4.2, R4.3, R4.4
CU asociados	
Precondiciones	
Requisitos especiales	

Descripción

Interfaz I.



<p>1. Control de aceptar una acción. Nombre: Aceptar. Tipo: Botón.</p> <p>2. Control de cancelar una acción. Nombre: Cancelar. Tipo: Botón.</p> <p>3. Control para seleccionar el tipo de operación por la cual que se desea filtrar. Nombre: Tipo de Operación. Tipo: Radiobutton.</p> <p>4. Control para seleccionar el tipo de cliente. Nombre: Tipo de Cliente Tipo: Radiobutton.</p> <p>5. Control que aparece visible de acuerdo al tipo de operación que se selecciona. Nombre: Criterio. Tipo: Select Multiple.</p> <p>6. Control que aparece visible si se selecciona el criterio Inversiones. Nombre: Concepto. Tipo: Chek Box.</p> <p>7. Control para buscar el cliente a partir de la denominación. Nombre: Cliente. Tipo: Text box.</p> <p>8. Control para mostrar la denominación de las especialidades. Nombre: Grid. Tipo: Grid.</p> <p>9. Control para marcar el o los clientes, se pueden seleccionar todos además. Nombre: Chek box. Tipo: Chek box.</p>	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El contador selecciona del menú principal Deudas. Luego Mostrar Deudas por Cliente.	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas.
3. El contador selecciona el tipo de operación que desea filtrar (3).	4. El sistema muestra los criterios de acuerdo al tipo de operación seleccionada (5).
5. El contador selecciona el tipo de cliente (4).	6. El sistema muestra los clientes de acuerdo al tipo seleccionado (8).
7. El contador selecciona él o los criterios (5), selecciona él o los clientes que desea filtrar (8) y presiona la opción Aceptar (1).	8. El sistema muestra la Interfaz II.
Interfaz II.	

Deudas por Cliente

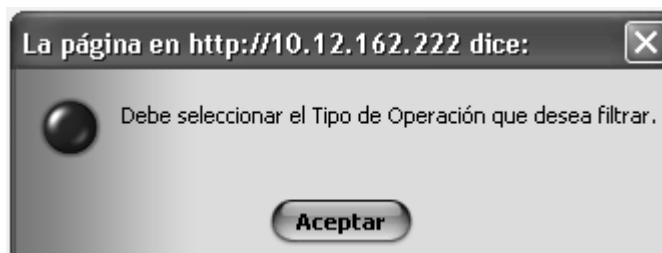
CLIENTE: 1392					
RG	CUENTA	NO.FACTURA	ESPECIALIDAD	EDAD*DIAS	IMPORTE
1	265.2	1	acero	16	23.00
				Subtotal	23
				TOTAL	23

CLIENTE: 1420					
RG	CUENTA	NO.FACTURA	ESPECIALIDAD	EDAD*DIAS	IMPORTE
2	265.1	2	bcero	9	2.00
				Subtotal	2
				TOTAL	2

9. El contador revisa los datos, cierra la ventana y se termina el caso de uso.

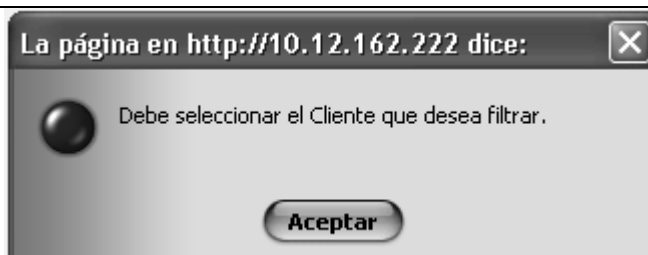
Cursos alternos

Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.

Acción 6: Si el Contador no selecciona el tipo de cliente (4) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 7: El Contador puede buscar el cliente (7) por la denominación del mismo.

Acción 8: Si el Contador no selecciona ningún criterio el sistema asume que se escogieron todos.

Acción 8: Si el contador selecciona el criterio Inversiones el sistema muestra los conceptos asociados (6), si el contador no selecciona ningún concepto el sistema asume que se escogieron todos.

Acción 8: Si el Contador no selecciona ningún cliente el sistema asume que se escogieron todos.

Requerimientos no Funcionales	
Postcondiciones	
Prioridad	Secundaria

Caso de Uso	Mostrar Submayor por Cliente
Actores	Contador
Propósito	Mostrar el Submayor de Cuentas por Pagar
Resumen	El contador inicia el caso de uso cuando decide conocer el Submayor de Cuentas por Pagar teniendo en cuenta varios criterios. El caso de uso termina cuando el contador revisa los datos del reporte.
Responsabilidades	R6, R6.1, R6.2, R6.3, R6.4
CU asociados	
Precondiciones	
Requisitos especiales	
Descripción	
Interfaz I.	

1. Control de aceptar una acción.

Nombre: Aceptar. Tipo: Botón.

2. Control de cancelar una acción.

Nombre: Cancelar. Tipo: Botón.

3. Control para seleccionar la fecha inicial por la cual se quiere emitir el reporte.

Nombre: Fecha Inicial. Tipo: Text Box.

4. Control para seleccionar la fecha final por la cual se quiere emitir el reporte.

Nombre: Fecha Final. Tipo: Text Box.

