



Universidad de las Ciencias Informáticas
Facultad 9

MODELADO DE UNA APLICACIÓN WEB PARA EL MÓDULO INVENTARIO DEL SISTEMA CONDOR



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

**Autores: Indira Lilled Laurencio Fuentes.
Oscar Naranjo González.**

**Tutores: Msc. María Isabel Martínez García.
Tec. Rosa de Lima González Ruiz.**

**Ciudad de La Habana, Mayo de 2007
“Año 49 de la Revolución”**

A nuestros padres...

AGRADECIMIENTOS

La amistad y el cariño que se recibe de los que te rodean son factores que determinan de forma muy decisiva el desarrollo y equilibrio de una persona. La cualidad más hermosa que debe revestir un ser humano es reconocer a quienes lo han llevado a conseguir sus objetivos.

No es tarea fácil poder escribir en un papel a todos los que nos ayudaron en la realización de este trabajo o que compartieron momentos inolvidables de nuestras vidas, pero trataremos de agradecerles a todos aquellos que nos aportaron su granito de arena o que con su amor y cariño nos dieron fuerzas para poder realizar este trabajo.

Agradecemos a la Revolución y a nuestro Comandante en Jefe, Fidel Castro, por hacernos parte de este, su gran sueño y darnos esta oportunidad única que pocos jóvenes en el mundo tienen.

A la UCI, por habernos formado como profesionales a la altura de nuestros tiempos.

A todos nuestros profesores, por transmitirnos los conocimientos y valores que nos permitieron formarnos como profesionales. En particular a los profesores Mario González Arencibia, Yuniel Proenza, Febe Ciudad y Jorge Osorio por sus atenciones y por tendernos la mano en momentos de urgencia.

A nuestra tutora Rosita, por ser atenta, preocupada, cariñosa, por brindarnos todo su apoyo, por atendernos en todo momento, por poner a nuestra disposición todos sus conocimientos, sin usted no sería posible el desarrollo de este trabajo. Gracias Rosy, le debemos mucho y le queremos mucho más.

A los compañeros del SICS que nos atendieron muy bien, en especial a Papo, Perfecto, Alejandro, Blanca Mayda y Marieta.

Este es el comienzo de un nuevo camino en nuestras vidas, y los conceptos que hasta ahora nos eran difíciles de concebir comenzarán a hacerse realidad, pero nada de esto sería posible sin la ayuda de

ustedes. Esperamos humildemente que lo que se expresa en este papel les llegue a todos y toque de muy cerca su corazón.

A todos ustedes MUCHAS GRACIAS.

De Indira:

Para mi es muy difícil agradecerles a todas las personas que me han apoyado y ayudado a llegar hasta aquí o que simplemente con su amor y cariño me han impulsado a luchar para lograr cosas grandes como esta. Soy muy afortunada al contar con muchísima gente linda que me quiere y que llevo muy cerquita de mi corazón. Quiero destacar que no por dejar de mencionar a muchas de esas personas que contribuyeron de una forma u otra con mi desarrollo, son menos importantes.

A mis padres por ser lo más valioso que tengo, el más grande de mis tesoros, por ser mis guías, mi orgullo, mi ejemplo y más que eso, por ser parte de la vida misma. Gracias por existir y ser como son. A los dos, muchas gracias.

- Gracias mamita por ser como eres, buena, sencilla, franca, exigente, enorme de corazón. Gracias por ser dedicación, comprensión, cariño, ternura y pasión. Sin tu amor y preocupación no hubiese llegado a ser lo que soy. No le puedo pedir más a la vida porque tú eres lo mejor que me ha dado la vida, porque tú eres mi vida misma.

- Gracias papi por ser así como eres, dulce, bueno, modesto, humilde y carismático. Gracias por seguirme a donde voy, por estar ahí cuando te necesito, por quererme y preocuparte por mí. Gracias por depositar toda tu confianza en mí. Gracias sencillamente por ser el mejor padre del mundo.

Se cuanto les enorgullece verme graduada y cuanto desearon este momento, pero eso este triunfo es de los tres, porque no somos tres, somos uno.

A mi Guchy, porque fuiste mi apoyo y mi fuerza en estos cinco años, por brindarme la posibilidad de conocer el amor y por hacerme tan feliz en el tiempo que compartimos juntos. Por ayudarme y apoyarme

en todas mis cosas, por estar siempre que te necesitaba, por quererme tanto. Nunca te voy a olvidar Osmany. Te quiero mucho.

A mi mami Sara por todo el apoyo y la ayuda que me has brindado, por tu preocupación constante, por cuidarme y protegerme como una hija más. Gracias, muchas gracias mami, te llevo muy dentro.

A mi tía Nitza por todo lo que me quieres, por estar pendiente de mí, de mis cosas, por aceptarme como soy. No cambies nunca, eres maravillosa así.

A las dos muchas gracias por su apoyo incondicional, además gracias por son mis segundas mamas. Por si nunca se los dije, hoy les digo "Las quiero mucho".

A mis abuelos: Peme, Juan Manuel y Virgen, los quiero con todo mi corazón, y a mi abuelo Juan que ya no esta con nosotros pero de estarlo se sentiría feliz de mi triunfo.

A mis hermanos Yuny, Yunier, Eulis y Eulier, porque son los hermanos que no tuve.

A mi tío Onel porque a tu forma me enseñaste a esforzarme y estudiar desde chiquitica y me demostraste que estudiando se llega lejos y a mi tía Maribel por su ayuda y preocupación.

A Mode porque a pesar de que no llevas mi sangre te quiero como tal y un poco más y porque sé que me quieres y me consideras como una hija. Gracias por apoyarme y quererme tanto. Te quiero.

A Cachy por su simpatía y carisma. Gracias Churry por hacerme reír siempre y ser mi pareja de dominó. Gracias mi tío. Te quiero.

A mis tíos y primos.

A mis compañeros y amigos:

- A mis grandes amigos de la secundaria: Yudenia, Yudilén y Carlos porque a pesar de los años que han pasado no me olvidan. A mis amigos del IPVCE, entre ellos: Mariurka, Yeril, Magelis, Niurka, Dayaris, Mairin, Maireen, Yudenia, Yanelys y muchísimos otros gracias a todos por no olvidarme y tenerme siempre presente.

- A Gilber mi gran amigo de la universidad y quien siempre esta cuando lo necesito.
- A Tony que a pesar de estar lejos siempre puedo contar con su preocupación y cariño.

A Eniel porque en tres años me quisiste grande, me ayudaste y apoyaste en todo y me enseñaste muchas cosas importantes de la vida, entre ellas a luchar por lo que quiero.

A todos mis vecinos por su preocupación y apoyo en especial a Puyi, Morell, Darlenis.

A mis enemigos por obligarme a ser cada día mejor.

A todos, muchísimas gracias, ustedes forman parte de este logro.

De Oscar:

Para muchos universitarios al llegar a su último año la satisfacción más grande que pueda tener es sentir el roce del título en sus manos, estar completamente convencidos que tantos años dedicados a los estudios no fueron en vano y ser o sentirse el inmenso y gran orgullo de sus padres.

A mis padres, especialmente a ellos quiero agradecerles infinitamente. A ellos porque semana tras semana me supieron dar todo el apoyo que necesitaba, mucho ánimo en los estudios y por hacerme sentir seguro y con ansias de llegar a la meta final durante estos cinco años de la carrera.

Muchas gracias a ti padre por darme el ejemplo que me hizo crecer en la vida.

Y a ti madre gracias también, quiero que sepas que admiro mucho tu valioso sacrificio, un sacrificio que ha convertido ya nuestra familia en mucho más que lo soñado.

A mi hermana Aniley también le doy parte de mis agradecimientos por ser tan atenta y preocupada conmigo en todo momento.

A mi queridísima abuela Claudina mis más sinceros agradecimientos por estar siempre al tanto de mí y apoyarme en los estudios diciéndome en todo momento: “Estudia que el futuro está en tus manos y el sacrificio vale mucho”.

Le doy parte de mis agradecimientos a la profesora Yaqueline Zamora Mora por haberme enseñado a trabajar con el manejador de referencias “EndNote” para organizar toda mi referencia bibliográfica. Gracias también al profesor Yuniel Proenza quien fue de lo más atento, ayudándonos a evacuar cualquier duda surgida en la parte de Ingeniería de Software.

A Indira, mi compañera de tesis quiero agradecer mucho, a ella porque con su gran entusiasmo hemos logrado ser un buen equipo así como labrar esta experiencia inolvidable.

DECLARACIÓN DE AUTORÍA

Por este medio declaramos que somos los únicos autores de este trabajo y autorizamos a la empresa de Sistemas Informáticos, Consultaría y Sistemas (SICS) y a la Universidad de las Ciencias Informáticas (UCI) para que hagan el uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año 2007.

Indira Lilled Laurencio Fuentes

Oscar Naranjo González

Msc. Maria Isabel Martínez García

Tec. Rosa de Lima González Ruiz.

OPINIÓN DEL TUTOR

Tutor: Tec. Rosa de Lima González Ruiz.

Desde las primeras entrevistas y entrega de documentación acerca del trabajo en que se enfrentarían, ambos estudiantes mostraron un interés y disciplina dignos de elogio. Se realizaron diversos encuentros para la revisión de cada una de las etapas y aclaración de dudas para una mayor comprensión del problema. Siempre fueron puntuales con la hora acordada y presentaban los diferentes temas con una planificación tal que resultaban muy productivas las reuniones, lo que hacía que avanzara el trabajo con calidad.

Respecto al conocimiento de las herramientas con las que han desarrollado la tesis, tengo el criterio que dominan muy bien las metodologías de las diferentes tecnologías de punta propuesta para implementar el módulo “Control de Inventarios CONDOR” que formará parte del proyecto contable del mismo nombre. La presentación de la Tesis también tiene muy buena calidad dado que recoge en su totalidad todos los eventos en cuanto al control de los inventarios una vez llegados al almacén, con toda la documentación requerida por el Ministerio de Finanzas y Precios para las Entradas / Salidas de los productos..

En cuanto a la valoración del trabajo en general pienso que aparte de todos los méritos por la envergadura económica que tiene, también posee algo de gran importancia como es la generalización de la aplicación a cualquier entidad del país que controle la actividad de almacenes.

Debo resaltar que hicieron algunos aportes con ideas novedosas como el tratamiento individual por almacén. Esta nueva concepción brinda mayor flexibilidad al cliente en el momento de la explotación ya que realizará los cierres contables diarios al almacén que tenga todos sus documentos actualizados, no teniendo que esperar por todos los almacenes pues en ocasiones alguno está lejos de la entidad que los controla por medios computarizados. También han sido receptivos a sugerencias que se le han hecho en algunos aspectos. Mi evaluación personal es que ha sido excelente el desarrollo del trabajo de ambos estudiantes.

Rosa de L. González Ruiz

RESUMEN

La investigación se realiza en el SICS en uno de sus principales productos, el ConDor. Debido a las desventajas señaladas por los clientes surge la idea de modelar y diseñar una aplicación Web, en este caso, para uno de sus módulos: "Control de Inventarios", que reemplace esta aplicación de escritorio existente, para complacer las exigencias y sugerencias de los clientes del SICS y usuarios del sistema, además cuenta con mejoras y agregaciones de funcionalidades contables y en un entorno sencillo y amigable, fácil de instalar y utilizar. Se quiere la modelación de una aplicación que brinde la seguridad e integridad que requieren los datos a procesar y que agilice y facilite el proceso de control de inventario.

En este trabajo se presenta el análisis y diseño del negocio y el sistema propuesto para dar una solución factible al problema existente utilizando como metodología de construcción de software, El Proceso Unificado del Rational (RUP) y se hace un estudio de los sistemas existentes y de las herramientas adecuadas para la construcción del sistema, además se realiza el estudio de la factibilidad de la propuesta para la estimación de de costos y esfuerzo de la aplicación.

Palabras claves

- ConDor
- Inventario
- Aplicación Web
- Modelado

ABSTRACT

The investigation is carried out in the SICS in one of its main products, the Condor. Due to the disadvantages pointed out by the clients the idea it arises of modeling and to design a Web application, in this case, for one of his modules: "Control of Inventories", that replaces this existing desktop application, to satisfy the demands and suggestions of SICS' clients and users. In addition, this Web makes possible improvements and additional countable functionalities within a simple and friendly environment, easy to install and to use. Its goal is the modelling of an application that offers security and integrity when processing data as well as to fasten the process of inventory control.

The analysis and design of business as well as the proposed system are presented as a solution to the stated problem. For this reason, a methodology of software construction, the Rational Unified Process (RUP) is used. Also, a study of the existing systems, its suitable tools, and the feasibility of the proposal for costs estimation of the application is made.

Key words

- ConDor
- Inventory
- Web Application
- Modelling

Índice

| | |
|---|----|
| Introducción | 2 |
| Capítulo 1 Fundamentación Teórica..... | 7 |
| 1.1 Introducción | 7 |
| 1.2 Conceptos asociados al dominio del problema..... | 7 |
| 1.3 Objeto de Estudio | 8 |
| 1.3.1 Descripción General Módulo Inventario del Sistema de Gestión Contable “ConDor”..... | 9 |
| 1.3.1.1 Fase de Apertura | 9 |
| 1.3.1.2 Fase de Explotación..... | 10 |
| 1.3.2 Descripción Actual del Dominio del Problema | 15 |
| 1.3.2.1 Actualidad del sistema | 16 |
| 1.3.3 Fundamentación de la Situación Problémica..... | 17 |
| 1.4 Análisis de otras Soluciones Existentes..... | 18 |
| 1.4.1 GIB –Gestión de Inventario de Bienes..... | 19 |
| 1.4.2 Versat - Sarasola | 19 |
| 1.4.3 SIP- Sistema de Inventario del Patrimonio Cultural y Nacional. | 20 |
| 1.5 Conclusiones parciales | 21 |
| Capítulo 2 Tendencias y Tecnologías actuales utilizadas | 23 |
| 2.1 Introducción | 23 |
| 2.2 Aplicaciones Web..... | 23 |
| 2.2.1 Web dinámicas..... | 25 |
| 2.3 Arquitectura a implementar | 26 |
| 2.3.1 Modelo Cliente/Servidor..... | 27 |
| 2.3.1.1 Programación en lado del Cliente. Funciones | 28 |
| 2.3.1.2 Programación en lado del Servidor. Funciones | 28 |
| 2.3.2 Programación Multicapas..... | 29 |
| 2.3.3 Plataforma .NET..... | 30 |
| 2.3.3.1 ASP. NET | 32 |
| 2.4 Metodología | 33 |

| | | |
|------------|---|----|
| 2.4.1 | Fundamentación de la Metodología utilizada. Proceso Unificado de Desarrollo | 34 |
| 2.5 | Lenguaje para el Modelado de Objetos | 36 |
| 2.5.1 | Lenguaje Unificado de Desarrollo | 36 |
| 2.6 | Herramientas a utilizar | 38 |
| 2.6.1 | Herramienta de Modelado..... | 38 |
| 2.6.1.1 | Rational Rose Enterprise Edition | 38 |
| 2.6.2 | Herramientas de desarrollo..... | 39 |
| 2.6.2.1 | Microsoft Visual Studio 2005..... | 40 |
| 2.6.2.2 | SQL Server 2005 Express | 40 |
| 2.7 | Lenguajes de Programación | 42 |
| 2.7.1 | C#..... | 42 |
| 2.7.2 | SQL (Structure Query Language) | 43 |
| 2.8 | Conclusiones parciales | 44 |
| Capítulo 3 | Características del sistema. | 51 |
| 3.1 | Introducción | 51 |
| 3.2 | Modelo de Negocio | 51 |
| 3.2.1 | Actores y Trabajadores del Negocio | 52 |
| 3.2.2 | Proceso de Negocio..... | 53 |
| 3.2.3 | Diagrama de Caso de Uso del Negocio..... | 54 |
| 3.2.4 | Descripción Textual de los Casos de Uso del Negocio | 56 |
| 3.2.6 | Modelo de Objetos | 60 |
| 3.3 | Especificación de los Requisitos del Software..... | 61 |
| 3.3.1 | Requerimientos Funcionales del Sistema..... | 61 |
| 3.3.2 | Requerimientos no Funcionales del Sistema..... | 65 |
| 3.4 | Definición de los Casos de Uso | 68 |
| 3.4.1 | Diagrama de Caso de Uso del sistema..... | 69 |
| 3.4.2 | Expansión de los Casos de Uso | 70 |
| 3.5 | Conclusiones parciales | 72 |
| Capítulo 4 | Construcción de la solución propuesta | 75 |
| 4.1 | Introducción | 75 |

| | | |
|------------|--|----|
| 4.2 | Análisis..... | 75 |
| 4.2.1 | Diagrama de Clases de Análisis | 76 |
| 4.3 | Principios de Diseño | 76 |
| 4.3.1 | Estándares de la Interfaz de la Aplicación | 76 |
| 4.3.2 | Concepción General de la Ayuda | 77 |
| 4.3.3 | Tratamiento de Excepciones..... | 77 |
| 4.4 | Patrones..... | 78 |
| 4.4.1 | Patrones de arquitectura..... | 78 |
| 4.4.2 | Patrones de diseño | 79 |
| 4.5 | Diseño | 79 |
| 4.5.1 | Diagrama de Clases del Diseño..... | 80 |
| 4.5.2 | Diagramas de Iteración (Secuencia)..... | 80 |
| 4.5.3 | Diseño de la Base de Datos..... | 81 |
| 4.5.4 | Modelo de Despliegue | 82 |
| 4.6 | Generalidades de la implementación..... | 82 |
| 4.7 | Conclusiones parciales | 82 |
| Capítulo 5 | Estudio de Factibilidad | 85 |
| 5.1 | Introducción | 85 |
| 5.2 | Planificación y Estimación..... | 85 |
| 5.2.1 | Cálculo de Puntos de Casos de Uso sin Ajustar (UUCP) | 86 |
| 5.2.1.1 | Calcular el Factor de Peso de los Actores sin Ajustar (FPA) | 86 |
| 5.2.1.2 | Calcular el Factor de Peso de los Casos de Uso sin Ajustar (FPCU)..... | 87 |
| 5.2.2 | Cálculo de los Puntos de Casos de Usos Ajustados (PCUA) | 89 |
| 5.2.2.1 | Calcular el Factor de Complejidad Técnica | 89 |
| 5.2.2.2 | Calcular el Factor de Complejidad Ambiente (FA) | 91 |
| 5.2.2.3 | Calcular los Puntos de Casos de Uso Ajustados | 91 |
| 5.3 | Cálculo del Esfuerzo de Desarrollo (E) | 92 |
| 5.3.1 | Convertir los Puntos de Casos de Uso Ajustados a Esfuerzo de Desarrollo..... | 92 |
| 5.3.2 | Cálculo de Tiempo | 93 |
| 5.3.3 | Cálculo del Costo Total a partir del Esfuerzo en Horas - Hombres | 93 |

| | |
|--|-----|
| 5.4 Beneficios Tangibles e Intangibles | 95 |
| 5.4.1 Beneficios Tangibles | 95 |
| 5.4.2 Beneficios Intangibles | 95 |
| 5.5 Análisis de Costos y Beneficios | 96 |
| 5.6 Conclusiones parciales | 97 |
| Conclusiones | 98 |
| Recomendaciones | 100 |
| Referencias bibliográficas | 101 |
| Bibliografía | 104 |
| Glosario de términos | 105 |
| Anexos | 108 |



INTRODUCCIÓN

“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”.

Albert Einstein.

“La ciencia a pesar de sus progresos increíbles, no puede ni podrá nunca explicarlo todo. Cada vez ganará nuevas zonas a lo que hoy parece inexplicable. Pero las rayas fronterizas del saber, por muy lejos que se eleven, tendrán siempre delante un infinito mundo de misterio”.

Gregorio Marañón

INTRODUCCIÓN

Al analizar los principales impactos y desafíos que las nuevas Tecnologías de la Información y las Comunicaciones (TIC) plantean, obtenemos un escenario donde el creciente desarrollo de la Informática y las Telecomunicaciones emergen como el factor principal que impulsa la globalización a pasos agigantados, no solo porque permite realizar actividades y negocios¹ más rápidos y de manera más eficiente, sino también, porque abre las puertas para realizar nuevos negocios, y permite eliminar o reducir las barreras del costo², el tiempo y los errores.

La Informatización de la Sociedad, en el camino de la formación de la Sociedad de la Información en Cuba, es el proceso de utilización ordenada y masiva de las Tecnologías de la Información y las Comunicaciones en la formación del conocimiento necesario para la vida cotidiana de los ciudadanos y las instituciones, en todas las esferas de la sociedad y en todos los procesos, para lograr cada vez más eficacia y eficiencia que de como resultado el incremento constante de la calidad de vida de los ciudadanos y la generación de riquezas que la haga sostenible.

Como parte de las estrategias cubanas de Informatización se creó en el año 2000 el Ministerio de la Informática y las Comunicaciones (MIC), con la misión fundamental de impulsar, facilitar, ordenar y regular el uso masivo de servicios y productos de las Tecnologías de la Información, las Comunicaciones, la Electrónica y la Automatización para satisfacer las expectativas de todas las esferas de la sociedad.

El MIC regula, orienta y controla a las entidades que se dedican a la realización y actualización de tecnologías y medios informáticos entre las que se encuentra el SICS, empresa que brinda Servicios Informáticos, Consultoría³ y Sistemas al Ministerio de Transporte (MITRANS) y a otras entidades. Entre los servicios que brinda se encuentra el desarrollo y comercialización de aplicaciones, arquitectura de Base de Datos (BD) y lenguajes de programación, consultoría de Tecnologías de la Información,

¹ Actividad encaminada a obtener un beneficio.

² Coste, precio, importe, valor, gasto, tarifa.

³ Prestación de servicios de consejería sobre asuntos técnicos.

comercialización de equipamiento, partes, piezas de repuesto e insumo, servicios técnicos, entre otros servicios.

Aunque su campo de prestación de servicios es bastante amplio, su especialidad es la venta de software. Su principal producto en estos momentos es el ConDor, el cual es un sistema contable que está generalizado en más de mil entidades del territorio nacional siendo los organismos más representativos el MITRANS y el Instituto Nacional de Recursos Hidráulicos (INRH).

El Sistema Contable ConDor cuenta con una suite gigante integrada por siete módulos: Contabilidad General, Activos Fijos, Inventario, Nómina/prenómina, Disponibilidad Financiera, Condexce y Efectos. Actualmente este software se encuentra implementado para el Sistema Operativo Windows en cualquiera de sus versiones

Este trabajo se centra únicamente en el Módulo Inventario de este enorme sistema el cual constituye el **objeto de estudio**, siendo así el **campo de acción** todo el entorno de desarrollo de aplicaciones Web.

La versión del módulo existente presenta la característica que está implementado en un entorno monousuario, lo que constituye la **situación problemática** a la cual va enfocado la investigación, ya que ha sido señalado por parte de los clientes como una significativa desventaja. También este sistema presenta la característica que cada módulo tiene una Base de Datos diferente, lo que trae redundancia⁴ en la información, aspecto este que también se quiere mejorar.

Como consecuencia de esto nos planteamos como **problema científico**: la inexistencia de una aplicación Web que facilite el trabajo con el sistema, particularmente en el módulo Inventario, además con una Base de Datos única, centralizada para todos los módulos.

Para resolver este problema se ha planteado como **objetivo**:

- Modelar los artefactos de ingeniería para una aplicación Web en el Módulo Inventario del proyecto “ConDor”.

⁴ Repetición, reiteración, insistencia, demasía, exceso, superfluidad

El Módulo Inventario está compuesto por tres procesos: “Control de Inventario”, “Facturación” y “Útiles y Herramientas en Uso”. Debido a la complejidad del sistema y el tiempo de desarrollo con que se cuenta, se ha calculado el alcance de desarrollo del sistema, por lo que en este trabajo se modela solamente la parte de Control de Inventario del módulo, quedando para próximos análisis los demás procesos.

Para dar cumplimiento a este objetivo es necesario realizar las siguientes **tareas**:

- Estudio sobre el funcionamiento del ConDor y en particular del módulo Inventario.
- Selección de la plataforma y herramientas que se utilizaran para el diseño de la aplicación Web.
- Análisis del modelo de negocio, captura de los Requisitos y confección de sus Casos de Uso⁵ del Sistema.
- Diseño de la aplicación y de la BD.
- Estudio de la factibilidad del sistema.

Si se logra modelar una aplicación Web con una interfaz cómoda en el módulo Inventario, los usuarios interactuarían de forma más fácil, rápida y más eficiente, por lo tanto nuestra **idea a defender** es realizar la modelación de una aplicación Web que garantice gran eficiencia en los procesos de inventario en cualquier institución donde sea instalada, ganando tiempo y minimizando errores.