5. Control para seleccionar el tipo de operación por la cual que se desea filtrar.

Nombre: Tipo de Operación. Tipo: Radiobutton.

6. Control que aparece visible de acuerdo al tipo de operación que se selecciona.

Nombre: Criterio. Tipo: Select Multiple.

7. Control para mostrar los identificadores de las especialidades seleccionadas.

Nombre: IdEspecialidad. Tipo: Chek Box.

8. Control para seleccionar el tipo de cliente.

Nombre: Tipo de Cliente. Tipo: Radiobutton.

9. Control para seleccionar la especialidad.

Nombre: Especialidad. Tipo: Radiobutton.

10. Control para seleccionar el tipo de suma por la cual se quiere emitir el reporte.

Nombre: Suma Acumulada. Tipo: Chek Box.

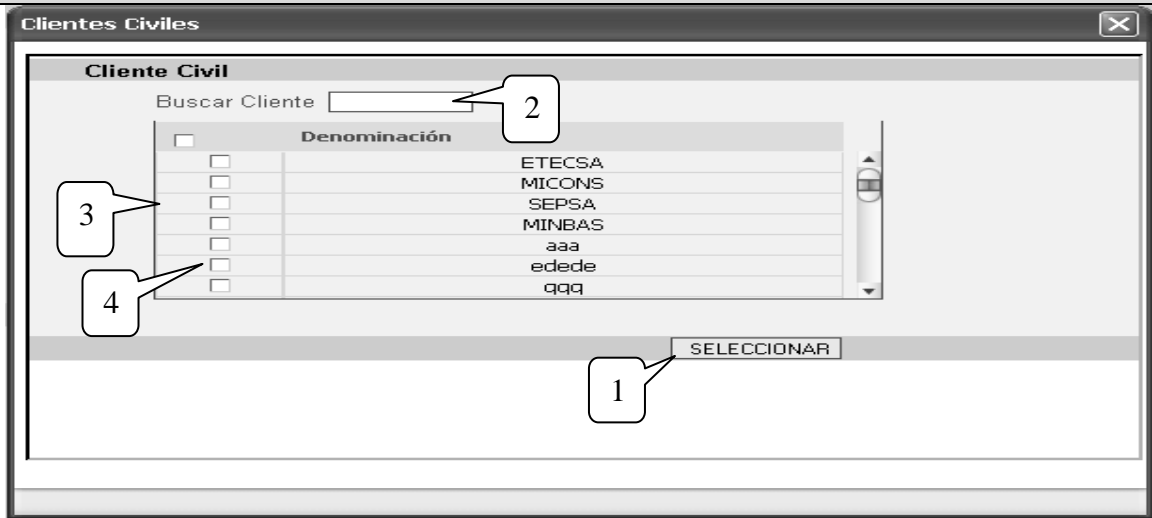
11. Control para seleccionar el tipo de suma por la cual se quiere emitir el reporte.

Nombre: Suma Desglosada. Tipo: Chek Box.
 12. Control para mostrar los identificadores de los clientes seleccionados.
 Nombre: IdCliente. Tipo: Chek Box.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El contador selecciona del menú principal Submayor. Luego Mostrar Submayor por Cliente.	2. El sistema muestra la Interfaz I con las opciones Aceptar (1) y Cancelar (2) activas.
3. El contador selecciona el rango de fecha, la fecha inicial (3) y la fecha final (4) por la cual quiere emitir el reporte.	4. El sistema muestra (3) y (4) respectivamente las fechas seleccionadas.
5. El contador selecciona el tipo de operación que desea filtrar (5).	6. El sistema muestra los criterios de acuerdo al tipo de operación seleccionada (6).
7. El contador selecciona los criterios (6). Selecciona si el cliente es Civil o Militar (8).	8. El sistema muestra la interfaz II si se seleccionó el tipo de cliente civil o interfaz III si se seleccionó el tipo de cliente militar, ambas con la opción seleccionar (1) activa.

Interfaz II.



1. Control para seleccionar el cliente civil.

Nombre: Seleccionar Tipo: Botón.

2. Control para buscar el cliente civil a partir de la denominación del mismo.

Nombre: Cliente Civil. Tipo: Text Box.