Para el desarrollo de este trabajo se tuvo en cuenta los siguientes **métodos**:

- Entrevistas a expertos, principalmente a aquellas personas que participaron en la realización de las primeras versiones del producto, como programadores, asesores, etc.
- Entrevistas a clientes y usuarios del sistema.
- Búsquedas en Internet y en bibliografías relacionadas con el tema, además de estudios de documentos escritos por los programadores del sistema en las anteriores versiones.

Con todo este estudio se espera como resultado obtener una aplicación Web que perfeccione el módulo inventario ya implementado en Windows, pero además de eso que guste a los clientes y les facilite su uso

⁵ Un caso de uso es una secuencia de transacciones que son desarrolladas por un sistema en respuesta a un evento que inicia un actor sobre el propio sistema.

y control resolviendo así el problema planteado. Además de constar con un sistema con un ambiente multiusuario y una Base de Datos centralizada para todos sus procesos.

Este trabajo esta conformado por cinco capítulos.

Capítulo 1: Se expone el estado del arte, donde se realiza la fundamentación teórica del tema. Al mismo tiempo se describe el objeto de estudio, se explica el funcionamiento del módulo, los procesos fundamentales y otros detalles considerables. También se mencionan y describen algunos sistemas existentes concernientes con inventario.

Capítulo 2: Se justifican las tendencias y tecnologías utilizadas para el modelado de la aplicación y otras necesarias para su futura implementación.

Capítulo 3: Describe el negocio y se hace el análisis del sistema a desarrollar. Se definen las funcionalidades del sistema y se describen detalladamente, utilizando el Rational Rose como herramienta de modelación.

Capítulo 4: Se enfoca en la construcción de la solución mediante diagramas de clases de análisis y diseño, modelo de datos, diagrama de despliegue y se plantean los principios para el diseño, conceptualización general de la ayuda, tratamiento de errores y estándares de la implementación. Aquí se construyen las funcionalidades que se definieron en el capítulo anterior.

Capítulo 5: Se realiza el estudio de factibilidad sobre el sistema, obteniendo los beneficios tangibles e intangibles y analizando los costos del desarrollo de esta propuesta.



FUNDAMENTACIÓN TEÓRICA

“La idea que las buenas técnicas restringen la creatividad es como decir que un artista puede pintar sin aprender nada sobre formas, o decir que un músico no necesita saber nada sobre teoría musical”.

Marvin Zelkowitz.

Capítulo 1

Fundamentación Teórica

1.1 Introducción

En este primer capítulo se hace una explicación de manera general abordando los principales puntos relacionados con el objeto de estudio, dando lugar a su desarrollo en dos momentos significativos. Como tema a desarrollar se plantea el Módulo Inventario del sistema ConDor y se hacen breves comparaciones con algunos software actualmente utilizados.

Hoy muchas empresas se han desarrollado grandemente en cuanto a su sector económico, esto no hubiera sido posible sin la gran ayuda que le ha concebido la informática, pues de esta ciencia se puede hablar en cualquier rama por su excelente compatibilidad ya que es capaz de realizar las actividades de una manera automatizada sustituyendo en muchos casos el esfuerzo humano.

1.2 Conceptos asociados al dominio del problema

A continuación se exponen conceptos importantes para la comprensión del dominio del problema.

Inventario

Inventarios son bienes tangibles que se tienen para la venta en el curso ordinario del negocio o para ser consumidos en la producción de bienes o servicios para su posterior comercialización. Los inventarios comprenden, además de las materias primas, productos en proceso y productos terminados o mercancías para la venta, los materiales, repuestos y accesorios para ser consumidos en la producción de bienes fabricados para la venta o en la prestación de servicios; empaques y envases y los inventarios en tránsito.

La base de toda empresa comercial es la compra y venta de bienes o servicios; de aquí la importancia del manejo del inventario por parte de la misma. Este manejo contable permitirá a la empresa mantener el control oportunamente, así como también conocer al final del período contable un estado confiable de la situación económica de la empresa. Ahora bien, el inventario constituye las partidas del activo corriente que están listas para la venta, es decir, toda aquella mercancía que posee una empresa en el almacén valorada al costo de adquisición, para la venta o actividades productivas.

Tarjetero de Productos

Formulario que permite registrar los datos que corresponden a todos los productos que se encuentran en los almacenes para su adecuada identificación y control, tanto físico como contable.

Nomencladores

Constituyen los conceptos claves, son las entradas a las tablas de la base de datos de las diferentes operaciones, registros y áreas con las que se cuentan. Estas entradas son importantes e imprescindibles para la gestión del sistema, sin las cuales no se puede realizar el proceso. Se pueden clasificar en específicos que son las únicas para el módulo⁴ Inventario y *comunes* que son compartidas con otros módulos del sistema.

1.3 Objeto de Estudio

El objeto de estudio de esta investigación está enmarcado en el módulo Inventario del Sistema de Gestión Contable ConDor, el cual ya se encuentra automatizado pero se quiere perfeccionar y actualizar con los nuevos avances tecnológicos. Para lograrlo se hace necesaria una modelación exitosa del módulo ya mencionado. Para esto se realiza un estudio detallado del módulo en sentido general con el fin de comprender su funcionamiento, y además algo de muy buena utilidad resulta ser el proyecto anterior implementado en Windows sirviendo este de apoyo para realizar partes de la modelación del nuevo sistema.

⁴ Pieza o conjunto unitario de piezas que se repiten en una construcción de cualquier tipo, para hacerla más fácil, regular y económica.

1.3.1 Descripción General Módulo Inventario del Sistema de Gestión Contable “ConDor”

El sistema de Control de Inventarios está concebido para el control de las entradas o salidas de uno o varios almacenes y además para la Contabilización de los movimientos que se realizan con los productos almacenados. Este módulo consta dos momentos importantes: Fase de Apertura, y Fase de Explotación.

1.3.1.1 Fase de Apertura

Ante todo se aclara brevemente el significado que contablemente tiene, la palabra Apertura. Esto quiere decir que hay que dar la información de las existencias de cada producto que tiene el almacén o almacenes de la entidad que utilice esta aplicación⁵, desde el momento que se va a implantar y que debe coincidir específicamente con el último día del mes del último Balance General. Esto es debido a que la Contabilidad informa cuál es el saldo que tiene al cierre contable de cada mes, en cada cuenta que representa a los diferentes tipos de productos, llamada Cuenta de Inventario y codificada⁶ con un número, para partir de esos saldos como datos con los que se iniciará la aplicación una vez llena la base informativa general. Cuando el tarjetero esté completo, los saldos deben coincidir con los de Contabilidad General (RUIZ 2005).

En esta primera fase se trabaja con Nomencladores y Tarjetero General de Productos.

Nomencladores

- Almacenes.
- Tipos de Movimientos.
- Órdenes de Trabajo.
- Opciones Contables.
- Cuentas de Inventario.

⁵ Programa preparado para una utilización específica.

⁶ Transformar mediante las reglas de un código la formulación de un mensaje.

Tarjetero General de Productos

Aquí se aclara el significado de la sigla CUP (Código Universal de Productos) referida a un código⁷ único establecido en el país para codificar de forma única cada producto. Dicho código consta de 15 dígitos. En este módulo se toma el CUP como llave y se adiciona el código Interno. En el tarjetero están registrados todos los productos que se encuentran en existencia en el almacén, de ahí la importancia del mismo.

- **Reportes:**

Listados del Tarjetero por diferentes Criterios, Resúmenes de Cuentas por almacenes, Por Orden de Introducción, etc., con el objetivo de ayudar en la revisión de los saldos para dar como buena la información tecleada. Estos reportes son de mucha importancia en esta fase.

- **Cierre de Apertura:**

- Valida la existencia de productos en el Tarjetero y de artículos en los nomencladores de Centros de Costo, Almacenes, Áreas, Clasificador de Cuentas, etc... También valida que exista una fecha de apertura lo que significa que se declaran las existencias y valores que cada producto tenía al final del mes del último balance.
- Imprime un reporte con los saldos de las Cuentas/Subcuentas por cada almacén que debe coincidir con los saldos emitidos en los reportes.

1.3.1.2 Fase de Explotación

Ante todo se explica brevemente el significado contable del concepto Explotación. Quiere decir que estamos en condiciones de comenzar a dar entrada en la Base de Datos (BD), a cada uno de los modelos previstos por el Ministerio de Finanzas y Precios que autorizan a Recepar⁸ y dar Salida a los productos en un almacén. A todo este trasiego de productos, autorizado por las distintas instancias, es a lo que

⁷ Conjunto de signos formados por una serie de líneas y números asociados a ellas, que se pone sobre los productos de consumo y que se utiliza para la gestión informática de las existencias.

⁸ Recibir, acoger.

llamaremos movimientos. También a partir de esta fase es que se emiten los comprobantes contables que en cada caso, y en dependencia del concepto de Entrada/Salida, se deben entregar a Contabilidad General en la tabla de comprobantes; también se realizan los Inventario Físicos con la ayuda automática, se puede solicitar información a partir del Tarjetero actualizado diariamente y muchos otros procesos que se describirán en este informe.

Una vez hecho el Cierre de Apertura, el Menú Principal habilita el resto de las opciones previstas para este módulo, aunque el que salgan habilitadas o no depende de los accesos que tenga autorizado el usuario que esté operando. Esto quiere decir que no todos los usuarios tienen acceso a todas las opciones. El módulo contempla esta posibilidad, muy útil para depurar responsabilidad en las operaciones que queden registradas (RUIZ 2005).

En esta segunda fase se trabaja con diferentes procesos donde el **proceso 1** lo llamaremos el proceso principal. Estos procesos se encuentran numerados del 1 al 10 pero no significa que sea un orden a seguir, solo es cuestión de categorización a la hora de identificarlos.

- **Proceso 1: MODELOS A PROCESAR**

- **Informes de Recepción:**

Este es un modelo estrictamente de Entrada de materiales al almacén. Primeramente es realizado por el almacenero al amparo de una factura. Irá escribiendo cada uno de los productos con sus códigos correspondientes, descripción, unidad de medida y contando las cantidades que dice la factura a fin de conciliar con la mercancía para plasmarla en el modelo. Este pasa posteriormente al departamento de Contabilidad a fin de que se complete por el especialista en inventario el resto de los datos y darle entrada en el sistema.

- **Vales de Entrada / Salida:**

Este modelo sirve tanto como entrada o salida de productos. En el primer caso está amparado por una Solicitud de Materiales que será, tanto para la producción, como alimentos para el comedor, materiales de

oficina, etc... A partir de esa solicitud llena el modelo detallando entre otros: el Centro de Costo, Orden de Trabajo, cada producto por código y nombre, las cantidades solicitadas, UM, y las existencias que quedan. A su vez, cuando uno o varios productos son devueltos al almacén, el almacenero llena el mismo modelo pero especificando en una casilla que es una devolución y llena de igual forma el resto del modelo.

- Transferencias de Materiales:

Este modelo se llena cuando se va a transferir mercancía hacia una dependencia de la empresa o se recibe mercancía desde una dependencia. Un ejemplo claro sería: una empresa importadora, que tiene almacenes en todas las provincias, cuando recibe la mercancía de un suministrador, debe distribuirla a todo el país para la venta o insumo⁹. Esta distribución la realiza con el modelo de Transferencia y marca en el escaque al efecto, que es de salida.

A su vez para la dependencia que recibe, debe introducirlo en el sistema como una Transferencia de Entrada. Por tanto decimos que el mismo modelo sirve para recepcionar como dar salida. A todas estas acciones se le denomina contablemente Operaciones entre Dependencias y genera un comprobante que irá a la Contabilidad con sus cuentas correspondientes.

- Control de la Producción:

Este modelo se utiliza sólo para recepcionar en el almacén, productos que vienen de la producción. Puede ser en un taller, laboratorios, etc. como fabricación o reparación de piezas, fabricación de envases, de productos líquidos, grasos etc... Se llena por los jefes de producción que detallan cada producto con sus especificidades así como la(s) orden(es) que participaron en el proceso. Estos modelos llevan la sigla CIP (Control Interno de la Producción).

⁹ Conjunto de bienes empleados en la producción de otros bienes.

- Ajustes:

Este modelo es tanto para sumar como restar existencias, ya que como su nombre lo indica en ocasiones se altera la existencia de un producto por Rotura, Merma dentro de la norma o en exceso de la norma, por faltante o sobrante en el conteo físico, etc. y hay que hacer ajustes en la base de datos. El modelo oficial contempla 15 tipos diferentes de ajustes. Estos están contemplados en el nomenclador de Ajustes, y contabilizan en muchos casos de forma diferente. Por ejemplo el comprobante de un ajuste por merma no es igual que un ajuste por faltante en conteo físico; en ambos aparece menos existencia en los controles del almacenero, pero hay que llevarlo a la contabilidad en cuentas diferentes.

• Proceso 2: TRANSFERENCIA INTERNA DE PRODUCTOS ENTRE ALMACENES

Esto ocurre cuando se controlan varios almacenes en la misma carpeta porque son de la misma entidad, pero productos que se receptaron en un almacén se desea transferirlos para otro. Esta transferencia no genera comprobante, pero habrá que dejar traza de la operación tanto escrita como en tablas para solicitarla cuando se desee.

• Proceso 3. AJUSTE AL PRECIO DE UNO O VARIOS PRODUCTOS.

El objetivo principal de este formulario es cambiar el precio a un solo producto o a varios a la vez. Calcular la diferencia entre el precio viejo y el nuevo para formar un comprobante y además hacer el proceso correspondiente para que las diferencias entre los saldos de las cuentas que se afecten intervengan también en el Cierre del Día a fin de actualizar los saldos de dicha tabla.

• Proceso 4. CAMBIO DE CUENTAS DE UNO O VARIOS PRODUCTOS.

El objetivo principal de este programa es poder realizar un cambio de Cuenta de Inventario al producto que seleccione el usuario, teniendo en cuenta en el Cierre del día, los valores de los saldos en el caso que cambie la cuenta como tal o sea, los tres primeros dígitos de la cuenta. Si lo que cambia es el análisis o análisis-detalle no se tocará el Cuadre diario. Ahora bien, en ambos casos sí se generará un comprobante a los efectos de contabilidad. También habrá que tener en cuenta ese cambio y hacerlo efectivo en todas las tablas donde halla el campo CUP.

- **Proceso 5. INVENTARIO FÍSICO.**

Debe realizar las siguientes funciones

- Selección del un % determinado, por Cuentas, por Ubicación (Estante o Casilla), por rangos de CUP o Aleatorio.
- En el módulo nuevo se implementan el % variable y el método aleatorio a fin de seleccionar puntualmente el grupo de productos que desee tanto el usuario del sistema como un auditor.
- Debe imprimir listados o la Triple Tarjeta para realizar el conteo físico.
- Fijar Fecha de existencias en e inventario seleccionado.
- Dar entrada al conteo y brindar un listado de las diferencias si hay.
- Impresión de la Hoja de Inventario Oficial para plasmar el resultado del chequeo físico.
- Impresión de un Reporte resumen de los chequeos realizados en el año.
- Listado de los productos que no se hayan chequeado.

- **Proceso 6. CAMBIO DE CÓDIGO A UN PRODUCTO.**

En la práctica se ha detectado que los clientes necesitan esta opción debido a que utilizan códigos internos para identificar los productos y se dieron cuenta que algún código no está correcto. Quizás hasta ya se han realizado movimientos con él. Debe garantizar cambiar el código que tiene por otro que teclee el usuario y entonces en todas las tablas donde aparezca el campo CUP se cambiará por el nuevo. Este proceso no tiene implicaciones contables pero sí debe quedar muy correcto todo internamente

- **Proceso 7: CANCELACIÓN DE CUALQUIER MOVIMIENTO DENTRO DEL MES**

Esta opción revierte totalmente cualquier movimiento que se realice, pero siempre dentro del mes en curso. No se permite cancelar ningún movimiento de un mes anterior al del proceso.

- **Proceso 8: CIERRE DEL DÍA**

- Selecciona los movimientos listos para cerrar e imprime un listado con los que tienen problemas de conciliación de existencia.
- Calcula: Saldo Inicial + Entradas / Salidas y lo pone en el campo Inicio del día. Limpia la información de Entradas/Salidas del último día cerrado a fin de volcar la nueva información, todo esto en ambas monedas.

- Va calculando a partir de los modelos que tiene en proceso los valores en Entradas o Salidas y adicionándolo en las cuentas que correspondan para cada almacén. También hace la operación en el acumulado del mes, pero en este caso sin limpiar Entradas ni Salidas.

- **Proceso 9: CIERRE DEL MES**

- Valida que se haya hecho el Cierre del Día.
- Valida que no haya movimientos pendientes de Ajuste: (Existencia < > Almacén).
- Borra la información referente a todos los comprobantes del mes.
- Hace los cálculos necesarios para prepararla con la fecha del nuevo mes y los saldos iniciales tanto del día como acumulado del mes en ambas monedas.
- Borra los campos de Entradas y Salidas tanto en el Acumulado del Día como el Acumulado del mes.

- **Proceso 10: CIERRE DEL AÑO**

- Valida que se haya realizado el Cierre del Mes.
- Proporciona Listados de los productos que no se hayan chequeado y de los que no hayan tenido movimientos en el año.
- Posibilita borrar del tarjetero todos los productos cuya existencia = 0.00
- Pasa el Tarjetero y los Movimientos del Año para un Histórico (no está en el módulo actual)
- Limpia las tablas de movimientos y pone en cero el campo en el fichero de consecutivos para los No. de Documentos automáticos por almacén-documento.
- Borra las prefacturas de más de 60 días de creadas.
- Cambia la fecha para el 1ro. / Enero del siguiente año.

1.3.2 Descripción Actual del Dominio del Problema

La gestión de inventario es uno de los procesos más complejos que se realizan en una empresa, ya que deben controlar los movimientos que se realizan en los almacenes, como Entradas/Salidas de productos, Inventarios Físicos, Cierres Contables y otros. En los grandes almacenes estos procesos se acomplejan

mucho más. Por tal motivo se han venido implementando sistemas que automaticen esta gestión agilizando de esta forma dichos procesos, entre los que se encuentra el sistema ConDor y más específicamente este módulo que digitaliza todos los procesos que se realizan en el almacén y en el control de los mismos.

En sus inicios las empresas y entidades realizaban sus operaciones en papel duro. Esto conllevaba a que muchas de estas operaciones tuvieran considerables errores de cálculos, retraso en el tiempo entrega, entre otras dificultades. Para darle solución a todas estas contradicciones se creó, un Sistema de Gestión automatizado que agilizara todos estos procesos y al mismo tiempo minimizara las posibilidades de errores. Así nace en 1995 el Sistema de Gestión Contable “ConDor”.

En sus inicios, este módulo fue implementado para el Sistema Operativo (S.O) MS-DOS, que en aquellos tiempos era el S.O más usado por las empresas cubanas. Debido a los grandes avances de la tecnología, el S.O MS-DOS se fue quedando obsoleto, las herramientas y tecnologías utilizadas fueron caducando, los ordenadores se hacían más potentes y requerían potentes sistemas. Los clientes comenzaron a instalar el S.O Windows, por lo que El ConDor se había quedado pequeño ante los nuevos recursos tecnológicos y fue necesario implementarlo al nuevo sistema que traía nuevas ventajas; tenía interfaz gráfica, más rapidez y seguridad para el cliente, comodidad, y un contexto monousuario, por lo que se hizo necesario implementarlo en la nueva plataforma sobre Windows.

1.3.2.1 Actualidad del sistema

El Sistema de Gestión Económico Contable “ConDor” repercute favorablemente en la certificación contable, pues registra todas las operaciones realizadas en la contabilidad de manera automatizada desde todos sus módulos, permitiendo la transparencia en el proceso, además garantiza el control de los recursos y activos¹⁰.

¹⁰ Conjunto de todos los bienes y derechos con valor monetario que son propiedad de una empresa, institución o individuo, y que se reflejan en su contabilidad.

Este Sistema permite la prevención de indisciplinas de ilegalidades al brindar resultados identificados como puntos vulnerables y conservar la traza de sus movimientos. Garantiza la confiabilidad y seguridad en los análisis a realizar, facilitando información fehaciente a los directivos, consultores y auditores de la misma, permitiéndose mayor calidad en la toma de decisiones, así como un buen análisis económico y financiero de la entidad.

Cuando una empresa adquiere el Sistema Contable ConDor sus beneficios fluyen hacia dos direcciones:

- I. Impacto económico favorable al obtener una Contabilidad confiable debido a la disminución significativa de errores humanos, agilidad y oportunidad en los procesos.
- II. La utilización en muchas entidades, redundando en mejoras continuas en los servicios postventa por parte del equipo de desarrollo del software, que se mantiene actualizado tanto en la legislación vigente para esos fines, como en las sugerencias de los propios clientes que lo explotan.

1.3.3 Fundamentación de la Situación Problemática

Se ha planteado como **situación problemática** el entorno monousuario con que cuenta el módulo Inventario del Sistema Contable “ConDor”, de modo que ha sido señalado por parte de los clientes como una significativa desventaja trayéndoles consigo consecuencias negativas como la pérdida de tiempo. El SICS trabaja para certificar todos sus productos y para lograr que el ConDor sea certificado debe cumplir ciertas condiciones que exige el MIC, entre las que se encuentra el tener un entorno multiusuario.

También se debe señalar que el ConDor, presenta una Base de Datos (BD) independiente para cada módulo, cuestión esta que ha traído grandes errores en el funcionamiento del sistema ya que muchos de los módulos se relacionan entre sí, utilizando campos comunes de las tablas de la BD y estos no han podido relacionarse por este detalle. Además ha traído grandes dificultades a la hora de analizar en Contabilidad General los comprobantes de cada uno de los diferentes módulos, trayendo consigo errores que se han convertido en pérdida de dinero.

Al mismo tiempo se analiza lo siguiente: Todas las empresas cubanas no tienen las mismas condiciones técnicas de hardware, y una aplicación en Windows requiere un proceso de instalación a diferencia de una aplicación Web, detalle que no solo conlleva al gasto de tiempo sino también de memoria de la máquina, aspecto que no todas las empresas están en condiciones de asumir. Además, el proceso de instalación muchas veces suele ser algo complejo para aquellas personas que tienen poco conocimiento informático. En este caso para el módulo Inventario, la modelación de una aplicación Web donde su Base de Datos sea centralizada y única, es un paso de avance para la nueva aplicación ConDor, significaría ganar ventajas, pues se contaría con gran impulso de la aplicación siendo UML muy útil en el desarrollo de la misma. En el próximo capítulo se exaltarán algunas de las características y ventajas del lenguaje de modelado UML.

1.4 Análisis de otras Soluciones Existentes

Tanto en Cuba como en otros países se han desarrollado sistemas para agilizar y mejorar la Gestión de Inventarios. Entre estos sistemas está el propio ConDor en sus diferentes versiones que ha sido distribuido por casi mil empresas de todo el país y mencionado en epígrafes anteriores. A continuación se explicarán brevemente, algunos de estos software existentes.

El Sistema de Gestión Contable desempeña un papel fundamental en la medición de los resultados financieros. En cuanto al procesamiento de los datos contables, hoy en día está bastante automatizado en la mayor parte de las empresas. Es importante insistir en el impacto que la informática ha supuesto en todos los aspectos de gestión de la contabilidad: diario, mayor, cuentas a cobrar y pagar, elaboración de la nómina, obtención estados contables como el balance, la cuenta de pérdidas y ganancias, el cuadro de financiación o el estado de valor añadido proporcionan información contable cuantitativa. Debido a que la información cualitativa es más difícil de tratar con las tecnologías actuales, documentos como la memoria no están completamente automatizados.

1.4.1 GIB –Gestión de Inventario de Bienes

La herramienta GIB – Gestión de Inventario de Bienes para Sistemas Operativos Windows ha sido desarrollada para gestionar de forma eficaz la elaboración y mantenimiento del Inventario de Bienes y Derechos de las Administraciones Locales, automatizando las amortizaciones y simplificando el trabajo del personal encargado de su mantenimiento (INFAPLIC.ES 2006).

Las principales características del software de GIB son:

- Ajustado a la legislación de las Administraciones locales.
- Control de usuarios, permisos de acceso y auditoria, donde cada usuario se identifica en la entrada a la aplicación para tener un control de sus privilegios.
- Parametrización del programa.
- Procedimientos avanzados y restringidos como la amortización masiva del inventario. Exportación de datos y comunicación con otras aplicaciones (contabilidad, GIS, Excel, Lotus.).
- Posibilidad de almacenamientos de fotos y planos de los bienes en la ficha de cada bien.
- Registro de las sucesivas mejoras realizadas sobre cada bien. Se puede realizar un informe anual con las altas, bajas y modificaciones realizadas sobre los bienes.
- Consultas simples y avanzadas al inventario.
- Emisión de listados y fichas de cualquier tipo de bien. Impresión de etiquetas con códigos de barras para identificar cada uno de los bienes
- Cuenta con una interfaz sencilla, cómoda e intuitiva.

1.4.2 Versat - Sarasola

Este es un software creado por el licenciado en Economía y villaclareño Miguel Cabrera González, este es un sistema de contabilidad confiable, ofrece mayor organización, control y disciplina en cada gestión, posibilita enviar información eficaz, de forma inmediata y desde lugares apartados.

Este proyecto resulta ser un sistema integrado, constituido por 12 módulos que incluyen configuración y seguridad, contabilidad general y de gastos, costos y procesos, análisis económico empresarial y control

de activos fijos, además de intervenir finanzas y cajas, planificación y presupuestos, control de inventarios, de productos terminados, pago de salario, paquete de gestión, contratación y facturación.

Este producto le ha traído grandes beneficios al país, pues se ahorró un millón 186 mil dólares que costaban las licencias al evitarse la importación del sistema foráneo, más labor de consultaría, atenciones y otros gastos, además de que actualmente lo utilizan alrededor de 200 entidades de varias provincias y siguen aumento los clientes (MARTIN 2005).

1.4.3 SIP- Sistema de Inventario del Patrimonio Cultural y Nacional.

El sistema de Inventario del Patrimonio Cultural y Natural (SIP) que se desarrolla desde 1988, fue creado con diversos fines (CNPC.CULT.CU 2006):

- Facilitar el cumplimiento de la legislación nacional e internacional en lo que respecta al registro de los bienes patrimoniales.
- Contribuir a ejercer el control y a priorizar la conservación de los bienes patrimoniales más valiosos del país.
- Facilitar el intercambio de información relacionado con el patrimonio cultural y natural.
- Responder a las necesidades de diferentes usuarios: museólogos, museógrafos, conservadores, investigadores y otros con intereses más generales.

El SIP contempla bases de datos conformadas con los bienes de relevancia que integran el patrimonio cultural del país. Los que se encuentran en las instituciones museales y los que poseen y custodian organismos e instituciones estatales de connotada significación.

Este formado por 20 base de datos que tienen un formato común y en cada manifestación se han elaborado listados de términos y Tesoros que facilitan la entrada, búsqueda y recuperación de la información. En su diseño han participado grupos de especialistas de alta calificación en cada materia, informáticos y especialistas en computación.

Entre sus ventajas están:

- Capacidad de manejar un número ilimitado de bases de datos.
- Poderoso lenguaje de formateo, incluyendo enlace entre registros y búsqueda en tablas.
- Permite el intercambio de formato con otros sistemas convirtiendo archivos con formato ISO desde o hacia el formato de CDS/ISIS (el programa ISODB3 y DB3ISO permiten el intercambio entre sistemas que utilizan archivos con formato .DBF).

1.5 Conclusiones parciales

A través de este capítulo se han detallado algunas de sus generalidades sistema ConDor, y se ha particularizado en la composición del módulo analizado. También se abordó la actualidad del sistema para acercarnos a la situación problemática; se analizaron algunas soluciones existentes, con el objetivo de brindar apoyo a la solución del problema científico.



TENDENCIAS Y TECNOLOGÍAS ACTUALES UTILIZADAS

“La tecnología es positiva solo si la sabemos aprovechar, utilizándola para nuestras más profundas y valiosas intenciones.”

Jacobo.

Capítulo **2**

Tendencias y Tecnologías actuales utilizadas

2.1 Introducción

En este capítulo se detalla todo el entorno tecnológico en el cual se basa y se desarrolla la aplicación. Primeramente se realiza un análisis sobre la Web, debido a que esta constituye el **campo de acción** de la investigación y es el entorno en que se va a desarrollar, por su importancia, se caracteriza, se hace referencia a sus tipos, estructuras, ventajas, entre otros aspectos importantes.

La metodología a utilizar es un punto importante de este capítulo, por esto se conceptualiza, mencionan y explican brevemente en qué consisten algunas de ellas, enfatizando en la utilizada, RUP. El lenguaje de modelado de objetos (UML) es otro de los epígrafes tratados y se explican características específicas de él. Se fundamenta las herramientas utilizadas, tanto la herramienta de diseño como para el futuro desarrollo de la aplicación

La arquitectura desarrollada constituye un tema fundamental. Aquí se aborda el modelo cliente /servidor, modelo que guía la implementación del sistema, la plataforma. NET en la cual se va a asentar la aplicación y se hace énfasis en la programación multicapa, patrón arquitectónico que rige el diseño. No podían faltar los lenguajes de programación utilizados donde se fundamente la selección de dichos lenguajes utilizando como base las ventajas y desventajas que ofrecen cada uno de ellos.

2.2 Aplicaciones Web

Cuando se habla de aplicaciones Web, a lo que nos referimos es a la construcción de sitios que hagan algo, sitios que permitan introducir información y que respondan inteligentemente a las solicitudes. La

Web introduce un concepto fundamental: la posibilidad de lectura universal, que consiste en que una vez que la información esté disponible, se pueda acceder a ella desde cualquier ordenador, desde cualquier país, por cualquier persona autorizada, usando un único y simple programa. Para que esto fuese posible, se utilizan una serie de conceptos, el más conocido es el hipertexto.

Una aplicación Web es una interfase entre un formulario diseñado específicamente para cubrir con las necesidades de su negocio y su información que actualmente tiene hacia el interior de su empresa, como pueden ser: sistemas administrativos, inventarios, facturación, cuentas por cobrar, productos, etc. (La información puede ser de dominio público o restringida a ciertas personas a través de un nombre de usuario y contraseña) con el objetivo de que cualquier persona pueda consultarla e interactuar con ella desde Internet.

Con las aplicaciones Web el usuario no sólo recibe páginas del servidor⁶ como respuesta a su solicitud, sino que puede también enviar información de regreso a través de formularios. Están diseñadas para interactuar con bases de datos con el fin de recoger, almacenar, organizar y distribuir información, creando herramientas poderosas a ser utilizadas en la administración consistente de la información.

La Web recupera el papel del servidor que se convierte ahora en un servidor Web. Los datos se almacenan en bases de datos accesibles desde un navegador Web o una Terminal móvil gracias a los métodos que se ejecutan en el servidor y al diseño del interfaz que es transferido a dichas terminales. Todos conocemos buenos ejemplos de este tipo de programas, como puede ser el correo electrónico basado en Web, las comunidades en línea y los sitios de Comercio Electrónico que se están expandiendo a un ritmo acelerado.

Estructura

Aunque muchas variaciones son posibles, una aplicación Web está comúnmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador Web es la primera capa, un motor usando alguna tecnología Web dinámica (ejemplo: PHP, ASP) es la capa de en medio, y una base de

⁶ Más adelante se explicará en qué consiste el término de páginas en el servidor.

datos como última capa. El navegador Web manda peticiones a la capa media, que la entrega valiéndose de consultas y actualizaciones a la base de datos generando una interfaz de usuario.⁷

Ventajas

Son numerosas las ventajas y beneficios que otorga el poseer un sitio Web puesto que el mantenimiento del sistema se concentra en el servidor, el gasto se reduce. En general, es el proveedor del servicio quien se preocupa de tener la aplicación siempre disponible y actualizada. Se podrían encontrar numerosas ventajas más, pero probablemente, la propiedad más destacada sea la conectividad que proporciona Internet permitiendo el acceso a la aplicación desde cualquier punto. Esto permite ahorrarnos invertir en costosas infraestructuras de comunicaciones que en muchos casos podrían ser sencillamente imposibles.

Teniendo en cuenta la complejidad y el funcionamiento de las aplicaciones Web, se pueden clasificar en dos grandes tipos: las aplicaciones de Web estáticas, las cuales son mucho más sencillas y las aplicaciones Web dinámicas.

2.2.1 Web dinámicas

A diferencia de las páginas Web estáticas, una **página dinámica** es una página Web personalizada por el servidor de aplicaciones antes de que la página se envíe a un navegador. Los sitios Web dinámicos son aquellos que permiten crear aplicaciones dentro de la propia Web, otorgando una mayor interactividad con el navegante. Aplicaciones dinámicas como encuestas y votaciones, foros de soporte, libros de visita, envío de e-mails inteligentes, reserva de productos, pedidos on-line, atención al cliente personalizada.

El desarrollo de este tipo de Web es más complicado, pues requieren conocimientos específicos de lenguajes de programación así como creación y gestión de bases de datos, pero la enorme potencia y servicio que otorgan este tipo de páginas hace que merezca la pena la inversión y esfuerzo invertidos respecto a los resultados obtenidos.

⁷ En el epígrafe 2.4.2 se explica en qué consiste dicha estructura.

Entre sus principales **ventajas** se encuentran:

- Registrar usuarios en una base de datos.
- Restringir acceso a páginas.
- Administrar grandes cantidades de información o fotografías desde una base de datos.
- Cargar la información desde la misma Web.
- Encuestas.
- Mostrar los usuarios activos.
- Recabar información con formularios que se cargan en una base de datos.
- Recibir la información de los formularios por e-mail.
- Poseer CARRITO DE COMPRAS.

Su principal **desventaja** es lo complejo que puede llegar a ser su implementación.

2.3 Arquitectura a implementar

“Una arquitectura de software es el producto de trabajo de desarrollo que ofrece la mayor inclinación a invertir en lo que se refiere a calidad, planificación y coste”

Len Bass

“La arquitectura de sistema constituye un amplio marco que describe su forma y estructura – sus componentes y cómo estos encajan juntos”

Jerrold Grochow.

En este epígrafe titulado “Arquitectura a implementar” se abordará sobre el modelo cliente/servidor, la plataforma .NET y sobre la programación multicapas. Estos tres elementos arquitectónicos rigen el desarrollo de la aplicación.

2.3.1 Modelo Cliente/Servidor

IBM define al modelo Cliente/Servidor: "Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso del grupo de trabajo y/o, a través de la organización, en múltiples plataformas. El modelo soporta un medio ambiente distribuido en el cual los requerimientos de servicio hechos por estaciones de trabajo inteligentes o "clientes", resultan en un trabajo realizado por otros computadores llamados servidores".

La arquitectura cliente-servidor llamado modelo cliente-servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realizada se efectúe con la mayor eficiencia, y permita simplificarlas (WIKIPEDIA.ORG 2006a).

Características del modelo.

En el modelo Cliente-Servidor podemos encontrar las siguientes características:

1. Pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.
2. Un servidor da servicio a múltiples clientes en forma concurrente.
3. Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.
4. Un sistema de servidores realiza múltiples funciones al mismo tiempo que presenta una imagen de un solo sistema a las estaciones Clientes. También es importante hacer notar que las funciones Cliente/Servidor pueden ser dinámicas. Ejemplo, un servidor puede convertirse en cliente cuando realiza la solicitud de servicios a otras plataformas dentro de la red.
5. Designa un modelo de construcción de sistemas informáticos de carácter distribuido.

Ventajas de la arquitectura Cliente/Servidor.

- El servidor no necesita tanta potencia de procesamiento, parte del proceso se reparte con los clientes.

Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre para otra conexión.

2.3.1.1 Programación en lado del Cliente. Funciones

Cliente: Es el que inicia un requerimiento de servicio. Es el ordenador que pide información a otro mediante la aplicación de un programa llamado cliente

Con la programación del lado del cliente se pueden validar algunos de los datos en la máquina cliente antes de enviarlos al servidor. Esto proporciona a los usuarios informes de error inmediatos, mientras siguen en esa página de formulario y sin necesidad de volver atrás tras recibir un mensaje de error. Puede resultar necesario acceder a una base de datos para validar determinados valores, mientras que no suele disponer de un acceso directo a la base de datos en la máquina del cliente, aunque ese acceso a la base de datos es factible.

Sus funciones son:

- Contacta con el servidor.
- Da formato a la petición de información.
- Da formato a la respuesta.

2.3.1.2 Programación en lado del Servidor. Funciones

"Servidor": Es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente. Es el ordenador que ofrece la información mediante la aplicación de un programa llamado servidor.

La programación del lado del servidor es un elemento agregado muy importante en el diseño o construcción de sitios Web, ya que permite de una u otra forma el manejo de datos de forma dinámica.

Este tipo de programación da la posibilidad de ejecutar programas en un gran sistema, añadir y recuperar datos de una base de datos, y construir una página con los resultados queda fuera de las posibilidades de HTML estándar y programación del lado del cliente (REYES 2006).

Realiza las siguientes funciones:

- Recibe la petición y la procesa.
- Responde enviando la petición al cliente.

2.3.2 Programación Multicapas

La programación en múltiples capas⁸ facilita una administración eficiente de los componentes que la integran, proporciona rapidez a todas las funcionalidades Cliente/Servidor y la magnitud de la aplicación lo exige. Tal y como plantea la arquitectura de esta tecnología, los componentes primarios de la aplicación serán divididos y programados por separados y en tiempo de ejecución serán unidos. De forma tal que si una de las capas definidas sufre cambios, no se vean afectados el resto de las capas ni el resultado final del producto.

Se definen tres capas, tal es el caso de: Presentación, Reglas del Negocio y Acceso a Datos.

- **Presentación:** En esta capa se diseña todo lo que constituye la interfaz gráfica y la interacción del usuario con el sistema.
- **Reglas del Negocio:** Contiene todas las subrutinas creadas con el propósito de regular alguna acción del usuario.
- **Acceso a Datos:** En esta capa se programa todo lo que tiene que ver con el acceso a la base de datos. Esta capa queda encargada de tomar la información de la base de datos dada una petición de la capa de Reglas del Negocio, que a su vez es generada por la capa de presentación.

⁸ Partes bien definidas de un sistema, definidas a partir de un paquete o un subsistema.

Ventajas del modelo.

- Desarrollos paralelos (en cada capa).
- Aplicaciones más robustas debido al encapsulamiento.
- Mantenimiento y soporte más sencillo (es más sencillo cambiar un componente que modificar una aplicación monolítica).
- Mayor flexibilidad (se pueden añadir nuevos módulos para dotar al sistema de nueva funcionalidad).
- Alta escalabilidad. La principal ventaja de una aplicación distribuida bien diseñada es su buen escalado, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware. El crecimiento es casi lineal y no es necesario añadir más código para conseguir esta escalabilidad (PERAL 2004).

2.3.3 Plataforma .NET

Plataforma .NET es una tecnología sencilla pero muy potente para el desarrollo y ejecución de aplicaciones. Esta infraestructura se adapta a todo un conjunto de lenguajes y servicios que posibilitan un ambiente de desarrollo distribuido que permite la implementación de aplicaciones robustas.

".NET es una plataforma llena de servicios para construir aplicaciones basadas en Web y desarrollar experiencias interactivas para los usuarios y sus sistemas" (MORAN 2006).

.NET es un conjunto de tecnologías diferentes. Incluye:

- Los lenguajes .NET.
- El entorno de ejecución común (CLR).
- Las bibliotecas de clases.
- ASP: NET.
- Visual Studio .NET.

Existen más de 40 lenguajes adaptados a .Net, entre estos se encuentran: Visual Basic .NET, Visual C# .NET, Visual C++ .NET, Visual J#, entre otros. Además ofrece plena interoperabilidad entre ellos, por lo que es posible construir un componente en un lenguaje, introducirlo en una aplicación escrita en otro e incluso heredarlo y añadir nuevas características en un tercero, pero el lenguaje nativo que ha sido desarrollado con el objetivo de ser utilizado en .NET es C# (MORAN 2006).

La plataforma .Net ha revolucionado la informática en el siglo XXI porque es la primera plataforma que aprovecha al máximo las ventajas y posibilidades de este mundo digitalizado. Esta persigue llevar los conceptos de sencillez y facilidad a límites insospechados y todo esto con una sola finalidad: la calidad del software.

Con la biblioteca ADO.NET para acceso a bases de datos se permite realizar consultas y modificaciones a una base de datos de forma off-line, es decir:

1. Se realiza una conexión para traer los datos necesarios.
2. Se libera la conexión.
3. A continuación se trabaja con esos datos, y como se ha liberado la conexión se permite que otros usuarios se conecten a la base de datos.

A continuación se resumen algunas **ventajas** importantes que proporciona .Net:

- **Código administrado:** El CLR realiza un control automático del código para que este sea seguro, es decir, controla los recursos del sistema para que la aplicación se ejecute correctamente.
- **Interoperabilidad multilenguaje:** El código puede ser escrito en cualquier lenguaje compatible con .Net ya que siempre se compila en código intermedio (MSIL). Se puede mezclar grupo de programadores expertos en diferentes lenguajes.
- **Seguridad de acceso al código:** Se puede especificar que una pieza de código tenga permisos de lectura de archivos pero no de escritura. Es posible aplicar distintos niveles de seguridad al código, de forma que se puede ejecutar código procedente de la Web sin tener que preocuparse si esto va a estropear el sistema.
- **Despliegue:** Por medio de los ensamblados resulta mucho más fácil el desarrollo de aplicaciones distribuidas y el mantenimiento de las mismas. El Framework realiza esta tarea de forma automática

mejorando el rendimiento y asegurando el funcionamiento correcto de todas las aplicaciones (DESARROLLOWEB.COM 2006).

- **Escalabilidad:** La escalabilidad es la capacidad de un sistema de incrementar sus prestaciones en función del número de usuarios simultáneos que lo utilizan. La plataforma .NET puede escalar desde 16.000 transacciones por minuto a más de 500.000 transacciones por minuto, logrando dar un servicio de forma simultánea.

2.3.3.1 ASP. NET

ASP .NET es un entorno de programación que forma parte de la nueva plataforma .NET, ideal para crear aplicaciones y servicios Web.ASP .NET es mucho más que la siguiente versión de ASP. Su arquitectura ha sido totalmente reconstruida para facilitar al máximo la creación de aplicaciones Web dinámicas. Permite a los desarrolladores escribir código más limpio y más fácil de reutilizar y compartir, incrementando el rendimiento y la escalabilidad al poder acceder a lenguajes compilados, no interpretados.

Mejoras de ASP.NET

- Rendimiento: la aplicación de compila en una sola vez al lenguaje nativo, y luego, en cada petición tiene una compilación Just In Time, es decir se compila desde el código nativo, lo que permite mucho mejor rendimiento. También permite el almacenamiento del caché en el servidor
- Rapidez en programación: mediante diversos controles, podemos con unas pocas líneas y en menos de 5 minutos mostrar toda una base de datos y hacer rutinas complejas.
- Servicios Web: trae herramientas para compartir datos e información entre distintos sitios.
- Seguridad: tiene diversas herramientas que garantizan la seguridad de nuestras aplicaciones (FAHNLE 2006).

2.4 Metodología

La metodología⁹ en el desarrollo de un software es el plano de apoyo, la guía que nos conducirá a realizar un software de calidad, es el conjunto de métodos que se deben realizar para realizar un software con calidad. Con la selección y utilización de la metodología utilizada tendremos clientes más complacidos y desarrolladores satisfechos, además servirá para que el desarrollo y confección del software no parezca complicado, riguroso y difícil de controlar.

En un proyecto de desarrollo de software la metodología define ¿Quién debe hacer Qué?, ¿Cuándo y Cómo debe hacerlo? Una metodología es un proceso.

No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable (COLECTIVO_DE_AUTORES 2005-2006).

Algunas Metodologías.

Extreme Programing (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto (SANCHEZ 2006).

Lo fundamental en este tipo de metodología es:

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

⁹ Conjunto de reglas para organizar, diseñar y construir una aplicación.

Microsoft Solution Framework (MSF)

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas (SANCHEZ 2006).

MSF tiene las siguientes características:

- Adaptable.
- Escalable.
- Flexible.

2.4.1 Fundamentación de la Metodología utilizada. Proceso Unificado de Desarrollo

Luego de un extenso análisis y relacionando las características de la metodología con los requerimientos y características de la aplicación, se determinó que la metodología indicada para la realización de la aplicación Web de este tipo era el Proceso Unificado de Desarrollo (RUP).

El Proceso Unificado de Desarrollo es un proceso de desarrollo de software que utiliza el estándar de modelado visual, el Lenguaje Unificado de Modelado (UML). Está basado en componentes. Es dirigido en caso de uso, centralizado en la arquitectura¹⁰, iterativo e incremental. Este proceso es polifacético pues divide en cuatro fases¹¹ el desarrollo del software (inicio, elaboración, construcción y transmisión) en cada una se obtiene un producto final y cada una está desarrollada mediante el ciclo de iteraciones, las cuales tienen como función la evaluación de las iteraciones precedentes. En cada iteración se necesita el uso de

¹⁰ La arquitectura se usa como un artefacto primordial para la conceptualización, construcción, gestión y evolución del sistema en desarrollo.

¹¹ Período de tiempo entre dos hitos principales de un proceso de desarrollo.

artefactos, cosa que lo convierte en una metodología importante para alcanzar un grado de certificación de desarrollo del software.

El modelo de desarrollo del RUP está compuesto por: la Ingeniería de negocios, Requerimientos, Análisis y Diseño, Implementación y Prueba. Esta metodología también evalúa Gestión de Riesgos, Control de Calidad, Gestión de Proyectos y Control de Configuraciones. Todas estas facetas se encuentran de forma integrada.

Características del RUP.

Los verdaderos aspectos definitorios del Proceso Unificado, y que lo convierten en único, se resumen en tres frases clave - dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental.

Dirigido por Casos de Uso, se basa en lo que el usuario/cliente necesita y desea, se capturan a través de las diferentes vías que se modelan en el negocio y se plasman en los requisitos, luego los modelos que se realizan guían todo el proceso, o sea, avanza a través de una serie de flujos de trabajo que se derivan de los casos de uso.

Proceso Centralizado en Arquitectura: Incluye aspectos estáticos y dinámicos según los requerimientos del sistema. Es la Arquitectura la que se ajusta a los requerimientos del software y requisitos del hardware para ajustarlo a la vista del diseño completo de nuestro sistema. Es la vista total del proyecto la cual se aprueba en total acuerdo entre el equipo y los usuarios. Describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.

Proceso Iterativo e Incremental: Se reproducen ciclos de vida en cascadas a menor escala. En cada una de estas iteraciones se hace referencia a los pasos del flujo de trabajo y se van incrementando según el crecimiento de producto. El objetivo de cada iteración se establece en la función de la evaluación de las iteraciones precedentes.

En cada iteración, los desarrolladores, se identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura seleccionada como guía, implementa el diseño mediante componentes y verifican que los componentes satisfacen los casos de uso (JACOBSON *et al.* 2004).

2.5 Lenguaje para el Modelado de Objetos

Un lenguaje para el modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos. El uso de un lenguaje de modelado es más sencillo que la auténtica programación, pues existen menos medios para verificar efectivamente el funcionamiento adecuado del modelo. Esto puede suponer también que las interacciones entre partes del programa den lugar a sorpresas cuando el modelo ha sido convertido en un software funcionando (WIKIPEDIA.ORG 2006b).

Toda metodología de desarrollo de software utiliza de un lenguaje para el modelado de objetos para la representación de sus diagramas y artefactos.

2.5.1 Lenguaje Unificado de Desarrollo

UML es un lenguaje estándar para el modelado de software – lenguaje para visualizar, especificar y documentar los artefactos del sistema con gran cantidad de software. Lenguaje usado por el Proceso Unificado. Lenguaje que permite a los desarrolladores visualizar el producto de su trabajo (artefacto) en esquemas o diagramas estandarizados (JACOBSON *et al.* 2004).

UML es la interrelación Elementos – Relaciones – Diagramas, es un modelo gráfico que incluye aspectos conceptuales como de negocio y del sistema y aspectos concretos como expresiones del lenguaje de programación, esquema de Base de Datos y componentes de software reutilizable. Es un lenguaje para especificar no es un método ni un proceso es una notación y este termino se llega a confundir. Soporta la

metodología RUP pero el no constituye una metodología. Se usa para definir un sistema, sus artefactos, documentos y construir el software.

El UML estándar está compuesto por 3 partes: bloques de construcción (tales como clases, objetos, mensajes), relaciones entre bloques (asociación, generación) y diagramas (por ejemplo el diagrama de actividades). Los perfiles de UML son las extensiones a las notaciones estándares del UML usando los mecanismos de extensión del UML: los estereotipos, los valores etiquetados y restricciones (CAMACHO PUPO 2006).