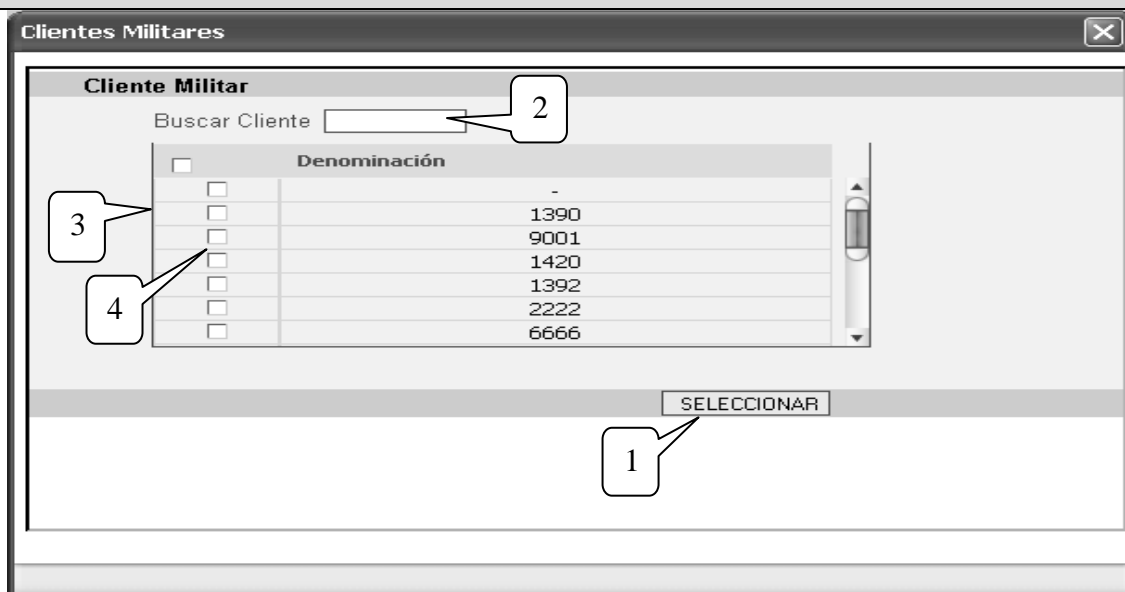
3. Control para mostrar la denominación de los clientes civiles.

Nombre: Grid. Tipo: Grid.

4. Control para marcar el o los clientes civiles, se pueden seleccionar todos además.

Nombre: Chek box. Tipo: Chek box.

Interfaz III.



1. Control para seleccionar el cliente militar.

Nombre: Seleccionar Tipo: Botón.

2. Control para buscar el cliente militar a partir de la denominación del mismo.

Nombre: Cliente Militar. Tipo: Text Box.

3. Control para mostrar la denominación de los clientes militares.

Nombre: Grid. Tipo: Grid.

4. Control para marcar el o los clientes militares, se pueden seleccionar todos además.

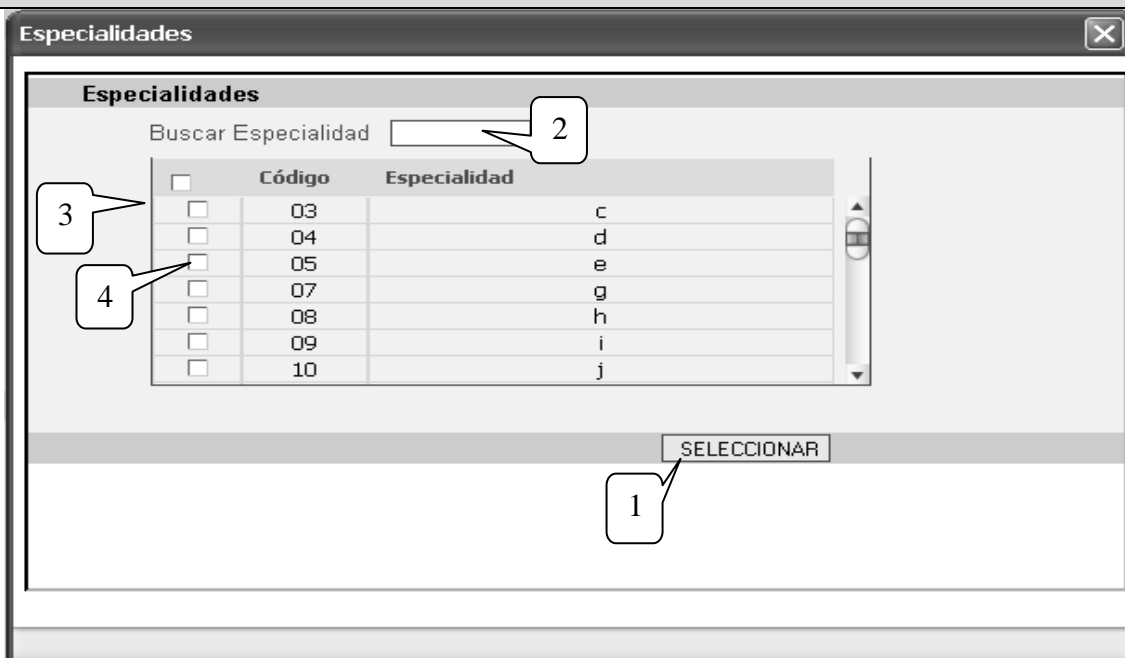
Nombre: Chek box. Tipo: Chek box.

9. El contador selecciona el o los clientes y luego presiona la opción Seleccionar (1).

10. El sistema muestra el identificador del o los clientes seleccionados en el campo (12).

11. El contador selecciona la Especialidad (9). 12. El sistema muestra la interfaz IV con la opción Seleccionar (1) activa.

Interfaz IV.



1. Control para seleccionar la especialidad.

Nombre: Seleccionar Tipo: Botón.

2. Control para buscar la especialidad a partir de su código.

Nombre: Cliente Militar. Tipo: Text Box.

3. Control para mostrar el código y la denominación de las especialidades.

Nombre: Grid. Tipo: Grid.

4. Control para marcar la o las especialidades, se pueden seleccionar todas además.

Nombre: Chek box. Tipo: Chek box.