Se escoge el UML porque:

- Permite modelar sistemas utilizando técnicas orientadas a objetos (OO).
- Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, etc.).
- Es un lenguaje muy expresivo que cubre todas las vistas necesarias para desarrollar y luego desplegar los sistemas.
- Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.
- UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.
- La metodología seleccionada utiliza UML.

2.6 Herramientas a utilizar

“Las herramientas de la Ingeniería de software¹² proporcionan un enfoque automático o semi-automático para el proceso y para los métodos. Cuando se integran para que la información creada por una herramienta la pueda utilizar otra, se establece un sistema de soporte para el desarrollo del software llamado en ingeniería del software asistida por computadoras (CASE).”

Roger S. Pressman

Para desarrollar la propuesta del software se hace necesario el uso de varias herramientas. Estas herramientas facilitan el trabajo, haciendo que este se realice de una forma mucho más rápida y eficiente. Para su análisis y fundamentación, las herramientas utilizadas se han dividido en dos grupos: las herramientas de modelado y las herramientas de desarrollo.

2.6.1 Herramienta de Modelado

Las herramientas de modelado de objetos, son fundamentales para el análisis del sistema. Para modelar la aplicación se utilizó el Rational Rose Enterprise Edition.

2.6.1.1 Rational Rose Enterprise Edition

El Rational Rose es una herramienta CASE¹³ para “modelado visual”, que forma parte de un conjunto más amplio de herramientas que juntas cubren todo el ciclo de vida del desarrollo de software. Permite completar una gran parte de flujos fundamentales del Proceso Unificado de Rational (RUP). Es la mejor herramienta para traducir requisitos de alto nivel a una arquitectura flexible basada en componentes.

¹² Establecimiento y uso de principios robustos de la ingeniería a fin de obtener económicamente software que sea fiable y que funcione eficientemente sobre máquinas reales.

¹³ Computer Aided Software Engineering. Herramientas apropiadas para aplicaciones complejas, que requieren de modificaciones con regularidad.

Rational se encuentra a la cabeza en cuanto al desarrollo del Unified Modeling Language (UML), que se ha convertido en la notación estandarizada empleada en Rational Rose para especificar, visualizar y construir desarrollos de software y sistemas.

Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes **ventajas** de Rose es que permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto (CAMACHO and SANCHEZ 2006).

Características principales:

- Admite como notaciones: UML, COM, OMT y Booch.
- Realiza Chequeo semántico de los modelos.
- Ingeniería “de ida y vuelta”: Rose permite generar código a partir de modelos y viceversa.
- Desarrollo multiusuario.
- Integración con modelado de datos.
- Generación de documentación.
- Tiene un lenguaje de script para poder ampliar su funcionalidad.
- Disponible en múltiples plataformas.

2.6.2 Herramientas de desarrollo

En toda aplicación Web para lograr un buen desarrollo se necesitan herramientas que posibiliten el modelado del diseño, otras para la implementación, además, un gestor de Base de Datos (BD).

En el caso de la aplicación Web de ConDor se quiere utilizar como herramienta para el modelado: Rational Rose Enterprise Edition, para la implementación: Microsoft Visual Studio 2005 y SQL Server 2005 Express como gestor de BD.

2.6.2.1 Microsoft Visual Studio 2005

Visual Studio 2005 proporciona una amplia gama de herramientas que ofrecen multitud de ventajas para desarrolladores individuales y equipos de desarrollo de software (MSDN 2005b)

Se selecciona Microsoft Visual Studio 2005 porque:

- Tiene mayor productividad y obtención más rápida de resultados.
- Crea soluciones dinámicas basadas en Windows, la Web, dispositivos móviles y Office.
- Tiene comunicación y colaboración más eficaz en sus equipos de software.
- Garantiza calidad rápida y continua en todo el proceso de desarrollo.
- Hay más ediciones diferenciadas por el precio y las características.
- Ayuda con refactorización.
- Añadido soporte de tests para todo tipo de aplicaciones.

2.6.2.2 SQL Server 2005 Express

SQL Server Express es una versión de SQL Server 2005 diseñada para ayudar a los desarrolladores a construir aplicaciones robustas y fiables ofreciendo una sencilla pero potente base de datos que es además gratuita. SQL Server Express ya está incluida como opción al instalar cualquier herramienta Visual Studio Express (MSDN 2005a).

SQL Server 2005 Express Edition proporciona muchas nuevas funcionalidades mejoradas, como son:

- Una base de datos robusta para crear aplicaciones dinámicas.
- Fuerte soporte de XML.
- Herramientas y características para mejorar la gestión y facilidad de uso.
- Instalación y setup sencillos.
- Escalabilidad y rendimientos.
- Gran integración con Visual Studio 2005.
- Robusta seguridad.
- La instalación típica de SQL Server 2005 Express requiere 500MB de espacio en disco.

SQL Server Express es un producto de base de datos gratis y fácil de utilizar basado en tecnología SQL Server 2005. El mismo está diseñado para brindar una plataforma de base de datos que ofrece una simplicidad de uso superior, permitiendo instalaciones más rápidas en sus escenarios. La facilidad de uso comienza con una instalación robusta de la interfase del usuario gráfica que guía al usuario a través del proceso de instalación. El diseño y desarrollo de aplicaciones de bases de datos se vuelve más sencillo con la integración de Visual Studio. Además, presenta la habilidad de instalar aplicaciones de bases de datos al moverlas como archivos Windows típicos. Los servicios y correcciones también son simples y automatizados (MSDN 2005a).

SQL Server Express utiliza el mismo motor de base de datos confiable y de alta seguridad que las otras versiones de SQL Server 2005. De hecho, se diferencia del resto de las ediciones de SQL Server 2005 por lo siguiente:

- Falta de características de soporte enterprise.
- Limitado a una sola CPU.
- Límite de memoria de un GB para el buffer.
- Las bases de datos tienen un tamaño máximo de 4GB.

2.7 Lenguajes de Programación

Para desarrollar la propuesta del software se hace necesario el uso de una tecnología. En este caso se usará la tecnología orientada a objetos ya que esta permite la manipulación de los objetos para un mejor desarrollo del software. Como lenguaje de programación se usarán C# y SQL.

2.7.1 C#

C# es el nuevo lenguaje de propósito general orientado a objetos creado por Microsoft para su nueva plataforma .NET. Combina la potencia de lenguajes como C++ con la sencillez de lenguajes como Visual Basic, y que además la migración a este lenguaje por los programadores de C/C++/Java sea lo más inmediata posible.

Aunque es posible escribir código para la plataforma .NET en muchos otros lenguajes, C# es el único que ha sido diseñado específicamente para ser utilizado en ella, por lo que programarla usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes ya que C# carece de elementos heredados innecesarios en .NET. Por esta razón, se suele decir que C# es el lenguaje nativo de .NET

Características de C#.

- **Sencillez:** C# elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET. Por ejemplo:
 - El código escrito en C# es autocontenido, lo que significa que no necesita de ficheros adicionales al propio fuente tales como ficheros de cabecera
 - El tamaño de los tipos de datos básicos es fijo e independiente del compilador, sistema operativo o máquina para quienes se compile
 - No se incluyen elementos poco útiles de lenguajes como C++ tales como macros, herencia múltiple o la necesidad de un operador diferente del punto (.) acceder a miembros de espacios de nombres (::)

- **Modernidad:** C# incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular, como un tipo básico decimal que permita realizar operaciones de alta precisión con reales de 128 bits (muy útil en el mundo financiero), la inclusión de una instrucción foreach que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico string para representar cadenas o la distinción de un tipo bool específico para representar valores lógicos.
- **Orientación a objetos:** Como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos. Una diferencia de este enfoque orientado a objetos respecto al de otros lenguajes como C++ es que el de C# es más puro en tanto que no admiten ni funciones ni variables globales sino que todo el código y datos han de definirse dentro de definiciones de tipos de datos, lo que reduce problemas por conflictos de nombres y facilita la legibilidad del código.

C# soporta todas las características propias del paradigma de programación orientada a objetos: encapsulación, herencia y polimorfismo.

Presenta soporte XML para la interacción de componentes basado en Web. Realiza la arquitectura de una amplia gama de componentes, desde el objeto de negocios de alto nivel hasta aplicaciones al nivel del sistema. Ha sido diseñado para conseguir más, con menos líneas de código y menos posibilidades de error (CORPORATION 2001).

2.7.2 SQL (Structure Query Language)

Es mucho mejor manejar datos desde bases de datos, que escribir y leer datos para/desde archivos de texto. SQL nos brinda esa posibilidad.

El SQL (Structured Query Language), es un lenguaje de consulta estructurado, es un lenguaje surgido de un proyecto de investigación de IBM para el acceso a bases de datos relacionales. Actualmente se ha

convertido en un estándar de lenguaje de bases de datos, y la mayoría de los sistemas de bases de datos lo soportan, desde sistemas para ordenadores personales, hasta grandes ordenadores.

SQL es un lenguaje para describir todas las acciones u operaciones que se realizan sobre los motores de las bases de datos relacionales. Se realizan para consultar y programar bases de datos, acceder a los datos y para consultar, actualizar y gestionar sistemas de bases de datos.

Se divide en dos secciones:

1. Las DDL: (Data Description Language), lenguaje de definición de datos, incluye órdenes para definir, modificar o borrar las tablas en las que se almacenan los datos y de las relaciones entre estas. (Es el que más varía de un sistema a otro). Normalmente las instrucciones incluidas son CREATE, ALTER, DROP aplicadas sobre diferentes objetos.
2. Las DML: (Data Manipulation Language), lenguaje de manipulación de datos, nos permite recuperar los datos almacenados en la base de datos y también incluye órdenes para permitir al usuario actualizar la base de datos añadiendo nuevos datos, suprimiendo datos antiguos o modificando datos previamente almacenados. La manipulación se hace a nivel de tuplas (filas) (CLIC 2007).

Algunas de las características del SQL son:

- Es una forma estándar de consulta de datos específicos.
- Es una forma de extraer y manipular datos de una base de datos.
- Usado para todas las funciones de bases de datos, incluyendo administración.
- Creación de esquemas y datos recuperables.
- Puede ser usado de forma implícita dentro de una aplicación.

2.8 Conclusiones parciales

En este capítulo se analiza las tendencias y tecnologías actuales, dando los conceptos necesarios para comprender el por qué de las tecnologías que serán utilizadas en el desarrollo de la aplicación. Se

fundamenta la metodología a utilizar UML. Se hizo un estudio de las aplicaciones Web y el modelo cliente servidor, de los lenguajes de programación Web. Además se aborda de las herramientas de desarrollo a utilizar para la concepción del sistema, explicado su gran potencialidad y facilidades que brinda a la hora de desarrollar la aplicación y por último se fundamentan los lenguajes de programación a utilizar.



CARACTERÍSTICAS DEL SISTEMA

“La parte más dura de la construcción de un sistema software es decidir como construirlo...Ninguna parte del trabajo mutila el resultado del sistema si está mal hecho. Ninguna parte es más dificultosa para rectificarlo después.”

Fred Brooks.

Capítulo **3**

Características del sistema.

3.1 Introducción

Para optimizar la comprensión de la solución propuesta para el problema planteado, en este capítulo se realiza el análisis del proceso de negocio correspondiente al objeto de estudio y se comienza con la modelación del sistema propuesto.

Primeramente se realiza todo el análisis del negocio del problema. Posteriormente se realiza el proceso de la captura de requerimientos del sistema, tanto los funcionales como no funcionales, para lo cual se utilizaron los métodos planteados en la Introducción. Por último se pasa a la construcción de los casos de uso del sistema y a los diagramas que representan a la solución propuesta del sistema y para finalizar la descripción textual de cada uno de ellos.

3.2 Modelo de Negocio

Para modelar el proceso de negocio, primeramente se seleccionan los actores y trabajadores que intervienen en el negocio, luego se definen los casos de usos que lo integran conformando el diagrama de caso de uso del negocio. Por último con los trabajadores y los objetos utilizados en el proceso se confecciona el modelo de objetos del negocio, cumpliendo así con la salida de todos los artefactos del flujo de trabajo.

3.2.1 Actores y Trabajadores del Negocio

Un actor del negocio es cualquier persona, grupo, entidad, máquina, programa, sistema de información externo que interactúa con el negocio. Lo que se modela como actor es el rol que se juega cuando se beneficia de los resultados del negocio. El actor siempre se encuentra en las fronteras del negocio que estamos investigando. A continuación se muestra la justificación de los actores y trabajadores del negocio.

| Actores del negocio | Justificación |
|-----------------------|--|
| Directivo | Es una generalización de los directivos de la entidad. Realizan las solicitudes para la salida de productos del almacén para la producción o insumo. |
| Jefe de Brigada | Persona que está al frente de una brigada de producción. Entrega los productos terminados de la producción al almacén y puede realizar pedidos para las salidas de productos para la producción. |
| Jefe de Departamento | Persona que dirige un departamento de la entidad. Es un directivo que autoriza la salida de mercancías para el uso de su departamento. |
| Jefe de Contabilidad | Persona que dirige el departamento de contabilidad de la entidad. Encargado de crear la comisión para la realización de inventario físico y autoriza la salida de productos del almacén. |
| Comercial Solicitante | Persona que realiza las operaciones comerciales en una entidad solicitante de mercancías. Solicita transferencia y transfiere a dependencias. |
| Proveedor | Entidad encargada de suministrar productos a entidades. |

Tabla 3.1. Actores del Negocio.

Un trabajador del negocio representa a personas, o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso. Estos trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema que se quiere construir. Cada trabajador del negocio debe definirse brevemente con su responsabilidad dentro del negocio (DEPARTAMENTO_INGENIERÍA_SOFTWARE 2005-2006).

| Trabajadores del negocio | Justificación |
|---------------------------------|--|
| Jefe de Producción | Persona responsable de la producción de la empresa. Gestiona dicha producción. |
| Jefe Comisión de Inventario | Persona que dirige la comisión que realiza inventario físico. Llena la hoja de inventario al terminar de realizarlo y gestiona la triple tarjeta. |
| Especialista de Inventario | Contador que gestiona el Inventario de la empresa y desempeña la mayoría de las actividades que se realiza en el control de inventario. |
| Comercial | Persona encargada de gestionar las compras y transferencias. Confecciona el modelo de solicitud de compra y el modelo de despacho que se utiliza como autorización de salidas del almacén. |
| Almacenero | Persona que trabaja directamente en el almacén, desempeña un importante rol en el control de inventario. Llena los Modelos de Entrega, Modelos de Transferencias y el Modelo de Control Interno de la Producción, además inicia la confección del informe de recepción y confecciona los Vales de Salida de productos del almacén. |

Tabla 3.2. Trabajadores del Negocio.

3.2.2 Proceso de Negocio

En el negocio de control de inventario del sistema ConDor se realizan tres procesos fundamentales: la Recepción, el Control de Almacenamiento y las Salidas (todos estos procesos se realizan en el almacén).

La Recepción o Entradas de mercancías al almacén, pueden ser principalmente por tres vías: mediante las Compras, por Transferencias entre almacenes dependientes y por Producción Terminada, en el caso que la entidad se dedique a la producción.

El control de almacenamiento consiste en el control de la mercancía que está en existencia en el almacén. Este control se lleva a cabo por dos procesos: El Inventario Físico que verifica y chequea todos los productos del almacén, y mediante los Ajustes que contabilizan las anomalías en los productos en existencia.

Las Salidas del almacén pueden realizarse fundamentalmente por dos vías¹⁴: mediante Las Transferencias a almacenes pertenecientes a la entidad por solicitud de esta y para la Utilización en la Producción o Insumo de la empresa.

3.2.3 Diagrama de Caso de Uso del Negocio

Este modelo de casos de uso describe los procesos del negocio en términos de casos de uso del negocio y actores del negocio, representando en estos últimos a los clientes. Dicho modelo se describe mediante diagramas de casos de uso que muestran cómo los actores del negocio y los casos de uso del negocio se encuentran relacionados.

A continuación se representa el modelo de negocio de la aplicación:

¹⁴ Pueden existir salidas por otras causas por ejemplo por la venta, pero estas operaciones no se encuentran dentro del alcance de trabajo.

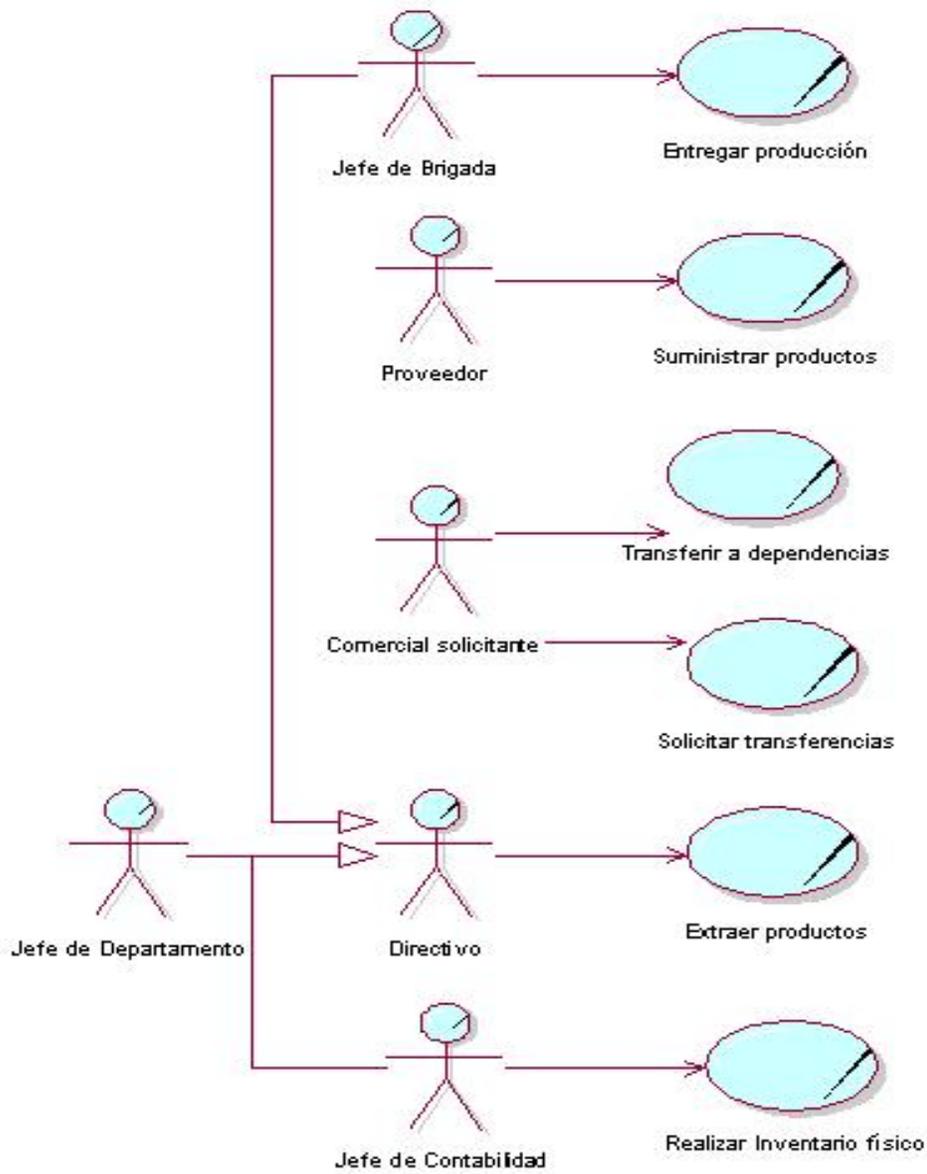


Fig. 3.1. Modelo del Negocio.

3.2.4 Descripción Textual de los Casos de Uso del Negocio

| | | |
|---|--|--|
| Caso de Uso: | Suministrar Productos | |
| Actores: | Proveedor (Inicia) | |
| Trabajadores: | Comercial, Almacenero, Especialista en Inventario. | |
| Resumen: | El caso de uso se inicia cuando proveedor oferta los productos que están en venta. El Comercial solicita la compra. Luego de ser pagada es entregada por el proveedor y recibida y almacenada por el almacenero. Estos datos son introducidos ala sistema por el especialista en inventario. | |
| Precondiciones: | El Comercial debe de entregar el Modelo de solicitud de compras. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Negocio | |
| 1. El Proveedor oferta los productos en venta. | 1.1. El Comercial realiza la solicitud de compra, llenando el Modelo de solicitud de compra. | |
| 2. El proveedor recibe la solicitud y hace la prefactura. | 2.1. El Comercial recibe la prefactura y efectúa el pago. | |
| 3. El proveedor recibe pago y envia la mercancía. | 3.1. El Almacenero recibe la mercancía y comienza la confección del Informe de recepción. 3.2 El especialista en Inventario termina el Informe de Recepción e introduce los datos en la PC. | |
| Poscondiciones | Se satisface el défisis de productos. | |

| | |
|----------------------|---|
| Caso de Uso: | Entregar Productos |
| Actores: | Jefe de Brigada (inicia) |
| Trabajadores: | Jefe de Producción, Almacenero y Especialista en Inventario. |
| Resumen: | El caso de uso se inicia cuando el Jefe de Brigada entrega la producción terminada al almacén, entregando también el Modelo de Orden de Producción confeccionado pro el Jefe de Producción. Estos productos son recibidos por el Almacenero quien los almacena y contabiliza. Estos datos |

| | | |
|--------------------------------|---|--|
| | son introducidos al sistema por el Especialista en Inventario. | |
| Precondiciones: | Tiene que llevar para la entrega en el almacén la Orden de Producción. | |
| Flujo Normal de Eventos | | |
| | Acción del Actor | Respuesta del Negocio |
| | 1. El Jefe de Brigada entrega la producción terminada | 1.1. El Jefe de Producción recibe la producción y confecciona el Modelo de Producción. 1.2 El Jefe de producción producción entrega la producción y el modelo al almacén. |
| | | 2.1. El Almacén recibe el modelo y la producción terminada y llena el Modelo Control Interno de la Producción. 2.2. El Especialista en Inventario introduce los datos en la PC. |
| Flujos Alternos | | |
| | Acción del Actor | Respuesta del Negocio |
| | Actividad 1.2. Si el Jefe de Brigada no entrega la Orden de Despacho no se puede recibir los productos en el almacén. | |
| Poscondiciones | Se evalúa la producción. | |

| | |
|------------------------|---|
| Caso de Uso: | Transferir a Dependencias |
| Actores: | Comercial solicitante (inicia) |
| Trabajadores: | Comercial, Almacenero y Especialista en Inventario. |
| Resumen: | El caso de uso se inicia cuando el Comercial solicitante realiza una solicitud de transferencia a un almacén dependiente. Esta es recibida por el Comercial quien elabora una Orden de despacho que autoriza al Almacenero la entrega de la mercancía. El especialista en inventario registra la transferencia. |
| Precondiciones: | Tiene que llevar para la entrega en el almacén la Orden de Despacho. |

| Flujo Normal de Eventos | |
|---|---|
| Acción del Actor | Respuesta del Negocio |
| 1. El Comercial solicitante realiza solicitud de transferencia. | 1.1 El Comercial recibe la solicitud y elabora la Orden de despacho. |
| | El almacenero al recibir la orden de despacho, accede a entregar la mercancía. El almacenero llena el Modelo de Transferencia. El especialista en inventario registra la transferencia. |
| Flujos Alternos | |
| Acción del Actor | Respuesta del Negocio |
| Actividad 2.1. Si el Jefe de Brigada no entrega la Orden de Despacho no se puede recibir los productos en el almacén. | |
| Poscondiciones | Se satisface el pedido. |

| | |
|------------------------|--|
| Caso de Uso: | Solicitar Transferencia. |
| Actores: | Comercial solicitante (inicia). |
| Trabajadores: | Almacenero y Especialista en Inventario. |
| Resumen: | El caso de uso se inicia cuando el Comercial solicitante recibe la solicitud de transferencia, este envía la mercancía, la cual es recibida por el Almacenero. El especialista en Inventario registra la transferencia recibida. |
| Precondiciones: | |

| Flujo Normal de Eventos | |
|--|--|
| Acción del Actor | Respuesta del Negocio |
| 1. El Comercial solicitante recibe la solicitud de transferencia y envía la mercancía. | 1.1. El Almacenero recibe la mercancía y confecciona el modelos de transferencia. 1.2 El Especialista en Inventario registra la transferencia recibida. |

| | |
|-----------------------|---------------------------------------|
| Poscondiciones | Se satisface el déficit de mercancía. |
|-----------------------|---------------------------------------|

| | |
|------------------------|---|
| Caso de Uso: | Extraer Productos |
| Actores: | Directivo (inicia). |
| Trabajadores: | Almacenero y Especialista en Inventario. |
| Resumen: | El caso de uso se inicia cuando el Directivo autoriza la entrega de uno o varios productos para insumo o para la producción. El Almacenero recibe la solicitud, entrega la mercancía y llena el vale de salida. Estos datos son introducidos al sistema por el Especialista en Inventario. |
| Precondiciones: | Par sacar los productos del almacén tiene que llevar el Modelo de entrega |

Flujo Normal de Eventos

| Acción del Actor | Respuesta del Negocio |
|---|---|
| 1. El Directivo realiza la solicitud de entrega confeccionando el Modelo de entrega | 1.1. El Almacenero recibe el modelo y entrega la mercancía. |
| | 1,2. El Almacenero confecciona el Vale de salida. 1.3. El Especialista en Inventario introduce los datos en la PC. |
| Poscondiciones | Se satisface la necesidad de un área. |

| | |
|------------------------|---|
| Caso de Uso: | Realizar Inventario Físico. |
| Actores: | Jefe de Contabilidad (inicia). |
| Trabajadores: | Jefe de la Comisión, Miembros de la Comisión y Especialista en Inventario. |
| Resumen: | El caso de uso se inicia cuando el jefe de Contabilidad crea la Comisión de Inventario y selecciona su jefe, para realizar inventario físico. El Jefe de la Comisión selecciona el 10% a inventariar y llena la Triple Tarjeta. Los miembros del equipo realizan los conteos y se cuadra el inventario. El final se llena la Hoja de Inventario Físico. |
| Precondiciones: | Seleccionar el 10 % mensual de todos lo productos del almacén. |

| Flujo Normal de Eventos | |
|--|--|
| Acción del Actor | Respuesta del Negocio |
| 1. El Jefe de Contabilidad crea la Comisión de Inventario | |
| 2. El Jefe de Contabilidad selecciona el Jefe de la Comisión. | 1.1. El jefe de la Comisión selecciona el 10% a inventariar y llena la Triple Tarjeta. 1.2. Los Miembros de la Comisión realizan el conteo físico y entregan los datos al Jefe de la Comisión. 1.3. El Jefe de la Comisión recibe los datos y chequea las diferencias. 1.4. Si no hay diferencias llena la Hoja de Inventario Físico. |
| Flujos Alternos | |
| Actividad 1.4. Si hay diferencias se vuelve a contar y se informa para se ajustes. Para esto se le informa al especialista en inventario quien realiza el ajuste | |
| Poscondiciones | El inventario queda cuadrado. |

Tabla 3.3. Descripción de los Casos de Uso del Negocio.

3.2.6 Modelo de Objetos

El diagrama de clases del Modelo de Objeto, es un artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos. [Anexo 1].

3.3 Especificación de los Requisitos del Software

Los requerimientos del sistema es uno de los aspectos más importantes a considerar cuando se desarrolla un software, pues estos constituyen la base, el fundamento de la solución propuesta y el argumento para desarrollar el modelado del sistema. Por eso la captura de los requerimientos es uno de los procesos críticos de la ingeniería del software.

“El proceso de reunión de requisitos se intensifican y se centran específicamente en el software. Para comprender la naturaleza del (los) programas (s) a construirse, el ingeniero (analista) del software debe comprender el dominio de la información del software, así como la función requerida, comportamiento, rendimiento e interconexión.”

Roger S. Pressman

Los requerimientos constituyen las especificaciones de los clientes según sus necesidades en el sistema. Como dijo Bernard Boar: “Los desarrolladores pueden construir y probar las especificaciones, pero los usuarios son los que en realidad aceptan o rechazan la operativa actual”.

3.3.1 Requerimientos Funcionales del Sistema

Luego de ser analizado el problema de origen y los posibles procesos para automatizar, encontrado con el modelo de negocio fueron identificados los siguientes requisitos funcionales para el desarrollo del sistema propuesto:

R1 - Autenticarse en el Sistema.

R2 - Cambiar Contraseña.

R3 - Gestionar Usuario.

3.1 - Crear usuario

3.2 - Modificar usuario.

3.3 - Eliminar usuario.

3.4 - Mostrar usuarios.

3.5 - Mostrar reporte de usuario.

R4 - Gestionar Roles.

4.1 - Insertar nuevo rol.

4.2 - Modificar nuevo rol.

4.3 - Eliminar nuevo rol.

R5 - Gestionar Almacenes.

5.1 - Insertar almacén.

5.2 - Modificar almacén.

5.3 - Eliminar almacén.

5.4 - Mostrar almacenes.

R6 - Mostrar Tipos de Movimientos.

R7- Realizar Transferencia de Materiales.

7.1 Mostrar reporte.

R8 - Gestionar Ajustes.

8.1 - Insertar ajuste.

8.2 - Modificar ajuste.

8.3 - Eliminar ajuste.

8.4 - Mostrar ajustes.

R9 - Gestionar Órdenes de Trabajo.

9.1 - Insertar orden.

9.2 - Modificar orden.

9.3 - Eliminar orden.

9.4 - Mostrar órdenes.

R10- Gestionar Cuentas Inventario.

10.1 – Mostrar cuentas.

10.1 – Modificar cuenta.

R11 - Gestionar Opciones Contables.

11.1 - Insertar opción.

11.2 – Eliminar opción.

11.3 – Modificar opción.

11.4 – Mostrar opciones.

R12 - Gestionar Tarjetero de Productos.

12.1 - Insertar producto.

12.2 - Modificar producto.

12.3 - Mostrar tarjetero.

R13 - Cierre Apertura.

R14 - Listar Reportes.

14.1 - Vinculados a Contabilidad.

14. 2 - Vinculados a compras.

14.3 - vinculados a gasto de materiales.

14.4 - Vinculados a tarjetero de productos.

14.5 - Vinculados a inventario físico.

14.6 - Vinculados a movimientos.

14. 7 - Resumen del mes.

14.7.1 - Ajustes.

14.7.2 - Transferencia de materiales.

14.7.3 - Transferencia entre almacenes.

14.7.4 - Cancelaciones de movimientos.

14.7.4 - Entradas.

14.7.4 - Salidas.

R15 - Gestionar Control Interno de la Producción.

15.1- Insertar producto.

15.2- Eliminar producto.

15.3- Modificar producto.

15.4- Mostrar productos.

R16 - Gestionar Vales de Entrada/Salida.

16.1 - Insertar vale de entradas o salidas.

16.2 - Eliminar vale de entradas o salidas.

16.3 - Mostrar vale de entradas o salidas.

16.4 - Mostrar reporte.

R17 - Realizar Informe de Recepción.

17.1 – Mostar reporte.

R18 - Realizar Informe de Recepción. A ciegas.

R19 - Realizar Transferencia entre Almacenes.

R20 - Ajustar Precio.

R21 – Realizar Cambio Cuenta.

R22 - Gestionar Inventario Físico.

22.1 - Seleccionar método.

22.2 - Fijar fecha existencia.

22.3 - Entrar datos (conteo físico).

22.4 - Listar diferencias.

R23 - Realizar Cambio de CUP.

R24 - Ver Disponibilidad.

24.1 - Productos en escasés.

24.2 - Productos en caducidad.

24.3 - Ver existencia de productos.

R25 - Cancelar Movimientos.

R26 - Hacer Cierre Día.

R27- Hacer Cierre Mes.

R28 - Hacer Cierre Año.

R29 - Fijar fecha.

3.3.2 Requerimientos no Funcionales del Sistema

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Los requerimientos no funcionales del sistemas son fundamentales para el buen funcionamiento del sistema, aquí radica su importancia.

“El entorno de un sistema en tiempo real a veces contiene dispositivos que actúan como los 5 sentidos del sistema”

Paul Ward y Stephen Mellor.

A continuación se presentan los requerimientos no funcionales para esta aplicación:

Requerimientos de usabilidad.

Este sistema está concebido para ser usado por diferentes usuarios que no necesariamente tengan conocimientos informáticos, por consiguiente debe ser práctico y fácil de usar.

Requerimientos de rendimiento

El sistema debe ser eficiente y preciso en la información que le suministra al usuario para evitar cualquier tipo de error. El tiempo de respuesta ante cualquier solicitud del usuario debe ser el mínimo posible por lo que debe implementar varias transacciones por segundo para dar una respuesta rápida al usuario y evitar demoras, además, se deben poder conectar varios usuarios a la BD y satisfacer sus peticiones en un corto plazo de tiempo. Debe estar disponible todo el tiempo para trabajar. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas en la BD.

Requerimientos de soporte

Para garantizar el soporte a los clientes de esta herramienta, se documentará la aplicación con un manual de ayuda para los usuarios y el administrador, así como la posibilidad de emitir sus quejas y sugerencias a los desarrolladores de la herramienta, por correo o por teléfono, realizar mantenimiento al sistema y con la importancia de los procesos contables y el rápido avance de las tecnologías, se necesitaran posteriores versiones.

Requerimientos de portabilidad

El sistema se almacena en soporte CD para de esta forma llevarlo a sus clientes donde será instalado.

Requerimientos de seguridad

1. **Confidencialidad**: La información manejada por el sistema deberá estar protegida de acceso no autorizado y divulgación. Los usuarios accederán a la información correspondiente a cada uno.
2. **Integridad**: La información manejada por el sistema será objeto de cuidadosa protección contra estados inconsistentes y corrupción.

3. **Disponibilidad**: A los usuarios autorizados se les deberá garantizar el acceso a la información solicitada en todo momento.
4. **Encriptación**: La información de autenticación (contraseña) se encriptará para el envío por la red. El proceso de encriptado se hará en el cliente usando el lenguaje Java Script. El algoritmo de encriptación será MD5.
5. **Firewall**: El servidor debe de implementar un firewall para la protección, se recomienda Zone Alarm.
6. **Contraseña**: Las contraseñas usadas en los servidores debe de ser de 10 o más caracteres y caducan en un período de 1 mes. Las contraseñas usadas por los usuarios de la aplicación debe ser de 7 o más caracteres y caducan en un período de un mes. En ambos casos las contraseñas deberán usar la combinación de números, letras y símbolos (".", "-", "+", "_"). El SQL Server debe de llevar una contraseña combinada al igual que las demás.

Requerimientos legales

Tanto esta aplicación como la información utilizada en la construcción de la aplicación, pertenecen al SICS, específicamente al Departamento de Software. Debe cumplir con las normas, leyes y políticas de seguridad establecidas por el MFP (Ministerio de Finanzas y Precios), por el MIC y por la entidad.

Requerimientos de interfaz

Como la aplicación propuesta será usada por personas que no necesariamente tienen habilidades en el trabajo con la computadora, debe ser una interfaz amigable, legible, interactiva, fácil de usar, profesional, clara, sencilla y debe mantener el mismo formato en todas las páginas. La interfaz de usuario está conformada por páginas Web basadas en formularios, con la posibilidad de realizar tratamiento diferenciado a los usuarios que accedan al sistema y de esta forma se logra una interfaz lo más cercana posible a lo que desea el usuario. La resolución de la aplicación será de 800 x 600. Ante cualquier cambio de resolución de pantalla, la resolución de la aplicación no debe cambiar.

Requerimientos de confiabilidad

Además de ser seguro es muy confiable ya que garantiza que los datos lleguen íntegros e intactos a su destino. Los datos son almacenados en una BD segura.

Requerimientos de ayuda y documentación en línea

El sistema debe implementar una ayuda online disponible al usuario en todo el momento de forma tal que el usuario que esté en el sistema pueda visitarla en caso de cualquier duda, esta debe brindarle información de todas las actividades a realizar por él. El software debe tener la documentación completa de todas las tareas y operaciones que realiza así como todo sobre su implementación.

Requerimientos de instalación

Para la instalación se debe de crear dos instaladores, uno para todas páginas y otro para la BD.

Requerimientos de software

El software se debe ejecutar sobre una plataforma Windows 2000, XP o superior para la PC clientes con un navegador de Internet. En la computadora que haga función de servidor se necesita Windows Server 2003, además del SQL Server 2005 para implantación de la BD.

Requerimientos de hardware

Para el servidor:

- Procesador: Procesador de 3 gigahertz o superior.
- RAM: Memoria RAM de 2 GB o superior.
- Disco duro: 80 GB de capacidad, 7200 rpm.
- Equipo: Pentium IV.

Para el cliente:

- RAM: Memoria RAM de 128 GB o superior.
- Equipo: Celeron, Pentium II o superior.

3.4 Definición de los Casos de Uso

Determinación y justificación de los actores del Sistema.

| Actores | Justificación |
|--------------------|---|
| Usuario | Es una representación de todos los usuarios del sistema. |
| Esp. de Inventario | Es uno de los actores principales del sistema. Se beneficia de la mayoría de las acciones. |
| Almacenero | Los únicos permisos que tiene asignado son los que tienen relación directo con el trabajo en el almacén. |
| Contador Principal | Es el que dirige y controla todas las operaciones y realiza las de mayores prioridades. Es el Administrador del sistema. |
| Comercial | Es el encargado de ver la disponibilidad de los productos en existencia para gestionar las compras y transferencias. |
| Administrador | Es el encargado de gestionar los usuarios y los roles de los mismos, además fija la fecha de operaciones del sistema. |

Tabla 3.4. Actores del Sistema.

3.4.1 Diagrama de Caso de Uso del sistema

Debido a la gran cantidad de casos de usos que presenta el sistema y para lograr una mejor organización del trabajo se decidió agrupar la mayor cantidad de casos de usos en paquetes. Para ello se asignaron los casos de uso según el criterio de agrupamiento para dar soporte a una determinada organización en el sistema. De esta forma se crearon 4 paquetes:

- Paquete Administración: En este paquete se agrupan los casos de usos relacionados con la administración del sistema, CU que solamente son realizados por el administrador del sistema.
- Paquete Servicios Comunes: En este paquete se agrupan los casos de usos que son realizados por el actor Usuario, o sea, los CU que son realizados por todos los actores del sistema.
- Paquete Fase Apertura: En este paquete se agrupan todos los casos de uso relacionados con las principales entradas del sistema.
- Paquete de Fase de Explotación: En este paquete se agrupan todos los casos de uso relacionados con las principales gestiones del control de inventario.

A continuación se muestra el Diagrama del sistema. Para ver los diagramas de CU para cada paquete remitirse a los anexos. [Anexo 2].

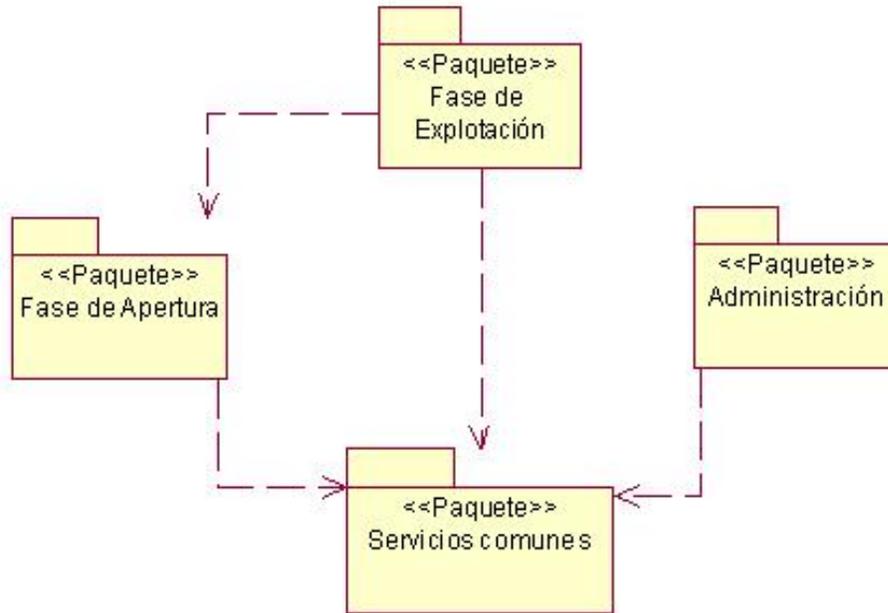


Fig. 3.4. Modelo del Sistema.

3.4.2 Expansión de los Casos de Uso

La expansión ampliada de los CU críticos se encuentra en los anexos [Anexo 3]

| Caso de Uso | Actor | Resumen |
|--------------|---------|---|
| Autenticarse | Usuario | El caso de uso se inicia cuando el usuario selecciona la opción de entrada al sistema, aparece una ventana de diálogo para entrar su nombre de usuario y contraseña. El usuario entra los datos, el sistema los verifica y si es correcto, verifica los permisos que tiene asignado por el administrador y entra a la aplicación con ese rol. |

CAPÍTULO 3. CARACTERÍSTICAS DEL SISTEMA

| | | |
|--|----------------------------|---|
| Gestionar almacenes | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario registra todos y cada uno de los almacenes de la entidad. El Especialista en Inventario puede insertar, eliminar, modificar y mostrar datos de los almacenes. |
| Gestionar control interno de la producción | Especialista Inventario | El caso de uso se inicia cuando el Jefe de Brigada solicita hacer cualquier operación relacionada con la producción. El Jefe de Brigada puede insertar, modificar, eliminar y listar productos. |
| Gestionar cuentas de inventario | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario solicita parametrizar las cuentas de control de inventario. |
| Gestionar Inventario físico | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario registra el inventario físico, lo cual puede seleccionar por varias opciones. |
| Gestionar nomencladores* | Especialista Inventario | Es una representación de caso de uso realizada para una mejor organización y comprensión del modelo de CUS en la Fase de Apertura. |
| Gestionar ordenes de trabajo | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario introduce, modifica, elimina y/o muestra las órdenes de trabajo solicitadas al almacén. |
| Gestionar Tarjetero de productos | Especialista Inventario | El caso de uso se inicia cuando el almacenero llena todos los formularios del tarjetero, llenando todos los campos. El Especialista en Inventario debe mantener actualizado el tarjetero. |
| Gestionar usuarios | Administrador | El caso de uso se inicia cuando el Administrador desea realizar cualquier operación con un usuario. Él puede insertar, eliminar, modificar o mostrar los datos de un usuario. |
| Gestionar vales de entrada/salida | Almacenero | El caso de uso se inicia cuando el Almacenero inserta, elimina o muestra cualquiera de las operaciones que se pueden |

* Está diferenciado en el diagrama del sistema correspondiente con un color gris.

| | | |
|--|-------------------------|--|
| | | realizar en el caso de uso. |
| Hacer cierre de apertura | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario solicita hacer cierre de apertura para pasar a una nueva fase de control de inventario. |
| Hacer cierre del año | Contador Principal | El caso de uso se inicia cuando el Contador Principal solicita cerrar el año. |
| Hacer cierre del día | Contador Principal | El caso de uso se inicia cuando el Contador Principal solicita cerrar el día. |
| Hacer cierre del mes | Contador Principal | El caso de uso se inicia cuando el Contador Principal solicita cerrar el mes. |
| Mostrar disponibilidad | Comercial | El caso de uso se inicia cuando el Comercial decide ver la disponibilidad de los productos para saber cuáles son las próximas compras a realizar o si se necesita solicitar alguna transferencia a los almacenes dependientes. |
| Realizar informe de recepción | Especialista Inventario | El caso de uso se inicia cuando el Especialista en Inventario llena todos los formularios correspondiente al informe, completando solamente los campos que le son activados por el rol que desempeña. |
| Realizar informe de recepción a ciegas | Almacenero | El caso de uso se inicia cuando el almacenero llena todos los formularios correspondiente al informe, completando solamente los campos que le son activados por el rol que desempeña. |
| Fijar fecha | Administrador | El caso de uso se inicia cuando el Administrador fija la fecha de operaciones. |

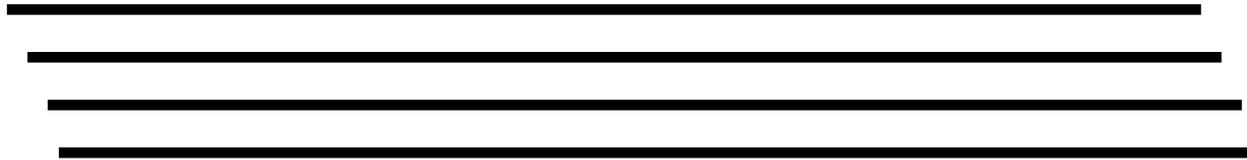
3.5 Conclusiones parciales

En este capítulo se cumple un objetivo fundamental del trabajo siendo este el modelamiento del negocio para la comprensión del problema a resolver. Se comenzó por un análisis profundo y completo del

proceso a automatizar y la información que manejaba para luego caer en la comprensión del negocio de la investigación. Se analizaron los requerimientos funcionales y no funcionales del sistema, respetando las solicitudes y necesidades de los clientes. Por último se diseñó el modelo del sistema donde se plantean todas las funcionalidades que el sistema debe cumplir y se hizo una descripción textual de todos los casos de usos especificando cuales eran los críticos, cumpliendo así uno de los objetivos y tareas propuestas.



CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA



“Hay dos formas de construir un diseño de software: una forma es hacer un diseño tan simple que no existan obviamente deficiencias, y la otra es hacer un diseño tan complicado que no existan deficiencias obvias. La primera forma es mucho más fácil.”

C.A.R. Itoare

Capítulo 4

Construcción de la solución propuesta.

4.1 Introducción

El análisis y diseño, es uno de los flujos de trabajo del RUP, metodología empleada en la modelación del sistema. Aunque esta metodología une al Análisis y Diseño en un solo flujo de trabajo ambos tienen funcionalidades diferentes. Estos dos procesos son muy importantes en el desarrollo de software dirigido por modelos ya que constituyen la vista lógica de la arquitectura del software.

En este capítulo se realiza el análisis del sistema, modelando su principal diagrama (Diagrama de Análisis), también se analiza los principios del diseño, donde se hace referencia a los estándares de interfaz, se hace una concepción de la ayuda y se fundamenta los tratamientos de errores. Para finalizar se realiza todo el proceso de diseño de la aplicación, donde se desarrolla el Diagrama de Clases de Diseño, el Modelo de Clases Persistente, el Modelo de Datos y el Modelo de Despliegue

4.2 Análisis

El análisis consiste en obtener una visión del sistema que se preocupa por ver ¿Qué hace?, de modo que sólo se interesa por los requisitos funcionales, transformar los requisitos funcionales en un diseño de clases viendo las relaciones e interacción que existe entre ellos, y teniendo en cuenta en el proceso una arquitectura robusta que permita adaptar el sistema al entorno de implementación que se está desarrollando.

4.2.1 Diagrama de Clases de Análisis

Una de las principales salidas o artefactos del análisis es el Diagrama de Clases de Análisis. En este se representan las Clases de Análisis (clase Interfaz, clase controlador y clases entidad), sus relaciones entre sí y con los actores del sistema. En una aplicación Cliente/Servidor de tres capas como la que se está desarrollando, en la capa de usuario aparecen fundamentalmente clases interfaz ya que allí se ejecutan las aplicaciones del cliente. En la capa intermedia están las clases de control donde se agrupan los servicios que son compartidos por múltiples aplicaciones. En la capa servidor estarían las clases entidad porque allí se tiene la base de datos.

Por la complejidad de los diagramas de clases de la aplicación, no se debe realizar un solo diagrama de clases para todo el sistema, ya que resulta complejo de comprender, por tal motivo se realiza la representación en paquetes¹⁵ de débil acoplamiento, los cuales coinciden con los subsistemas establecidos en la modelación del sistema. A su vez, estos paquetes están compuestos por otros paquetes correspondiendo a los casos de usos pertenecientes a cada uno de ellos. [Anexo 4]. Este particionamiento se realiza sobre la base de los requerimientos funcionales y el dominio del problema; y debe ser reconocible por las personas con conocimiento del dominio. [Anexo 5]

4.3 Principios de Diseño

4.3.1 Estándares de la Interfaz de la Aplicación

El sistema sigue la línea general de un diseño Web. Se le presenta al usuario una página principal donde en la parte izquierda se le brinda todas las opciones que el usuario tiene acceso según su rol en el sistema. Este diseño se mantiene en todas las páginas de la aplicación.

¹⁵ Mecanismo de organización de elementos que subdividen el modelo en otros más pequeños que colaboran entre sí. No agrupan funcionalidades.

En la parte superior aparece el logotipo que representa a este módulo. Presenta una navegación global pudiendo acceder a todos los módulos desde cualquier página. El color predominante es el amarillo, el cual representa el color del módulo, el fondo es gris y los botones funcionales grises con una tonalidad mas intensa. El tipo de letra utilizado es de color negro Arial o Verdana 10 para los textos y 11 para los encabezados, empleando para los hipertextos letras azules y subrayadas, esto se mantiene en todas las páginas de la aplicación. El idioma utilizado es el español.