13. El contador selecciona la o las especialidades y luego presiona la opción Seleccionar (1). 14. El sistema muestra el identificador de la o las especialidades seleccionadas en el campo (13).

15. El contador selecciona si desea el submayor por el criterio de Suma Desglosada (11) o Suma Acumulada (10) y presiona 16. El sistema muestra la interfaz V.

la opción Aceptar (1).

Interfaz V.

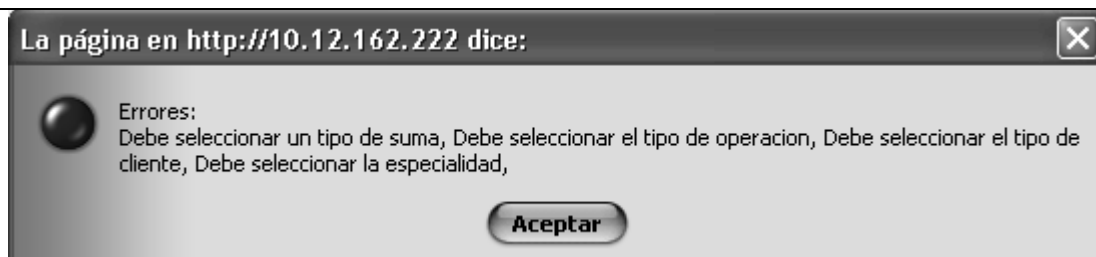
SUBMAYOR DE CUENTAS POR PAGAR Y/O COBRAR							
CLIENTE: MICONS							
FECHA DESDE: 04/12/2007 HASTA: 14/06/2008							
CRITERIO: Cuentas por Pagar - Otros Gastos							
RG	NO. CHEQUE	FACTURA	CUENTA	ESPECIALIDAD	DEBE	HABER	SALDO
SUMA DESGLOSADA							
7	13	7	410	d	0	2.33	2.33
7	13	10	410	d	0	2.00	4.33
TOTAL					0	4.33	4.33

SUBMAYOR DE CUENTAS POR PAGAR Y/O COBRAR							
CLIENTE: hjypppp							
FECHA DESDE: 04/12/2007 HASTA: 14/06/2008							
CRITERIO: Submayor Gasto Otros Gastos							
RG	NO. CHEQUE	FACTURA	CUENTA	ESPECIALIDAD	DEBE	HABER	SALDO
SUMA DESGLOSADA							
1	0	665566	871	e	66.00	0	66
1	0	582	871	e	44.00	0	110
TOTAL					110	0	110

17. El contador revisa los datos, cierra la ventana y se termina el caso de uso.

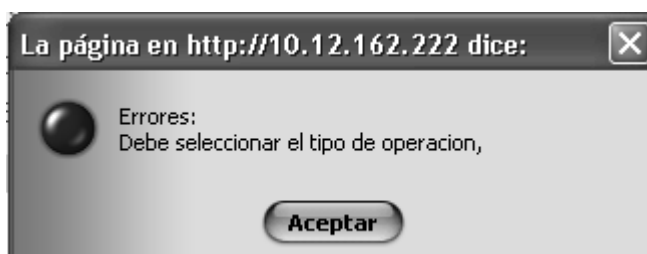
Cursos alternos

Acción 2: Si el contador selecciona la opción Aceptar (1) el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 2: Si el contador selecciona la opción Cancelar (2) el sistema no realiza ninguna acción y se termina el caso de uso.

Acción 6: Si el contador no selecciona el tipo de operación el sistema muestra el siguiente mensaje. El contador presiona Aceptar.

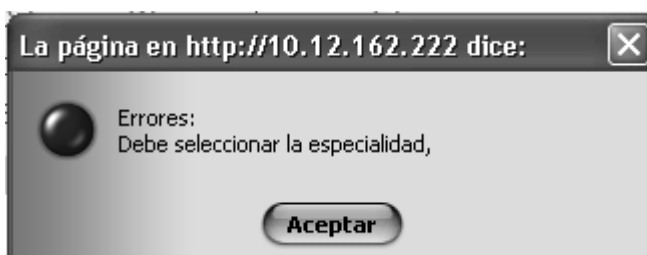


Acción 8: Si el contador no selecciona ningún criterio el sistema asume que se escogieron todos.

Acción 8: Si el Contador no selecciona el tipo de cliente el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 12: Si el Contador no selecciona la especialidad el sistema muestra el siguiente mensaje. El contador presiona Aceptar.



Acción 16: Si el Contador no selecciona alguna suma el sistema muestra el siguiente mensaje. El contador presiona Aceptar.