4.3.2 Concepción General de la Ayuda

La ayuda es una opción del sistema que está accesible en todas las páginas de la aplicación y para todos los usuarios, con el objetivo que el usuario pueda esclarecer cualquier inquietud respecto a la manipulación de la aplicación y se familiarice con el sistema.

En la ayuda se debe mostrar una descripción de todas las opciones que la aplicación brinda y la manera de utilizarlas, logrando que el usuario logre evacuar sus dudas en caso de desconocimiento de alguna opción del sistema.

4.3.3 Tratamiento de Excepciones

El sistema trata de minimizar al máximo los posibles errores que puedan existir. En el caso de los datos introducidos por los usuarios, se trata que el usuario no tenga que teclear la información, sino que la pueda seleccionar de una lista y de esta manera siempre serán válidos los datos de entrada.

En caso contrario se hace una validación de datos utilizando las facilidades que brinda C# y Asp .Net. Al ocurrir una excepción esta se manipula con las posibilidades que nos brinda el lenguaje utilizado mostrando al usuario una información correcta y explicativa sobre el posible error cometido. Todos los mensajes de error se muestran en color rojo para que resalten.

4.4 Patrones

“Una arquitectura orientada a objetos bien estructurada está llena de patrones. La calidad de un sistema orientado a objetos se mide por la atención que los diseñadores han prestado a las colaboraciones entre sus objetos.”

“Los patrones conducen a arquitecturas más pequeñas, más simples y más comprensibles”

G. Booch

Un patrón es una pareja de **problema / solución** con un nombre, que codifica (estandariza) buenos principios y sugerencias. Es aplicable a otros contextos, con una sugerencia sobre la manera de usarlo en diferentes situaciones. El objetivo de los patrones es crear un lenguaje común a una comunidad de desarrolladores para comunicar experiencia sobre los problemas y sus soluciones. Pueden referirse a distintos niveles de abstracción, desde un proceso de desarrollo hasta la utilización eficiente de un lenguaje de programación.

4.4.1 Patrones de arquitectura

Para el diseño de la arquitectura se emplea el patrón de 3 capas porque:

- Se navega de las capas superiores a las inferiores.
- Se pueden hacer modificaciones en una capa sin afectar a otras.
- La capa de presentación son todas las páginas Web que el usuario puede ver.
- La capa de negocio es la formada por las clases controladoras, las de acceso a datos y las clases entidades.
- La capa de datos es la formada por la BD en SQL Server.
- Las capas se van a encontrar en diferentes servidores, en este caso la capa presentación en el cliente, la capa de negocio en el Servidor Web y la capa de datos en el Servidor SQL.

4.4.2 Patrones de diseño

Para el diseño se utilizan los cinco patrones de asignación de responsabilidades (GRASP):

- Experto: Le asigna la responsabilidad a la clase que tiene mayor información sobre la tarea a realizar. Ej. Clase CIN_Existencia.
- Creador: Posibilita la creación de instancia de cada clase por si misma. Ej. Clase CIN_Almacén.
- Alta cohesión: Reduce la dependencia, aumenta la complejidad pero es una complejidad manejable y eficiente. La responsabilidad se reparte en varias clases y existe una fuerte colaboración entre ellas.
- Bajo acoplamiento: Las clases mas independientes, reduce el impacto al cambio y más reutilizables: Ej. Clase CAD_Conexion.
- Controlador: Evita el acceso directo a las clases entidades, evita que la capa de presentación no maneje los eventos del sistema. Ej. Clase CC_Gestionar usuario.

4.5 Diseño

Con el diseño se logra la comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, Sistemas Operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario y se crea una entrada apropiada y un punto de partida para actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases.

Uno de las principales salidas del diseño lo constituyen el Diagrama de Clases del Diseño, el Diagrama de Clases Persistentes, el Modelo de Datos y el Diagrama de Despliegue.

4.5.1 Diagrama de Clases del Diseño

Los Diagramas de Clases son los más utilizados en el Modelado de Sistemas Orientados a Objetos. Un Diagrama de Clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa (JACOBSON *et al.* 2004).

El Diagrama de Clases del Diseño para Web, difiere un poco del resto de las aplicaciones que se construyen, puesto que en ellas son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. Para obtener un nivel correcto de abstracción y detalle que permita obtener un resultado final, es mejor modelar los artefactos del sistema, es decir: modelar las páginas, los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de estas, una vez que estén en el navegador del cliente; estos son los artefactos que se necesitan modelar para que el desarrollador los implemente luego y obtener así el producto final.

Para la confección de los diagramas de clases del diseño se mantuvo la misma estructura que se creó para el diagrama de clases de análisis, o sea, se mantiene la estructura en paquetes y subpaquetes. En dichos diagramas, las clases con prefijo “SP” son páginas servidoras, páginas que interactúan con recursos en el servidor y tienen código que se ejecutan solamente en él. Las clases con prefijo “CP” son las páginas cliente, o sea, instancias de páginas Web, con formularios HTML. Los cuales son representados con los prefijos “Fr” y son elementos de entradas y partes de las páginas clientes y por último las entidades del sistema. Las clases fueron estandarizadas comenzando todas con el prefijo CIN (referente a Control de Inventario) [Anexo 6]

4.5.2 Diagramas de Iteración (Secuencia)

Los Diagramas de Interacción se utilizan para modelar los aspectos dinámicos de un sistema. La mayoría de las veces, esto implica modelar instancias concretas o prototípicas de clases, interfaces, componentes y nodos, junto con los mensajes enviados entre ellos, todo en el contexto de un escenario que ilustra un comportamiento. Los Diagramas de Interacción pueden utilizarse para visualizar, especificar, construir y documentar la dinámica de una sociedad particular de objetos, o se pueden utilizar para modelar un flujo de control particular de un Caso de Uso (COLECTIVO _ AUTORES 2006-2007). . Estos diagramas son de dos tipos: de Colaboración o Secuencia. Para el diseño del sistema se decide desarrollar Diagramas de Secuencia.

Un Diagrama de Secuencia es un Diagrama de Interacción que destaca la ordenación temporal de los mensajes entre objetos.

Los Diagramas de Secuencia se desarrollan para cada escenario que presente el Caso de Uso. En la aplicación se cuenta con una gran cantidad de casos de usos y la mayoría de ellos son de tipo “gestionar”, o sea, son casos de usos que presentan tres o mas escenarios, por tal motivo el número de diagramas de secuencia es bastante considerable por esta razón no se incluyeron en los anexos del documento sino que se encuentran adjuntos en un disco extraíble

4.5.3 Diseño de la Base de Datos.

Una de las tareas trazadas es diseñar una BD centralizada para los siete módulos existentes del sistema. Debido al gran número de tablas que componen dicha BD solamente se representan las relaciones entre las tablas utilizadas en el Módulo Inventario y dentro de este la parte que se está analizando: Control de Inventario.

El diseño de la BD está compuesto fundamentalmente por dos artefactos: El Modelo de Clases Persistentes [Anexo 7] y el Modelo de Datos [Anexo 8]

4.5.4 Modelo de Despliegue

Un Diagrama de Despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución. Se modelan la topología del hardware sobre el que se ejecuta el sistema y la distribución física del sistema.

Para la puesta en práctica de la aplicación se necesitará principalmente cuatro procesadores: uno que servirá como servidor de la aplicación, el cual se encontrará conectado directamente al servidor de Base de Datos mediante el protocolo de comunicación TCP/IP y a las PCs clientes mediante el protocolo HTTP. Las PCs clientes utilizarán una impresora siendo esta un recurso compartido conectada a la red utilizando un Print Server mediante el protocolo TCP/IP. [Anexo 9]

4.6 Generalidades de la implementación

- Todos los campos de textos deben de ser validados en la interfaz antes de ser enviados a la BD.
- El algoritmo de encriptación a usar será el MD5 y será implementado en el cliente usando el lenguaje Java Script.
- Se debe de aplicar estándares para la codificación.
- Se debe implementar un mecanismo de seguridad para bloquear la BD en caso de ataques.

4.7 Conclusiones parciales

En este capítulo se representaron las diferentes partes de la solución propuesta a través de diferentes diagramas utilizando las extensiones de UML para modelar los elementos relativos a la aplicación.

Se ha presentado una vía completa y coherente por paquetes de los elementos específicos de la modelación Web de la aplicación con los niveles de detalle y abstracción apropiados para diseñadores, desarrolladores y arquitectos de aplicaciones Web; lográndose mostrar los resultados de la etapa de Análisis y Diseño del Sistema.

Se obtuvo el Diagrama de Clases de Análisis Diseño donde se representaron las clases y sus asociaciones. Se definió el diseño de la Base de Datos, creando para ello el Diagrama de Clases Persistentes, y a través de este, el Modelo de Datos. Además se confeccionó el Diagrama de Despliegue. También se presentó la concepción del Tratamiento de Errores, la concepción general de la Ayuda del Sistema, la concepción del sistema de Seguridad y Protección. Y se plantearon los principios de diseño de la aplicación Web.



ESTUDIO DE FACTIBILIDAD



“Lo que caracteriza a una inteligencia formada es que pueda descansar satisfecha con el grado de precisión que la naturaleza de un asunto permite, y no buscar la exactitud cuando solo una aproximación de la verdad es posible...”

Aristóteles

Capítulo 5

Estudio de Factibilidad

5.1 Introducción

La Planificación y la Estimación de un software es lo primero que se realiza para la modelación del mismo, la cual da una visión global de la aplicación. Aunque estas no son lo suficientemente exactas, ayudan a organizar y planificar el trabajo individual y el del equipo de trabajo. La Planificación implica la Estimación, su intento por determinar cuánto dinero, esfuerzo recursos, y tiempo supondrá construir un sistema o producto específico de software.

Es objetivo de este capítulo el análisis de la factibilidad del software; para eso se tomarán como base las estimaciones realizadas así como los beneficios que propicia el software. Para las estimaciones realizadas en este proyecto se tuvo en cuenta una metodología de estimación de costo. Como dijera Capers Jones: “Buenos enfoques de estimación y datos históricos sólidos ofrecen una mejor esperanza de que la realidad pueda vencer sobre las demandas imposibles.”

5.2 Planificación y Estimación

La Planificación realizada para esta aplicación fue basada en Casos de Uso. La Planificación es un método que permite la estimación del tiempo de desarrollo de un proyecto mediante la asignación de “pesos” a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores. Entre los factores más importantes se encuentran:

- Esfuerzo: tiempo que necesita una persona para trabajar en el desarrollo del proyecto (hombres/mes, hombres/días, hombres/horas).
- Tiempo, Cantidad de Personas: recursos necesarios para desarrollar el software.
- Costo: relacionado con el software.

A continuación, se detallan los pasos a seguir para la aplicación de éste método.

5.2.1 Cálculo de Puntos de Casos de Uso sin Ajustar (UUCP)

El primer paso para la estimación por casos de uso es el cálculo de los puntos de casos de usos sin Ajustar (UUCP), para esto se suma Factor de Peso de los Actores sin Ajustar (UAW) y del Factor de Peso de los Casos de Uso sin Ajustar (UUCW), mediante la ecuación: $PCU = FPA + FPCU$ donde:

- PCU: Puntos de Casos de Uso sin Ajustar (UUCP)
- FPA: Factor de Peso de los Actores (UAW)
- FPCU: Factor de Peso de los Casos de Uso (UUCW)

5.2.1.1 Calcular el Factor de Peso de los Actores sin Ajustar (FPA)

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Actores se establece teniendo en cuenta en primer lugar si se trata de una persona o de otro sistema, y en segundo lugar, la forma en la que el actor interactúa con el sistema. Los criterios se muestran en la siguiente tabla

| Tipo de Actor | Descripción | Factor de Peso |
|---------------|--|----------------|
| Simple | Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface). | 1 |
| Medio | Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto. | 2 |
| Complejo | Una persona que interactúa con el sistema mediante una interfaz gráfica. | 3 |

Tabla 5.1. Criterio de Factor de Peso de los Actores.

Clasificación de la complejidad de los actores de acuerdo a la naturaleza de los mismos.

| Actores | Peso | Complejidad |
|--------------------|-------------|--------------------|
| Almacenero | 3 | Complejo |
| Contador | 3 | Complejo |
| Esp. De Inventario | 3 | Complejo |
| Administrador | 3 | Complejo |
| Comercial | 3 | Complejo |

Tabla 5.2. Clasificación de la Complejidad de los Actores.

Si clasificamos a los actores de la aplicación mediante la tabla anterior veríamos que se tienen 5 actores todos con peso 3 por lo tanto:

$$FPA = 5 \times 3 = 15$$

5.2.1.2. Calcular el Factor de Peso de los Casos de Uso sin Ajustar (FPCU)

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La Complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómica, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.

Los criterios se muestran en la tabla siguiente:

| tipo de Caso Uso | Descripción | Factor de Peso |
|-------------------------|---|-----------------------|
| Simple | El Caso de Uso que contiene de 1 a 3 transacciones. | 5 |
| Medio | El Caso de Uso que contiene de 4 a 7 transacciones. | 10 |
| Complejo | El Caso de Uso que contiene más de 8 transacciones. | 15 |

Tabla 5.3. Criterio de factor de peso de los CU.

Clasificación de la complejidad de los Casos de Uso de acuerdo al número de transacciones.

| Casos de Uso | Transacciones | Factor de peso | Complejidad |
|---|----------------------|-----------------------|--------------------|
| Ajustar Precios | 3 | 5 | Simple |
| Autenticarse | 2 | 5 | Simple |
| Cambiar Contraseña | 2 | 5 | Simple |
| Cancelar Movimientos | 3 | 5 | Simple |
| Fijar fecha | 2 | 5 | Simple |
| Gestionar Ajustes | 9 | 15 | Complejo |
| Gestionar Almacenes | 9 | 15 | Complejo |
| Gestionar Control Interno de Productos | 9 | 15 | Complejo |
| Gestionar Cuenta de Control de Inventario | 3 | 5 | Simple |
| Gestionar Opciones Contables | 9 | 15 | Complejo |
| Gestionar Órdenes de Trabajo | 9 | 15 | Complejo |
| Gestionar Roles | 9 | 15 | Complejo |
| Gestionar Tarjetero de Productos | 6 | 10 | Medio |
| Gestionar Usuarios | 12 | 15 | Complejo |
| Gestionar Vales de Entrada/Salida | 7 | 10 | Medio |
| Mostrar Reportes | 17 | 15 | Complejo |
| Mostrar Tipos de Movimientos | 2 | 5 | Simple |
| Realizar Cambio de Cuentas | 3 | 5 | Simple |
| Realizar Cambio de CUP | 3 | 5 | Simple |
| Realizar Cierre de Apertura | 3 | 5 | Simple |
| Realizar Cierres Año | 6 | 10 | Medio |
| Realizar Cierres Día | 4 | 10 | Medio |
| Realizar Cierres Mes | 5 | 10 | Medio |
| Realizar Informe de Recepción | 2 | 5 | Simple |
| Realizar Informe de Recepción a ciegas | 2 | 5 | Simple |
| Realizar Inventario Físico | 10 | 15 | Complejo |
| Realizar Transferencia de Materiales | 3 | 5 | Simple |

| | | | |
|--|---|----|--------|
| Realizar Transferencia entre Almacenes | 4 | 10 | Medio |
| Ver Disponibilidad | 2 | 5 | Simple |

Tabla 5.4. Clasificación de Complejidad de los Casos de Uso.

Se tienen 29 casos de uso, de ellos 14 tienen complejidad simple con factor de peso 5, 6 tienen complejidad media con factor de peso 10 y 9 son complejos con factor de peso 15, de ahí se tiene que:

$$FPCU = (14 \times 5) + (6 \times 10) + (9 \times 15) = 265$$

Luego se calculan los PCU

$$PCU = FPA + FPCU$$

$$PCU = 15 + 265 = 280$$

5.2.2 Cálculo de los Puntos de Casos de Usos Ajustados (PCUA)

Después de calculados los PCU (sin Ajustar) estos se deben ajustar teniendo en cuenta un grupo de factores técnicos y ambientales según la ecuación:

$PCUA = PCU \times FCT \times FA$ donde:

- PCUA : Puntos de Casos de Usos Ajustados (UCP)
- FCT : Factor de Complejidad Técnica (TCF)
- FA : Factor de Ambiente (EF)

5.2.2.1 Calcular el Factor de Complejidad Técnica

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0

significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente tabla se muestra el significado y el peso de cada uno de éstos factores y se utiliza la fórmula:

$$FCT = 0.6 + 0.01 \times \sum (\text{Peso } i \times \text{Valor } i)$$

Clasificación del Factor de Complejidad Técnica del Sistema

| Factor | Descripción | Peso | Valor | Σ (peso, valor) |
|--------|---|------|-------|-----------------|
| T1 | Sistema distribuido | 2 | 3 | 6 |
| T2 | Objetivos de performance o tiempo de respuesta | 1 | 5 | 5 |
| T3 | Eficiencia del usuario final | 1 | 5 | 5 |
| T4 | Procesamiento interno complejo | 1 | 5 | 5 |
| T5 | El código debe ser reutilizable | 1 | 4 | 4 |
| T6 | Facilidad de instalación | 0.5 | 5 | 2,5 |
| T7 | Facilidad de uso | 0.5 | 5 | 2,5 |
| T8 | Portabilidad | 2 | 5 | 10 |
| T9 | Facilidad de cambio | 1 | 4 | 4 |
| T10 | Concurrencia | 1 | 4 | 4 |
| T11 | Incluye objetivos especiales de seguridad | 1 | 4 | 4 |
| T12 | Provee acceso directo a terceras partes | 1 | 0 | 0 |
| T13 | Se requieren facilidades especiales de entrenamiento a usuarios | 1 | 4 | 4 |

Tabla 5.5. Clasificación de Complejidad Técnica del Sistema.

$$FCT = 0.6 + 0.01 \times \sum (\text{Peso } i \times \text{Valor } i)$$

$$FCT = 0.6 + 0.01 \times \sum (6 + 5 + 5 + 5 + 4 + 2,5 + 2,5 + 10 + 4 + 4 + 4 + 0 + 4)$$

$$FCT = 1, 16$$

5.2.2.2 Calcular el Factor de Complejidad Ambiente (FA)

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el Cálculo del Factor de Ambiente. El cálculo del mismo es similar al cálculo del Factor de Complejidad Técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5 y se utiliza la fórmula:

$$FA = 1.4 - 0.03 \times \sum (\text{Peso } i \times \text{Valor } i)$$

Clasificación del Factor de Ambiente del Sistema

| Factor | Descripción | Peso | Valor | Σ (peso, valor) |
|---------------|--|-------------|--------------|------------------------|
| E1 | Familiaridad con el modelo de proyecto utilizado | 1.5 | 4 | 6 |
| E2 | Experiencia en la aplicación | 0.5 | 4 | 2 |
| E3 | Experiencia en orientación a objetos | 1 | 4 | 4 |
| E4 | Capacidad del analista líder | 0.5 | 4 | 2 |
| E5 | Motivación | 1 | 5 | 5 |
| E6 | Estabilidad de los requerimientos | 2 | 2 | 4 |
| E7 | Personal a tiempo compartido | -1 | 2 | -2 |
| E8 | Dificultad del lenguaje de programación | -1 | 3 | -3 |

Tabla 0.1. Clasificación del Factor de Ambiente del Sistema.

$$FA = 1.4 - 0.03 \times \sum (6 + 2 + 4 + 2 + 5 + 4 - 2 - 3)$$

$$FA = 0,86$$

5.2.2.3 Calcular los Puntos de Casos de Uso Ajustados

Una vez calculados los puntos de Casos de Usos sin Ajustar (PCU), los Factores de Complejidad Técnica (FCT) y los Factores de Ambiente (FA), se procede a calcular los puntos de Casos de Uso Ajustados (PCUA).

$$PCUA = PCU \times FCT \times FA$$

$$PCUA = 280 \times 1,16 \times 0,86$$

$$PCUA = 385,28$$

5.35.3 Cálculo del Esfuerzo de Desarrollo (E)

Una vez que se han obtenido los Puntos de Función sin Ajustar del Sistema a partir de los Casos de Uso, se puede estimar el esfuerzo por dos métodos diferentes.

“La estimación del coste y del esfuerzo del software nunca será una ciencia exacta. Son demasiadas las variables – humanas, técnicas, de entorno, políticas – que pueden afectar el coste final del software y al esfuerzo aplicado para desarrollarlo.”

Roger S. Pressman

5.3.1 Convertir los Puntos de Casos de Uso Ajustados a Esfuerzo de Desarrollo

$$E = PCUA \times FC$$

FC: Factor de Conversión

Analizando el factor ambiente se observa que por debajo del valor medio (3) para E1...E6 existen 2 valores, mientras que por encima para E7...E8 no existe ningún valor por lo que el total es 2. Por lo anterior se plantea que FC = 20 horas-hombre

$$E = PCUA \times FC$$

$$E = 385,28 \times 20$$

$$E = 7705,6 \text{ Horas/Hombre}$$

Este método proporciona una estimación del Esfuerzo contemplando solo la funcionalidad especificada en los CU. Para una estimación más completa es necesario adicionar, las estimaciones de Esfuerzo de las

demás actividades relacionadas con el desarrollo del software las cuales se muestran en la siguiente tabla:

| Tipo de actividad | Porcentaje que representa | Horas-Hombre |
|----------------------------------|---------------------------|--------------|
| Análisis | 10% | 19264 |
| Diseño | 20% | 3852,8 |
| Implementación | 40% | 7705,6 |
| Prueba | 15% | 2889,6 |
| Sobre Cargas (Otras actividades) | 15% | 2889,6 |
| Total | 100% | 19264 |

Tabla 0.2. Cálculo de Esfuerzo.

5.3.2 Cálculo de Tiempo

La Estimación del Proyecto de Software puede dejar de ser un oscuro arte para convertirse en una serie de pasos sistemáticos que proporcionen estimaciones con un grado de riesgo aceptable.

$TDES (total) = E (total) / CH (hombres)$ donde:

TDES: Tiempo de Desarrollo

E: Esfuerzo

CH: Cantidad de Hombres

El desarrollo de este proyecto es llevado a cabo por dos personas:

$$TDES = 19264 / 2$$

$$TDES = 9632 \text{ Horas}$$

5.3.3 Cálculo del Costo Total a partir del Esfuerzo en Horas - Hombres

$C (total) = E (total \text{ en HH}) \times CHH$ donde:

CHH: Costo por Hombre Horas

$$CHH = K \times THP$$

K: Coeficiente que tiene en cuenta los costos indirectos (1,5 y 2,0)

THP: Tarifa Horaria Promedio

El salario promedio de las personas que trabajan en el proyecto dividido entre 192 horas(horas laborables del mes)

Se asume que el salario estándar de un trabajador es \$100 mensuales.

$$E \text{ (Total en HH)} = 19264$$

$$THP = 100/192 = 0,5208$$

Se asume que el coeficiente (k) que tiene en cuenta los costos indirectos es de 2.0

$$C \text{ (total)} = E \text{ (total en HH)} \times K \times THP$$

$$C \text{ (total)} = 19264 \times 2.0 \times 0,5208$$

$$C \text{ (total)} = \$ 20065,38$$

Después de realizado todo el estudio de factibilidad se llegó a la conclusión de que el desarrollo del proyecto incurriría en:

| Cálculo de: | Valor |
|-----------------------------|-----------------------------|
| Esfuerzo | 19264 x Horas-Hombre |
| Tiempo de desarrollo | 9632 Horas |
| Cantidad de personas | 2 |
| Costo | \$ 20065,38 |
| Salario estándar | \$100 |

Tabla 0.3. Cálculos Efectuados.

5.4 Beneficios Tangibles e Intangibles

5.4.1 Beneficios Tangibles

El proyecto ConDor, en particular el Módulo Inventario es un producto con fines comerciales, es un producto donde los requerimientos que lo motivan son bastante generales y aplicables a cualquier empresa. Por esta razón tiene como beneficios tangibles:

- La existencia de una aplicación Web flexible, dinámica y amigable para el control de inventario, ya que actualmente se cuenta con una aplicación de escritorio, monousuario e independiente para cada módulo.
- Ganancia económica. Esta aplicación cuenta con políticas de seguridad que aseguran una mayor protección de los bienes, ahorro de tiempo, costes, trayendo consigo ganancia.
- Disminución del uso de papel tangible y tinta para la confección de reportes y modelos contables. Aunque los reportes, comprobantes y otros informes se deben registrar y entregar en papel tangible, las cuentas y registros internos se realizan de forma digital, ahorrando materiales.
- Posibilita un mayor control de los productos en el almacén, disminuye el desvío de los mismos. Como el sistema debe de estar actualizado constantemente y será chequeado por personal de alto rango, el inventario debe de estar cuadrado, aspecto este que disminuye el desvío de materiales a personal no autorizado.

5.4.2 Beneficios Intangibles

La puesta en práctica del Módulo trae consigo los siguientes beneficios intangibles:

- Facilidad para los Contadores, Especialistas en Inventario y Almaceneros en el control de sus medios de forma digital, forma esta mucho más cómoda.
- Rápida ubicación de los medios dentro de la empresa donde se aplique el módulo. Para conocer los productos y sus cantidades almacenados, no es necesario recorrer cada uno de los almacenes, solamente consultar el sistema.
- Brindar información de los productos de los almacenes de la empresa sin tener que realizar inventario físico.

- Ahorro de tiempo para conocer los detalles de los productos almacenados.
- Ahorro de tiempo para realizar el control de inventario de la empresa.
- Ahorro de tiempo para solicitar un movimiento de medios entre áreas.
- Uso de varios usuarios del servicio instantáneo, lo que ahorra tiempo de trabajo. Como la aplicación es multiusuarios, varios trabajadores de la empresa responsables del control de inventario pueden acceder al sistema en el mismo momento, aspecto este que agiliza el proceso de control.
- Fácil uso de la aplicación por el personal y el proceso de instalación en la empresa. Para la instalación solo se necesita de un servidor, los demás usuarios accederán sencillamente mediante un navegador de Internet.
- Se logra una seguridad y protección de los datos consecuente con el nivel de seguridad requerido.
- Se logra eliminar el trabajo automatizado insuficiente y de forma manual en el control de inventario, lo cual se traduce en una disminución notable a las respuestas ante una solicitud y una mayor confiabilidad en la información obtenida.
- Y se creará una nueva y rápida vía de comunicación entre todas las oficinas y almacenes, así como de actualización y búsqueda de información.

5.5 Análisis de Costos y Beneficios

El costo estimado del producto es aproximadamente de \$ 20065, teniendo en cuenta que los diseñadores y futuros programadores son estudiantes y su salario es solamente de \$ 100, siendo este un buen costo teniendo en cuenta que es una aplicación bastante grande y compleja. Además, es un producto realizado para la venta, detalle este a considerar teniendo en cuenta que traerá más ingresos que gastos. Suponiendo que sea implementada por dos estudiantes se estima que el tiempo de desarrollo sea aproximadamente de 13 meses, siendo este un tiempo aceptable para desarrollarse por tan pequeño número de personas, cumpliendo todos los roles de desarrollo de un proyecto y pasando por cada una de las fases de la vida del software.

Teniendo en cuenta todo lo planteado anteriormente y analizando los beneficios que este trae consigo, se puede decir que el producto multimedia es factible.

5.6 Conclusiones parciales

Para el desarrollo de un buen producto de software además de consistir en comprender el ámbito del trabajo a realizar, los recursos que se requieren, y las tareas específicas a ejecutar, es necesario tener en cuenta su costo y factibilidad de realización pues esto puede influir grandemente en la aceptación y desarrollo del sistema deseado. Sin lugar a dudas una óptima planificación del proyecto contribuye a un desarrollo organizado y por ende a la calidad del producto obtenido.

Para el estudio de factibilidad del producto se utilizó la estimación por casos de uso. Luego de los cálculos realizados se analizó la viabilidad del producto, concluyendo que su implementación es factible ya que los beneficios económicos que brinda son considerables. De esta forma se da cumplimiento a una de las tareas propuestas y el objetivo del capítulo.

CONCLUSIONES

Refiriéndose al concluyente trabajo se puede decir que se han cumplido exitosamente los objetivos trazados, modelándose una aplicación Web a la altura de lo que se quería y se necesitaba, cumpliendo estrictamente con los requisitos solicitados por el cliente y con las mejoras y ventajas propuestas; capaz de realizar el proceso de control de inventario a cualquier entidad u organismo.

Para llevar a cabo el modelamiento de la aplicación se realizó el análisis de algunas soluciones existentes en Cuba y el mundo con características similares al sistema propuesto que se maneja para el control de inventario empresarial y se utilizó como base la versión del sistema ConDor existente, tomando ideas fundamentales para la confección de un producto que permitiera una mayor gama de ventajas y funcionalidades.

El sistema se diseñó siguiendo la metodología RUP y se utilizaron representaciones para la modelación de todas las fases del proyecto excepto para las de: Implementación y Prueba, las cuales no constituyen objetivos del trabajo. Para la futura implementación de la aplicación se determinó que se debe utilizar la plataforma .NET, dentro de ella la herramienta Visual Web Developer que es una herramienta “free” y es lo que exige el SICS y Visual Studio. NET 2005. Para el lenguaje de la programación se propone que sea C# por las grandes ventajas que esta brinda y la rapidez con que se puede programar. Para la base de datos se propone utilizar el gestor de base de datos SQL Server 2005 Express siendo también una herramienta libre y porque ofrece muy buena seguridad.

En el desarrollo del trabajo se realizó un análisis del modelamiento del negocio y del sistema propuesto delimitado por el alcance del trabajo. Seguidamente se realizó el análisis de la aplicación la cual facilitó la comprensión y modelamiento del diseño de la propuesta. El sistema resultante está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares del diseño y utiliza técnicas modernas de programación orientada a objetos.

A pesar del costo total y la demora en la realización del proyecto se considera factible su implementación por los beneficios posibles de alcanzar, que no es más que el mejoramiento del control de los medios existentes en las empresas cubanas.

RECOMENDACIONES

Con el objetivo de continuar, concluir y mejorar la aplicación propuesta se recomienda:

- Extender la investigación en los procesos de Facturación y Herramientas en Uso, las cuales no se tocaron en la investigación debido al alcance estimado.
- Realizar la implementación de la aplicación, para la solución real de los problemas existentes en el módulo estudiado.
- Continuar el estudio del tema con el objetivo de encontrar nuevas funcionalidades en el proceso del control de inventarios.
- Llevar la aplicación a otras plataformas libres, por ejemplo LINUX.

REFERENCIAS BIBLIOGRÁFICAS

CAMACHO, P. and R. SANCHEZ. *Módulo de Gestión para la Residencia de la facultad tres en la Universidad de las Ciencias Informáticas.* La Habana, CUJAE, 2006. 126. p.

CNPC.CULT.CU. *Sistema de Inventario Cubano del Patrimonio Cultural y Nacional,* [Web]. [Citado el 4/12/2006]. 2006. [Disponible en: <http://www.cnpc.cult.cu/sip/SIP.htm>]

COLECTIVO_DE_AUTORES. *Conferencia 1. Introducción a la Ingeniería de Software,* [word]. Universidad de las Ciencias Informáticas, 2005-2006.

COLECTIVO_AUTORES. *Patrones de Diseño. Ingeniería de software II. Conferencia 1.* UCI, 2006-2007.

CORPORATION, M. *Así es Microsoft Visual Studio.NET.* España, 2001.

DESARROLLOWEB.COM. *Ventajas de .Net,* [Web]. [Citado el 26/11/2006]. 2006. [Disponible en: <http://www.desarrolloweb.com/articulos/1329.php>]

DEPARTAMENTO_INGENIERÍA_SOFTWARE. *Flujo de Trabajo de Modelación del Negocio. Clase Práctica 2. Ingeniería y gestión de software 1.* UCI, 2005-2006.

FAHNLE, P. *¿Qué es ASP.Net?,* [Web]. [Citado el 26/11/2006]. 2006. [Disponible en: <http://www.webexperto.com/articulos/art/180/que-es-aspnet/>]

JACOBSON, I.; G. BOOCH, et al. *El proceso Unificado de Desarrollo de Software.* Madrid, Eddison Wesley, 2004. 435 p. 0-201-57169-2

MARTIN, E. L. *Un efectivo sistema cubano de contabilidad,* [Web]. [Citado el 19/03/2007]. 2005. [Disponible en:

<http://www.eleconomista.cubaweb.cu/2005/edicionimpresa/premioeconomiamiquel.html>

MONOGRAFIAS.COM. *Control de Inventario*, [Web] [Citado el 19/03/2007]. 1997. [Disponible en: <http://www.monografias.com/trabajos11/conin/conin.shtml>]

MORAN, B. *Análisis comparativo entre Microsoft® .NET y Sun® J2EE*, [Web]. [Citado el 22/11/2006]. 2006. [Disponible en: <http://www.ciberteca.net/articulos/programacion/net>].

MSDN. *Microsoft SQL Server 2005 Express*. [Web]. [Citado el 14/01/2007]. 2005a. [Disponible en: <http://www.microsoft.com/spanish/msdn/vstudio/express/SQL/default.msp>]

---. *Visual Studio 2005*, [Web]. [Citado el 14/01/2007]. 2005b. [Disponible en: <http://www.microsoft.com/emea/msdn/visualstudio/eses/getthetrials/>]

MSDN. *Requisitos de hardware y software para instalar SQL Server 2005* [Web]. [Citado el 24/03/2007] 2005. Disponible en: <http://msdn2.microsoft.com/es-es/library/ms143506.aspx>

PERAL, L. E. *Subsistema de Gestión de matrícula*. La Habana, UCI, 2004. 255. p.

REYES, G. C. *Tecnología de la programación!!!*, [Web]. [Citado el 20/11/2006]. 2006. [Disponible en: <http://www.elquille.info/colabora/NET2005/gcuadra>WeDevCR2005.htm>]

RUIZ, R. G. *"Control de Inventario", "Facturación" y "Útiles y Herramientas en Uso"*. La Habana, 2005. 18 p.

SANCHEZ, M. M. *Metodologías De Desarrollo De Software*, [Web]. [Citado el 5/11/2006]. 2006. [Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software07062004.html]

WEBANDMACROS.COM. *Web Dinámica (Definición, Ejemplos, Aplicaciones Web...)*, [Web].
[Citado el 20/11/2006]. 2006a. [Disponible en: <http://www.webandmacros.com/webdinamica.htm>]

WIKIPEDIA.ORG. *Cliente-servidor*, [Web]. [Citado el 20/11/2006]. 2006a. [Disponible en:
<http://es.wikipedia.org/wiki/Cliente-servidor>]

---. *Lenguaje de modelado de objetos*, [Web]. [Citado el 18/11/2006]. 2006b. [Disponible en:
http://es.wikipedia.org/wiki/Lenguaje_de_modelado_de_objetos]

BIBLIOGRAFÍA

Bergantiños Hidalgo, Damian. Aplicaciones de Internet. Aplicaciones del Cliente (**Parte I**). **Revista Giga # 3. 2002.**

Bergantiños Hidalgo, Damian. Aplicaciones de Internet. Aplicaciones del Servidor (**Parte II**). **Revista Giga # 4. 2002.**

De Miguel Castellano, Adoración y Piattini Velthuis, Mario G. Fundamentos y Modelos de las Bases de Datos. **Universidad Carlos III de Madrid.** Ciudad de La Habana, 2003. 2da edición.

Jacobson, I.; Booch, G. y Rumbaugh, J.; El Proceso Unificado de Desarrollo de Software. **La Habana, 2004. Volumen I y II**

Larman Craig. UML y Patrones. Introducción al análisis y diseño orientados a objetos. **México, 1999.**

Ledesma Chirino, Andrés. CASES. **Revista Giga. #4. 1997.**

MacDonald, Mathew. ASP.NET. **Manual de referencia. Madrid, 2002.**

Núñez Camallea, Noel L. Gestión de Bases de Datos con ADO.NET. **Ciudad de La Habana, 2004.**

Rozic, Sergio Ezequiel. Bases de Datos y su aplicación con SQL. **Buenos Aires, 2004.**

S. Pressman, Roger. Ingeniería del software. Un enfoque práctico. **La Habana, 2005. Parte 1 y 2**

Valero, Vicent Antón. La Contabilidad Nacional como marco de las estimaciones de variables económicas. Instituto **Vasco de Estadísticas. Euskadi, 1991. Cuaderno 23.**

GLOSARIO DE TÉRMINOS

A

ADO.NET: Clases que brindan funcionalidad de conectividad y de recuperación de los datos, que permite crear programas para el manejo de la base de datos. Es parte de un conjunto de trabajo de .NET.

Aplicación Web: Sitio Web que contiene páginas con contenido sin determinar parcialmente o en su totalidad. El contenido final de estas páginas se determina sólo cuando un visitante solicita una página del servidor Web.

ASP: Active Server Page, tecnología creada por Microsoft destinada a la creación de sitios Web, marco sobre el cual se pueden construir aplicaciones basadas en Internet.

B

Base de Datos: Conjunto de datos almacenados en un soporte informático no volátil. En ella los datos se encuentran interrelacionados y estructurados de acuerdo con su modelo capaz de recoger el máximo de contenido. Se organizan en tablas. Cada fila de una tabla constituye un registro de datos, y cada columna constituye un campo del registro. Constituyen una parte integrante y fundamental del sistema de información y tiene su razón de ser en la misma existencia de esta.

C

Clase de Interfaz: Modelan la interacción entre el sistema y sus actores (usuarios y sistemas externos), reciben y presentan información y peticiones de y hacia los actores, reúnen los requisitos en los límites del sistema, suelen ser abstracciones de ventanas, formularios, interfaces de impresoras, sensores, terminales, y describen lo que se obtiene con la interacción, no el proceso físico de cómo se ejecuta.

Clase de Control: Representan coordinación, secuencia, transacciones, y control de otros objetos, se usan mucho para encapsular el control de un caso de uso concreto, También derivaciones y cálculos complejos, manejan y coordinan las acciones y los flujos de control principales, y delegan trabajo a otros objetos (de interfaz y de entidad),

Clase de Entidad: Modelan información que posee una larga vida, modelan la información y el comportamiento asociado a algún fenómeno o concepto, como una persona, un objeto del mundo real, o

un suceso del mundo real, derivan normalmente de una clase de entidad del negocio, puede tener comportamiento complejo, aísla los cambios en la información que representa.

CLR: Common Language Runtime. Es el núcleo de la plataforma .NET. Es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad. Es el entorno de ejecución que traducirá el código intermedio CIL a código máquina y por tanto permitirá ejecutar cualquier aplicación de la plataforma.

E

FCL: (Framework Class Library): Librería de clases que proporciona una gran cantidad de servicios en cualquier lenguaje que esté dentro de la plataforma .NET: Entrada/Salida, XML, ADO.NET (acceso a Bases de datos), Windows.Forms (aplicaciones gráficas), sockets, colecciones entre otras. A partir de estas clases prefabricadas el programador puede crear nuevas clases que mediante herencia extiendan su funcionalidad y se integren a la perfección con el resto de clases. A través de las clases suministradas en ella es posible desarrollar cualquier tipo de aplicación, desde las tradicionales aplicaciones de ventanas, consola o servicio de Windows NT hasta los novedosos servicios Web y páginas ASP.NET.

H

HTML: “Hyper Text Markup Language”, lenguaje de marcas, con que se programan las páginas Web. Brinda facilidades para mostrar imágenes, textos hipervínculos, tablas, etc., y es interpretado por los navegadores Web.

I

Interfaz: Una colección de operaciones que se usan para especificar el servicio de una clase o de un componente. Un juego nombrado de operaciones que caracterizan la conducta de un elemento. La Interfaz hombre-máquina es un canal comunicativo entre el usuario y el ordenador.

L

Lenguaje de programación: Conjunto de normas «lingüísticas» que permiten escribir un programa y que éste sea entendido por el ordenador y pueda ser trasladado a ordenadores similares para su

funcionamiento en otros sistemas. Cualquier lenguaje artificial que puede utilizarse para definir una secuencia de instrucciones para su procesamiento por un ordenador o computadora.

P

Página Web: Documento en formato ASCII que puede ser visualizado en distintos navegadores Web sin importar la plataforma. Este documento en su forma más sencilla permite presentar información textual, gráficos, tablas, etc., y permite el uso de enlaces o hipervínculos para saltar a otras páginas o referenciar a otros archivos.

S

Servidor Web: Software que suministra páginas Web en respuesta a las peticiones de los navegadores Web.

SGDB: Conjunto de programas de propósito general que permite a los usuarios controlar el acceso y utilización de la Base de Datos, para incluir, modificar o recuperar información, incluyendo prestaciones con el fin de conseguir independencia, integridad y seguridad de los datos y la concurrencia de los usuarios.

Sistema: Conjunto de cosas que ordenadamente relacionados entre sí constituyen un determinado objeto.

T

TCP/IP: Se refieren a dos protocolos de red: Transmission Control Protocol (Protocolo de Control de Transmisión) e Internet Protocol (Protocolo de Internet). Los diferentes protocolos de la suite TCP/IP trabajan conjuntamente para proporcionar el transporte de datos dentro de Internet (o Intranet). En otras palabras, hacen posible que accedamos a los distintos servicios de la Red.

U

UML: “Unified Modeling Language” Lenguaje gráfico que brinda un vocabulario y reglas para especificar, construir, visualizar y documentar los artefactos de un sistema utilizando el enfoque orientado a objetos.

ANEXOS

Anexo 1 Modelo de Objetos.

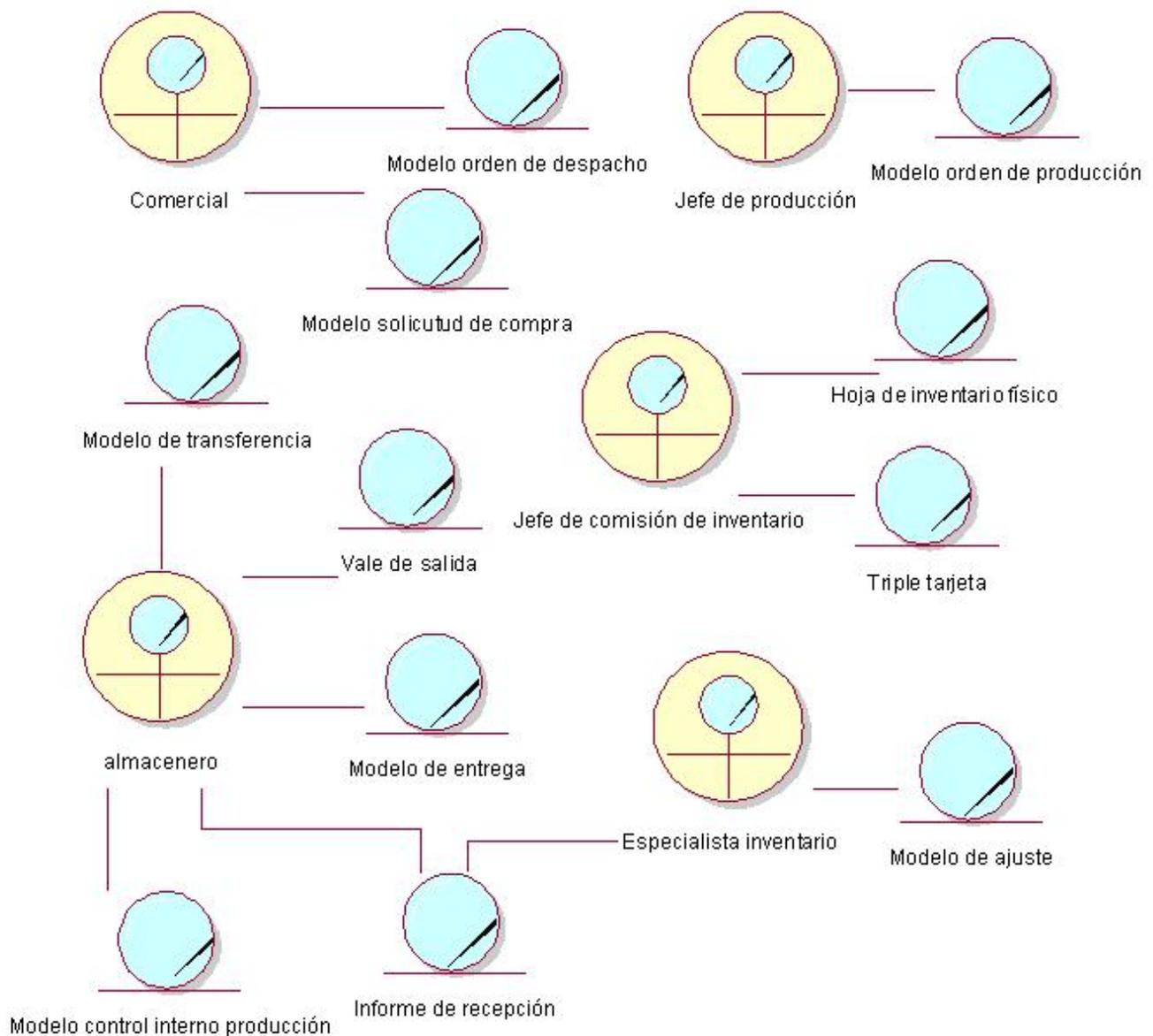


Fig. A1. Modelo de Objetos

Anexo 2 Diagramas de CU por subsistemas.

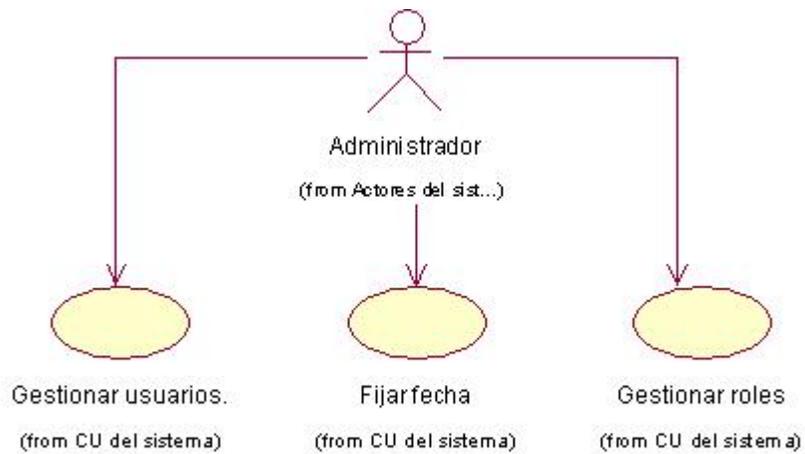


Fig. A2. 1. Diagrama de caso de Uso: Subsistema Administración.

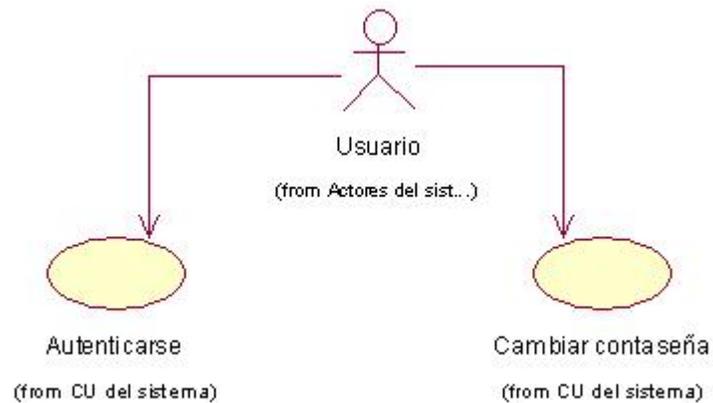


Fig. A2.2. Diagrama de Caso de Usos: Subsistema Servicios Comunes.