Requerimientos Funcionales	no
Postcondiciones	
Prioridad	Secundaria

Anexo 3. Arquitectura de tres capas

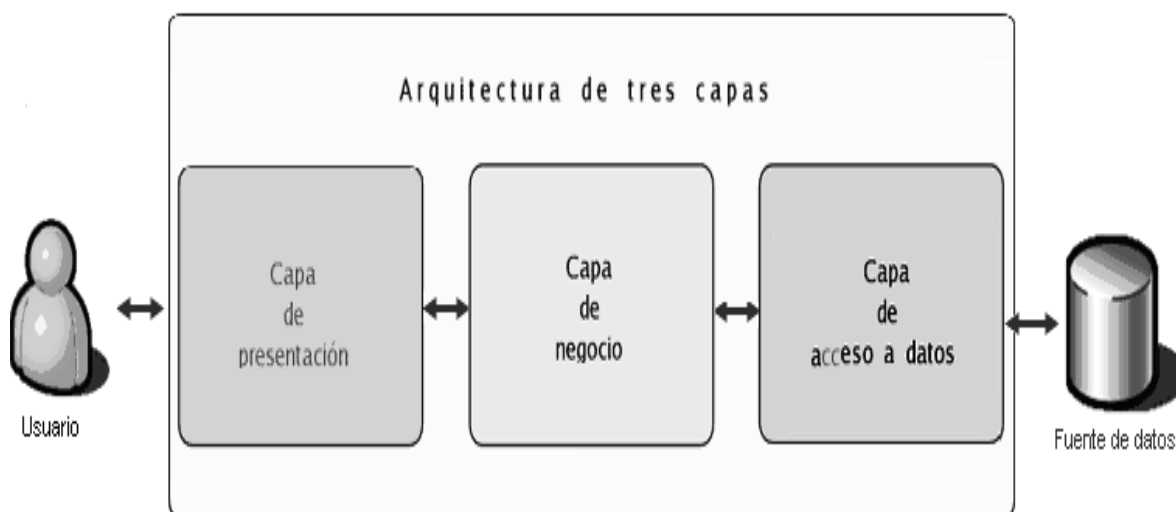


Fig.32 Arquitectura de tres capas

Anexo 4. Diagramas de secuencia

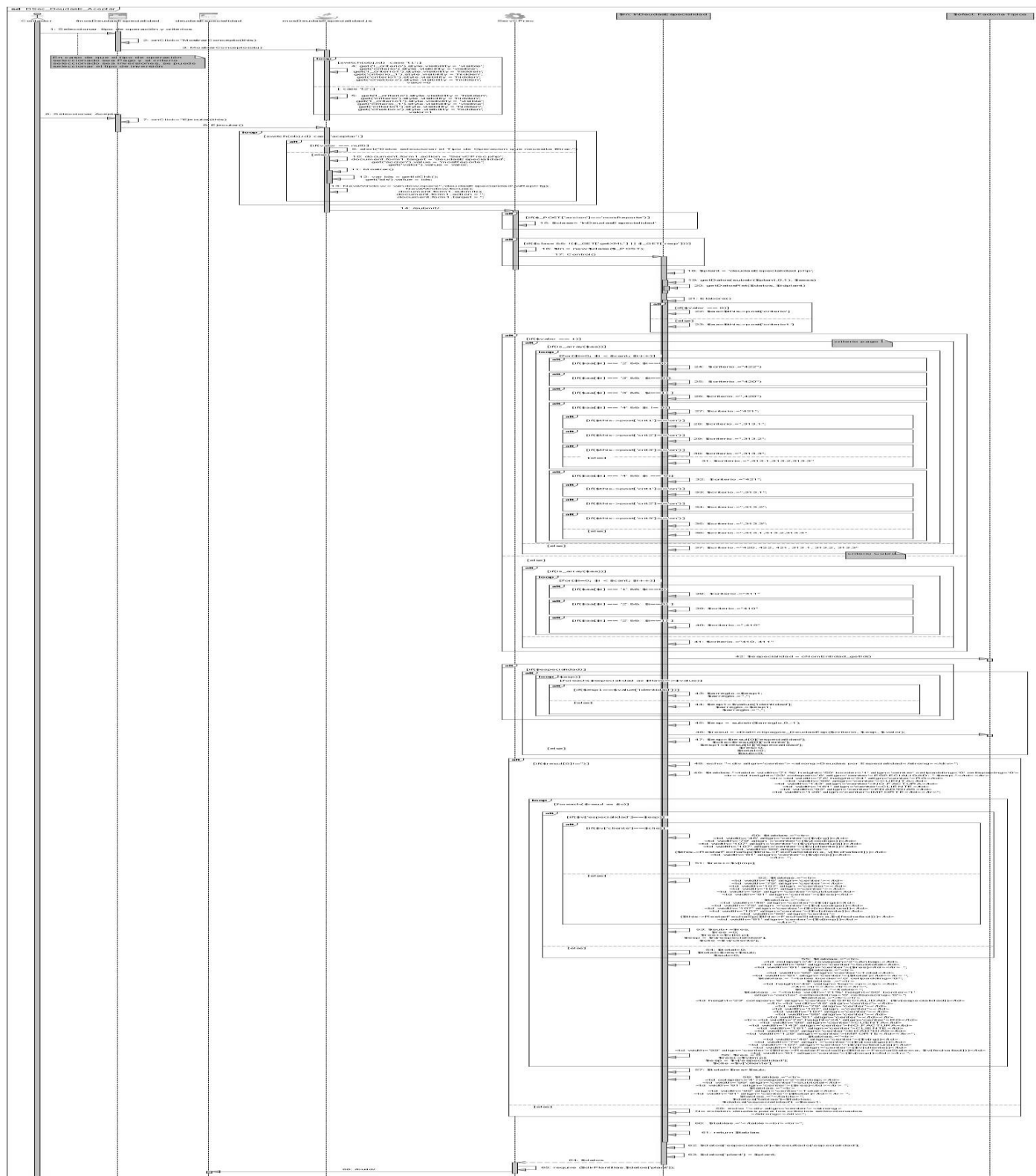


Fig.33 Diagrama de secuencia. CU Mostrar Deudas por Especialidad, Escenario Aceptar.

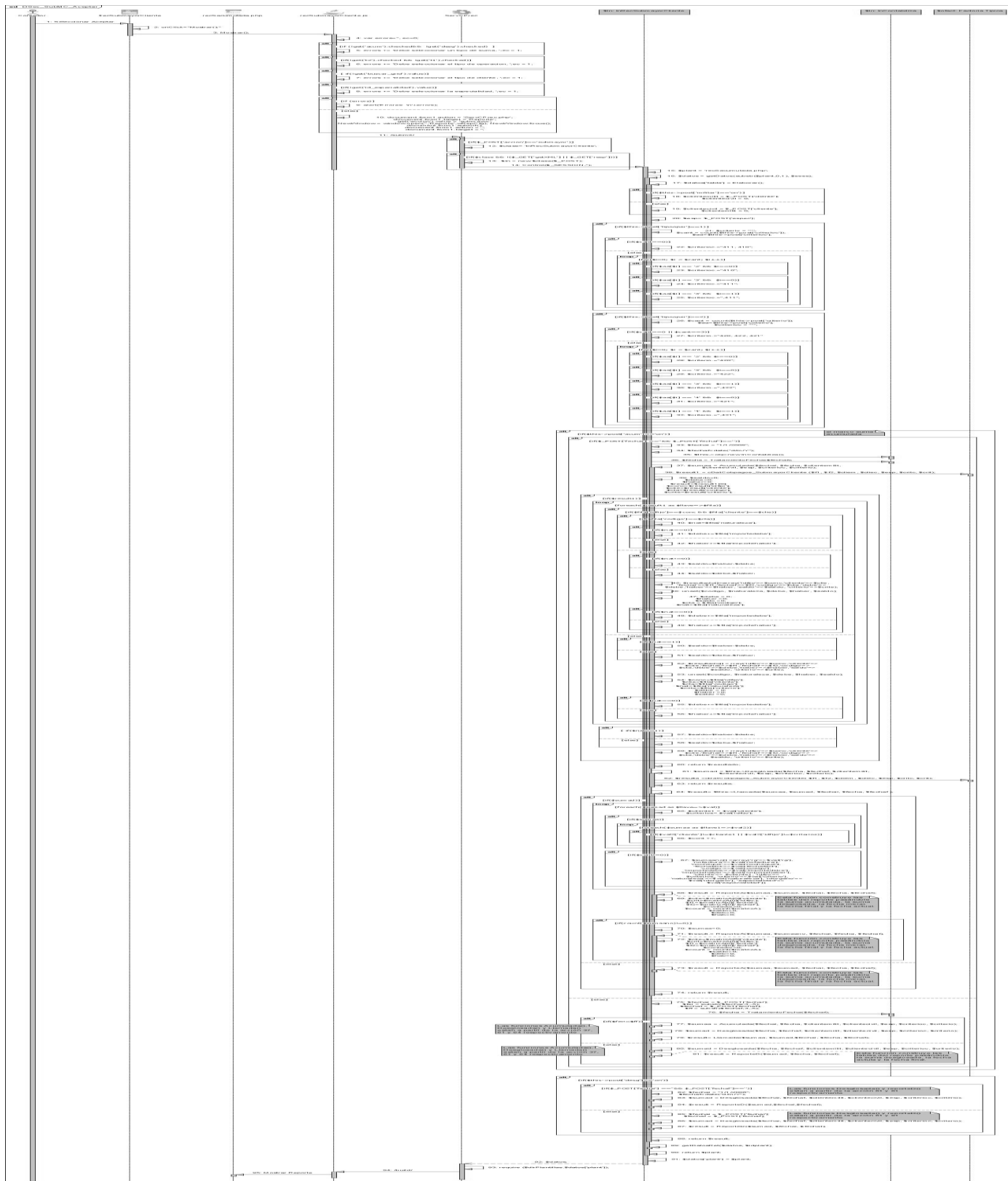


Fig.34 Diagrama de secuencia. CU Mostrar Submayor de Cuentas, Escenario Mostrar ventana emergente de lo Clientes y de las Especialidades.

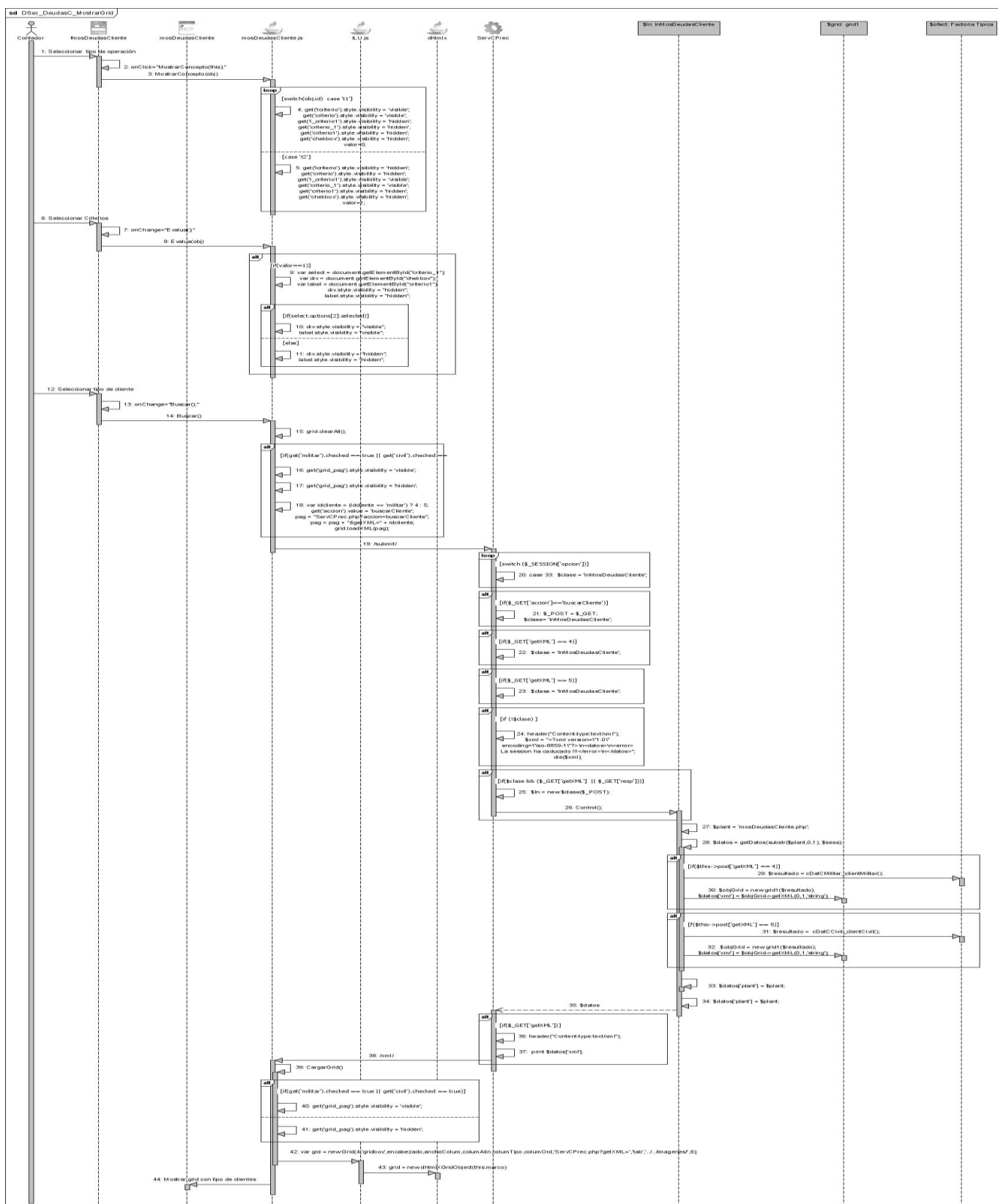


Fig.36 Diagrama de secuencia. CU Mostrar deudas por Cliente. Escenario Mostrar grid.

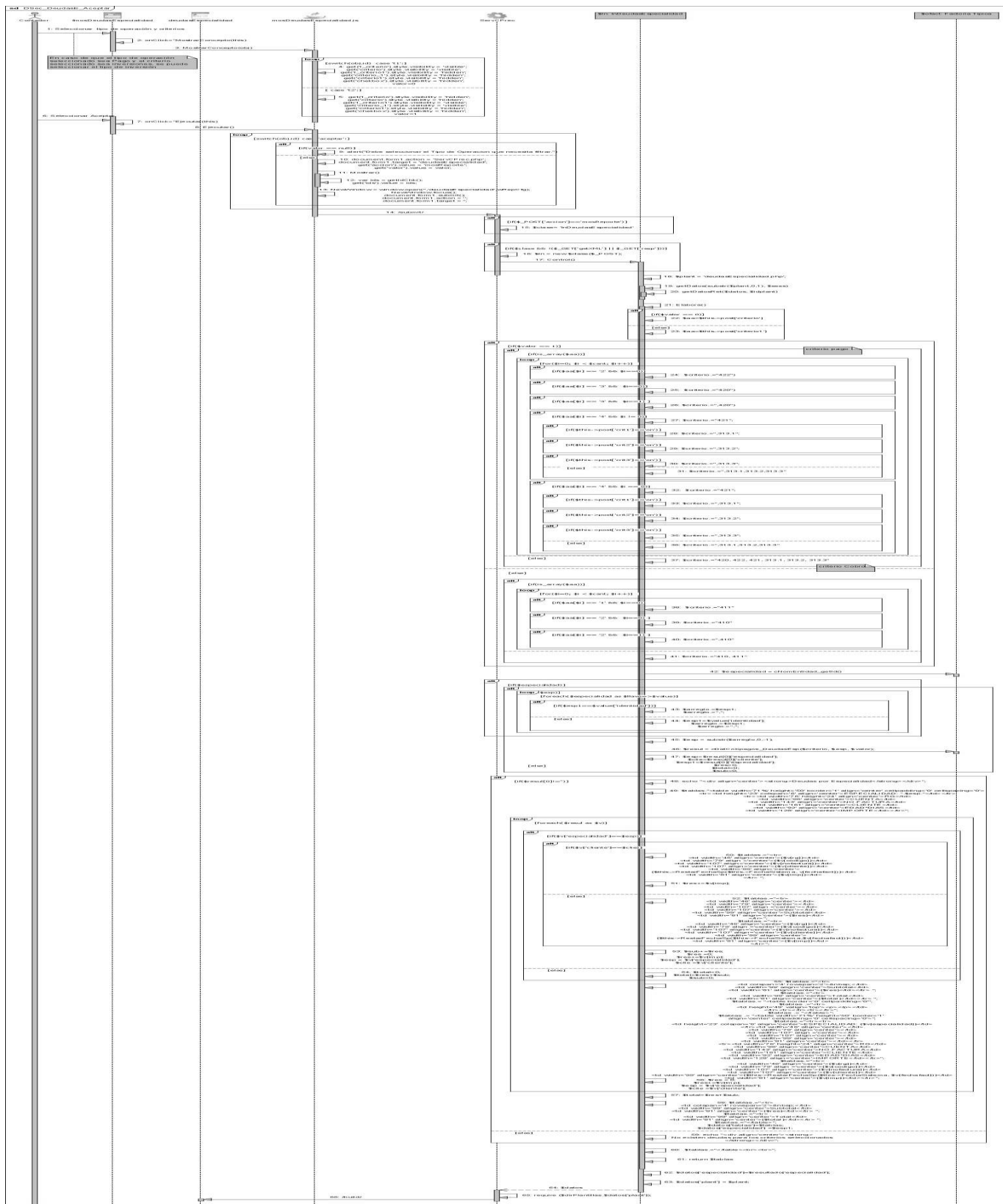


Fig.37 Diagrama de secuencia. CU Mostrar deudas por Cliente. Escenario Aceptar.

Anexo 5. Diagramas de Componentes

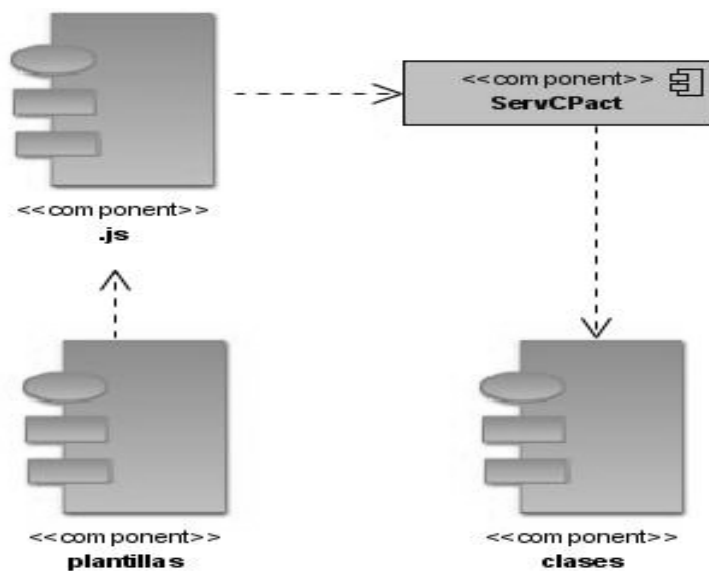


Fig.38 Paquete Actualizaciones

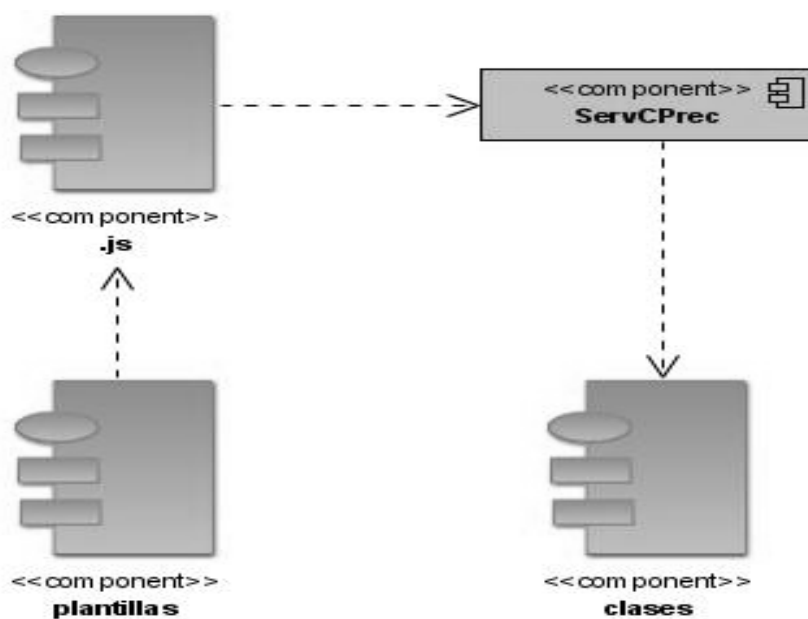


Fig.39 Paquete Recuperaciones