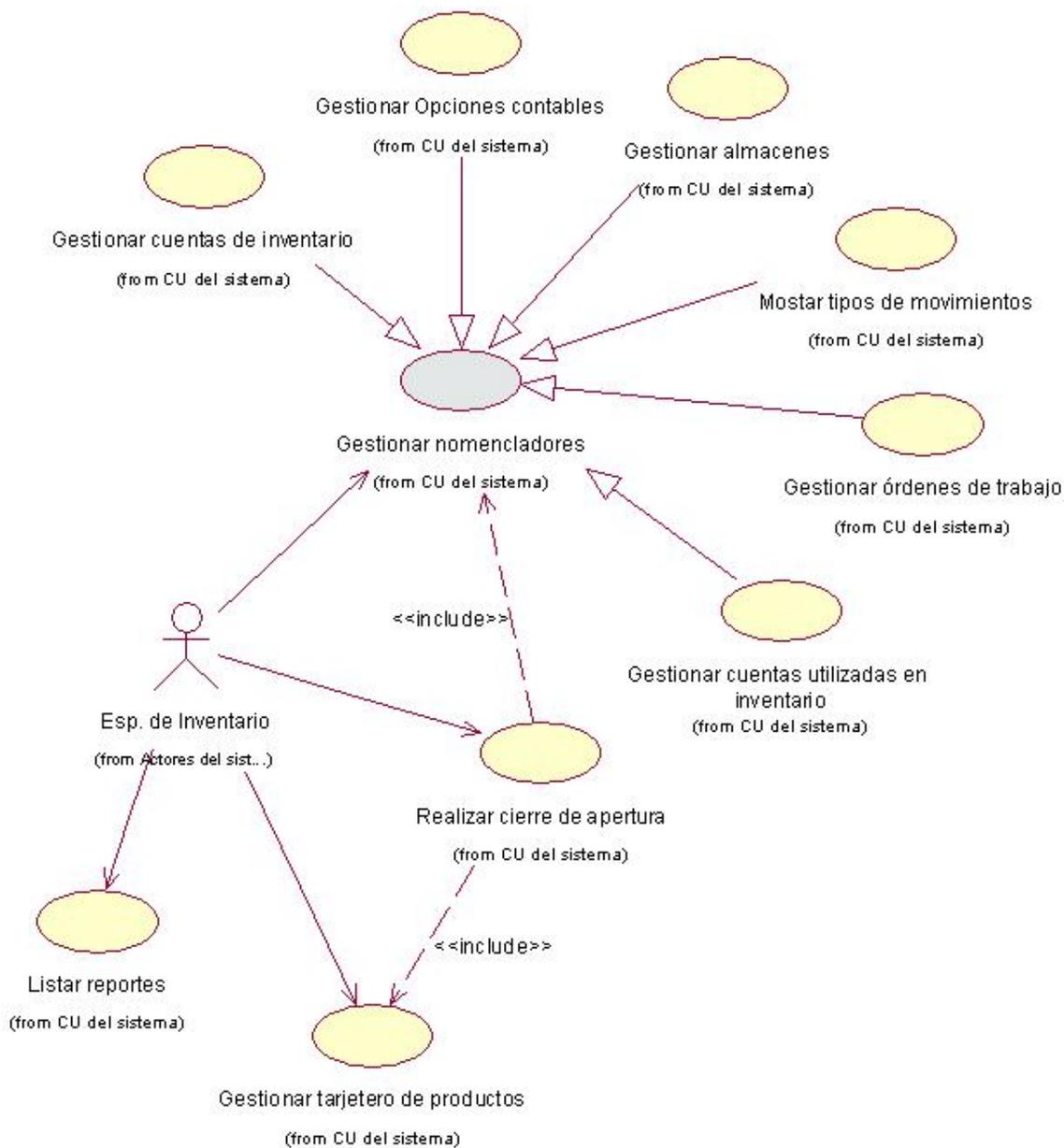


Fig. A2.3. Diagrama de Caso de Usos: Subsistema Fase de Apertura.

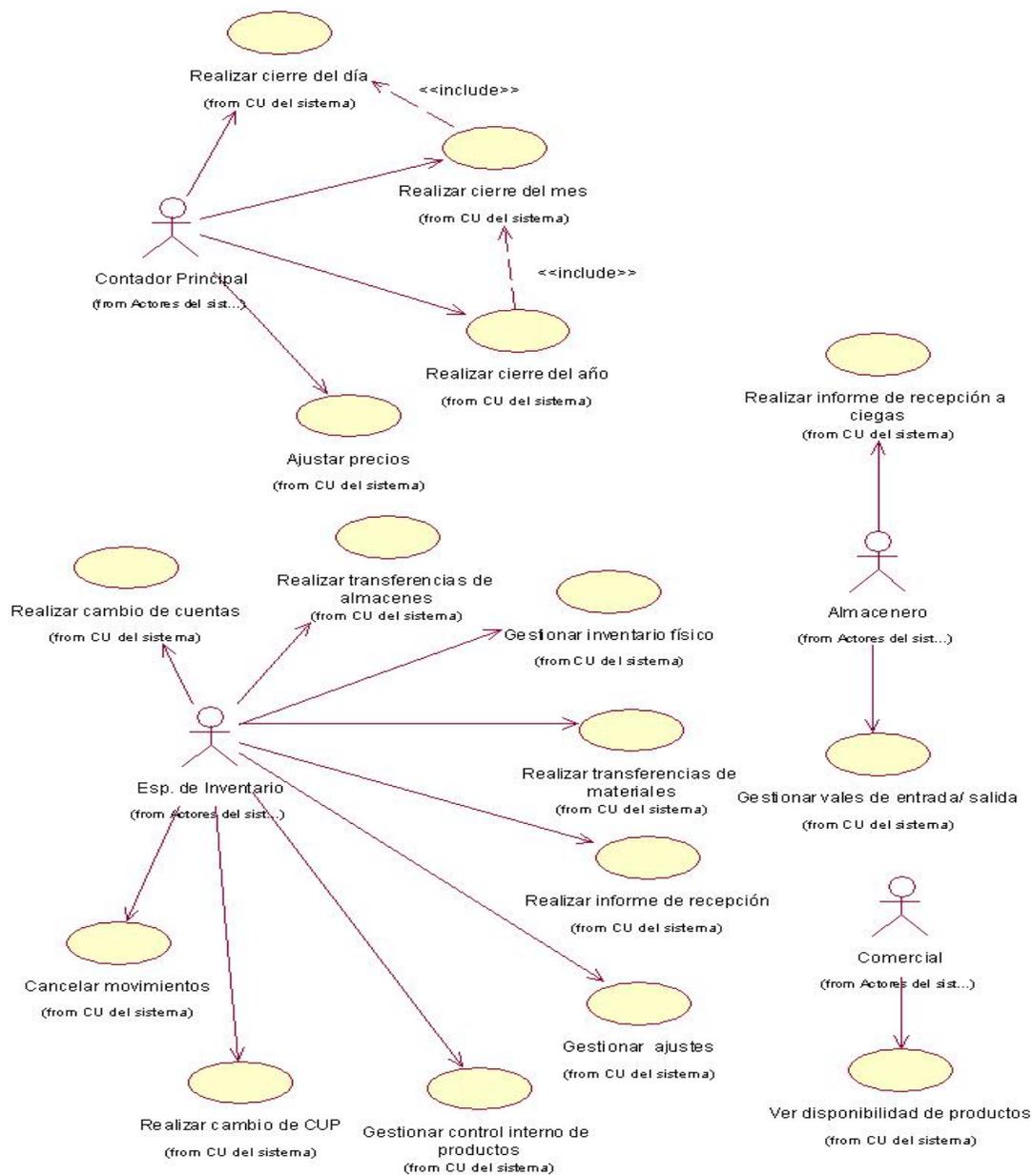


Fig. A2.4. Diagrama de Caso de Usos: Subsistema Fase de Explotación.

Anexo 3 Descripción de los casos de uso críticos del sistema.

| Caso de Uso | |
|---|--|
| 01 | Gestionar Almacenes |
| Propósito | Registrar todos almacenes pertenecientes a la entidad. |
| Actor | Especialista de inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario registra todos y cada uno de los almacenes de la entidad. El Especialista en Inventario puede insertar, eliminar, modificar y mostrar datos de los almacenes. |
| Referencias | R5. |
| Precondiciones | En caso de eliminar o modificar el almacén tiene que estar creado. |
| Curso Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario. selecciona que desea hacer con el almacén: | 1.1- a) Si desea insertar un nuevo almacén, ver sección: Insertar almacén. b) Si desea modificar un almacén, ver sección: Modificar almacén. c) Si desea eliminar un almacén, ver sección: Eliminar almacén. d) Si desea Mostrar los almacenes existentes, ver sección: Mostrar almacenes. |
| Sección "Insertar almacén" | |
| Acción del Actor | Respuesta del Sistema |
| 2. El Especialista en Inventario selecciona la opción "insertar" marcando el nomenclador almacenes. | 2.1. El sistema muestra los formularios correspondientes. |

| | |
|--|--|
| 3. El Especialista en Inventario entra los datos del un nuevo almacén y acepta. | 3.1. El sistema verifica los datos. 3.2 Si son correctos o no existe un almacén con el mismo número lo inserta en la base de datos. |
| Sección “Modificar almacén” | |
| Acción del Actor | Respuesta del Sistema |
| 4. El Especialista en Inventario selecciona la opción de modificar almacén. | 4.1. El sistema muestra todos los almacenes registrados. |
| 5. El Especialista en Inventario selecciona el almacén al que desea modificarle sus características. | 5.1. El sistema presenta las características registradas del almacén. |
| 6. El Especialista en Inventario especifica las características que desea cambiar del almacén. | 6.1. El sistema valida los datos. 6.2. Si son correctos salva los cambios al almacén. |
| Sección “Eliminar almacén” | |
| Acción del Actor | Respuesta del Sistema |
| 7. El Especialista en Inventario selecciona la opción de eliminar almacén. | 7.1 El sistema muestra todos los almacenes registrados |
| 8. El Especialista en Inventario selecciona el almacén que desea eliminar. | 8.1 El sistema pide confirmación de eliminación. |
| 9. El Especialista en Inventario confirma la eliminación del almacén. | 9.1. El sistema elimina almacén. |
| Sección “Mostrar almacenes” | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario selecciona “Mostrar almacenes”. | 2. El sistema muestra lista de los almacenes registrados de la entidad. |
| Curso alternativo de eventos | |
| Sección “Insertar” | |
| 3.2. Si los datos no son correctos muestra un mensaje “Los datos entrados no son válidos”, o si ya | |

existe un almacén con ese número el sistema muestra un mensaje “Ya existe este almacén”,
6.2 Si los datos no son correctos muestra un mensaje “Los datos entrados no son válidos”

| | |
|------------------|----------|
| Prioridad | Crítico. |
|------------------|----------|

| Caso de Uso | |
|--|---|
| 02 | Gestionar Órdenes de Trabajo |
| Propósito | Controlar las órdenes de trabajo solicitadas. |
| Actor | Especialista en Inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario introduce, modifica, elimina y/o muestra las órdenes de trabajo solicitadas al almacén. |
| Referencias | R9. |
| Precondiciones | En caso de eliminar y modificar la orden, debe primeramente existir. |
| Curso Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario. selecciona que desea hacer con la orden: | 1.1- a) Si desea insertar una nueva orden, ver sección: Insertar orden. b) Si desea modificar una orden, ver sección: Modificar orden. c) Si desea eliminar una orden, ver sección: Eliminar orden. d) Si desea Mostrar las órdenes existentes, ver sección: Mostrar orden. |
| Sección “Insertar orden” | |
| Acción del Actor | Respuesta del Sistema |
| 2. El Especialista en Inventario solicita insertar una nueva orden. | 2.1. El sistema muestra los formularios correspondientes. |

| | |
|--|---|
| 3. El Especialista en Inventario inserta los nuevos datos. | 3.1. El sistema verifica los datos. 3.2 Si son correctos o no existe una orden con el mismo número lo inserta en la base de datos. |
| Sección “Modificar orden” | |
| Acción del Actor | Respuesta del Sistema |
| 4. El Especialista en Inventario solicita modificar una orden | 4.1 El sistema muestra todas las órdenes registradas |
| 5. El Especialista en Inventario selecciona la orden que desea modificar. | 5.1. El sistema presenta las características registradas de la orden. |
| 6. El Especialista en Inventario especifica las características que desea cambiar. | 6.1. El sistema valida los datos. 6.2. Si son correctos salva los cambios a la orden. |
| Sección “Eliminar orden” | |
| Acción del Actor | Respuesta del Sistema |
| 7. El Especialista en Inventario solicita Eliminar una orden | 7.1 El sistema muestra todas las órdenes registradas |
| 8. El Especialista en Inventario selecciona la orden que desea eliminar. | 8.1. El sistema pide confirmación de eliminación. |
| 9. El Contador confirma la eliminación de la orden. | 9.1. El sistema elimina orden de trabajo. |
| Sección “Mostrar orden” | |
| Acción del Actor | Respuesta del Sistema |
| 10. El Especialista en Inventario selecciona mostrar órdenes de trabajo. | 10.1. El sistema muestra todas las órdenes de trabajo. |
| Curso alternativo de los eventos | |
| Sección “Insertar” | |
| 3.2. Si los datos no son correctos muestra un mensaje “Los datos entrados no son válidos”, o si ya existe una orden con ese número el sistema muestra un mensaje “Ya existe este almacén”, 6.2 Si los datos no son correctos muestra un mensaje “Los datos entrados no son válidos” | |

| | |
|------------------|----------|
| Prioridad | Crítico. |
|------------------|----------|

| | |
|---|--|
| Caso de Uso | |
| 03 | Gestionar Cuenta de Inventario |
| Propósito | Controlar las cuentas de control de inventario y modificarlas. |
| Actor | Especialista en Inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario solicita parametrizar las cuentas de control de inventario. |
| Referencias | R11. |
| Precondiciones | |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario. selecciona que desea hacer con la cuenta de inventario: | 1.1- a) Si desea modificar una cuenta, ver sección: Modificar cuenta. b) Si desea Mostrar las órdenes existentes, ver sección: Mostrar cuenta. |
| Sección "Modificar cuenta" | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario solicita parametrizar cuentas. | 2. El sistema muestra las los campos existentes. |
| 3. El Especialista en Inventario llena los formularios a modificar. | 4. El sistema valida los datos y actualiza los datos en el sistema. |
| Sección "Mostrar cuentas" | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario solicita mostrar cuentas de inventario. | 2. El sistema muestra las cuentas existentes. |
| Prioridad | Crítico. |

| Caso de Uso | |
|--|--|
| 04 | Gestionar Tarjetero de Productos |
| Propósito | Insertar y controlar todos los productos del almacén. |
| Actor | Especialista en Inventario |
| Resumen | El caso de uso se inicia cuando el almacenero llena todos los formularios del tarjetero, llenando todos los campos. El Especialista en Inventario debe mantener actualizado el tarjetero. |
| Referencias | R14. |
| Precondiciones | El almacenero solo puede insertar productos en el tarjetero. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario selecciona que desea hacer con los productos del tarjetero> | 1.1- a) Si desea insertar un nuevo producto, ver sección: Insertar producto. b) Si desea modificar un producto, ver sección: Modificar producto. c) Si desea Mostar el tarjetero, ver sección: Mostrar tarjetero. |
| Sección “Insertar producto” | |
| Acción del Actor | Respuesta del Sistema |
| 2. El Almacenero selecciona la opción de insertar productos. | 2.1. El sistema muestra los formularios correspondientes. |
| 3. El Almacenero llena los formularios. | 3.1. El sistema inserta el producto en el tarjetero. |
| Sección “Modificar ajuste” | |
| Acción del Actor | Respuesta del Sistema |
| 4. El Especialista en Inventario solicita modificar un producto | 4.1 El sistema muestra todas los productos registrados |
| 5. El Especialista en Inventario selecciona el producto al que desea modificarle sus | 5.1. El sistema presenta las características registradas del producto. |

| | |
|---|---|
| características en el tarjetero. | |
| 6. El Especialista en Inventario especifica las características que desea cambiar del producto. | 6.1. El sistema salva los cambios al producto. |
| Sección “Mostrar orden” | |
| Acción del Actor | Respuesta del Sistema |
| 7. El Especialista en Inventario selecciona “Mostrar Productos del Tarjetero”. | 7.1. El sistema muestra todos los productos existentes en el tarjetero. |
| Prioridad | Crítico. |

| | |
|--|---|
| Caso de Uso | |
| 05 | Hacer Cierres de Apertura |
| Propósito | Hacer cierres de apertura para luego pasar a la fase de explotación. |
| Actor | Especialista en Inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario solicita hacer cierre de apertura para pasar a una nueva fase de control de inventario. |
| Referencias | R15. |
| Precondiciones | Existencia de productos en el Tarjetero y de artículos en los nomencladores de Centros de Costo, Almacenes, Áreas, Clasificador de Cuentas, etc. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario selecciona la opción de cierre de apertura. | 2. El sistema valida que exista una fecha de apertura, lo que significa que se declaran las existencias y valores, que cada producto tenía al final del mes del último balance. |
| | 3. valida la existencia de los nomencladores. |
| | 4. El sistema a partir de los saldos del Tarjetero, salva los saldos por Cuentas-Subcuentas los que se |

| | |
|------------------|---|
| | continuarán actualizando posteriormente con los movimientos diarios en cada cierre contable que se realice. |
| Prioridad | Crítico. |

| Caso de Uso | |
|---|--|
| 06 | Gestionar Control Interno de la Producción |
| Propósito | Controlar todos los productos que son utilizados o creados en la producción. |
| Actor | Jefe de Brigada. |
| Resumen | El caso de uso se inicia cuando el Jefe de Brigada solicita hacer cualquier operación relacionada con la producción. El Jefe de Brigada puede insertar, modificar, eliminar y listar productos. |
| Referencias | R17. |
| Precondiciones | En caso de eliminar modificar, el producto debe existir. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario. selecciona que desea hacer con el control interno de la producción: | 1.1- a) Si desea insertar un nuevo producto, ver sección: Insertar producto. b) Si desea modificar un producto, ver sección: Modificar producto. c) Si desea eliminar un producto, ver sección: Eliminar producto. d) Si desea Mostrar los productos existentes, ver sección: Mostrar productos. |
| Sección “Insertar producto” | |
| Acción del Actor | Respuesta del Sistema |

| | |
|--|---|
| 2. El Jefe de Brigada solicita insertar un nuevo producto. | 2.1 El sistema muestra los formularios a llenar. |
| 3. El Jefe de Brigada inserta los datos de ese nuevo producto. | 3.1. El sistema verifica los datos. 3.2 Si son correctos o no existe una orden con el mismo número lo inserta en la base de datos. |
| Sección “Modificar producto” | |
| Acción del Actor | Respuesta del Sistema |
| 4. El Jefe de Brigada solicita modificar un producto. | 4.1 El sistema muestra los productos registrados. |
| 5. El Jefe de Brigada selecciona el producto al que desea modificar. | 5.1. El sistema presenta las características registradas del producto. |
| 6. El Jefe de Brigada especifica las características que desea cambiar del producto. | 6.1. El sistema salva los cambios al producto. |
| Sección “Eliminar producto” | |
| Acción del Actor | Respuesta del Sistema |
| 7. El Jefe de Brigada solicita eliminar un producto. | 7.1 El sistema muestra los productos registrados. |
| 8. El Jefe de Brigada selecciona el producto que desea eliminar. | 8.1. El sistema pide confirmación de eliminación. |
| 9. El usuario con permiso confirma la eliminación del producto. | 9.1. El sistema elimina producto. |
| Sección “Mostrar productos” | |
| Acción del Actor | Respuesta del Sistema |
| 10. El Jefe de Brigada selecciona el o los productos que desea mostrar. | 10.1. El sistema muestra productos. |
| Prioridad | Crítico. |

| Caso de Uso | |
|---|---|
| 07 | Gestionar Vales de Entrada/ Salida |
| Propósito | Controlar todas las acciones de Entrada/ Salida. Se realizaran operaciones como: <ul style="list-style-type: none"> - Entrega para Insumos. - Devolución de Insumos. - Entrega por Reestructuración. - Devolución de Compras. - Devolución de Ventas. - Devolución de Mercancía Entregada en Consignación. - Devolución de Mercancía Recibida en Consignación. - Insumo de Productos Recibidos en Consignación. |
| Actor | Almacenero. |
| Resumen | El caso de uso se inicia cuando el Almacenero inserta, elimina o muestra cualquiera de las operaciones que se pueden realizar en el caso de uso. |
| Referencias | R18. |
| Precondiciones | |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en inventario. selecciona que desea hacer con el vale de entrada / salida: | 1.1- <ul style="list-style-type: none"> a) Si desea insertar un nuevo vale, ver sección: Insertar vale. b) Si desea eliminar un vale, ver sección: Eliminar vale. c) Si desea Mostrar los vales existentes, ver sección: Mostrar vales. |
| Sección "Insertar vale" | |
| Acción del Actor | Respuesta del Sistema |
| 2. El Almacenero solicita insertar una operación al sistema. | 2.1 El sistema muestra los formularios correspondientes. |

| | |
|--|--|
| 3. El Almacenero introduce los datos de la operación. | 3.1. El sistema verifica si que todos los campos a llenar están completados y luego inserta. |
| Sección “Eliminar vale” | |
| Acción del Actor | Respuesta del Sistema |
| 4 El Almacenero solicita eliminar una operación al sistema. | 4.1 El sistema muestra los formularios correspondientes. |
| 5. El Almacenero selecciona el vale que desea eliminar. | 5.1. El sistema pide confirmación de eliminación. |
| 6. El Almacenero confirma la eliminación del vale. | 6.1. El sistema elimina el vale. |
| Sección “Mostrar vale” | |
| Acción del Actor | Respuesta del Sistema |
| 1. El almacenero solicita mostrar todas las veles realizados en el día, el mes o el año o que cumplan alguna característica en específico. | 2. El sistema muestra los vales. |
| Prioridad | Crítico. |

| | |
|-----------------------|---|
| Caso de Uso | |
| 08 | Realizar Informe de Recepción |
| Propósito | Validar los productos entrados al almacén recibidos por el Almacenero. |
| Actor | Especialista en Inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario llena todos los formularios correspondiente al informe, completando solamente los campos que le son activados por el rol que desempeña. |
| Referencias | R19. |
| Precondiciones | |

| Flujo Normal de Eventos | |
|---|---|
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario solicita realizar un informe de reopción | 1.1. El sistema muestra los formularios correspondientes. |
| 2. El Especialista en Inventario llena todos los y acepta. | 2.1 El sistema valida los datos. 2.2 El sistema salva los datos. |
| | 3.1 El sistema muestra el reporte de la operación |
| Prioridad | Crítico. |

| Caso de Uso | |
|---|---|
| 09 | Realizar Inventario Físico |
| Propósito | Controlar los productos existentes en el almacén y compararlo con los datos que se encuentran en factura. |
| Actor | Especialista en Inventario. |
| Resumen | El caso de uso se inicia cuando el Especialista en Inventario registra el inventario físico, lo cual puede seleccionar por varias opciones. |
| Referencias | R23. |
| Precondiciones | Tienen que estar creados los nomencladores. |
| Flujo Normal de Eventos | |
| Sección "" | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Especialista en Inventario solicita registrar inventario. | 2. El sistema muestra una lista con todos los inventarios realizados anteriormente y varias opciones para que le Esp. de inventario seleccione como lo quiere realizar. |
| 3. Si el Especialista en Inventario decide y llena formularios. | 4. El sistema salva los cambios. |
| Prioridad | Crítico. |

| Caso de Uso | |
|---|---|
| 10 | Cierre del Día |
| Propósito | Cerrar el día. |
| Actor | Contador Principal. |
| Resumen | El caso de uso se inicia cuando el Contador Principal solicita cerrar el día. |
| Referencias | R28. |
| Precondiciones | Deben estar creados los Nomencladores y estado hecho el Cierre de Apertura. |
| Flujo Normal de Eventos | |
| Acción del Actor | Respuesta del Sistema |
| 1. El Contador selecciona realizar cierre del día | 1.1. El sistema muestra los formularios correspondientes. |
| 2. El Contador Principal selecciona los movimientos listos para cerrar. | 2.1 El sistema calcula y limpia la información de Entradas-Salidas del último día cerrado a fin de volcar la nueva información, todo esto en ambas monedas. |
| | 3. El sistema va calculando a partir de los modelos que tiene en proceso los valores en Entradas o Salidas y adicionándolo en las cuentas que correspondan para cada almacén. También hace la operación en el acumulado del mes, pero en este caso sin limpiar Entradas ni Salidas. |
| | 4. El sistema presenta dos Reportes: "Cierre Diario" y "Cuadre Diario" para su impresión. A partir de aquí puede cerrar o no según desee el usuario. |
| Prioridad | Crítico. |

| Caso de Uso | |
|------------------|-----------------------|
| 11 | Cierre del Mes |
| Propósito | Cerrar el mes. |

| | | |
|--|--|--|
| Actor | Contador Principal. | |
| Resumen | El caso de uso se inicia cuando el Contador Principal solicita cerrar el mes. | |
| Referencias | R29. | |
| Precondiciones | Debe estar cerrado el día. | |
| Flujo Normal de Eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El Contador Principal solicita hacer cierre de mes. | 1.1 El sistema muestra el formulario correspondiente. | |
| 2. El Contador llena el formulario | 2. El sistema valida los datos. | |
| | 3. El sistema valida que se haya hecho el Cierre del Día y que no hayan movimientos pendientes de Ajuste. | |
| | 4. Borra la información referente a todos los comprobantes del mes y hace los cálculos necesarios para prepararla con la fecha del nuevo mes y los saldos iniciales tanto del día como acumulado del mes en ambas monedas. | |
| | 5. Borra los campos de Entradas y Salidas tanto en el Acumulado del Día como el Acumulado del Mes. | |
| Prioridad | Crítico. | |

| | | |
|--------------------------------|---|--|
| Caso de Uso | | |
| 12 | Cierre del Año | |
| Propósito | Cerrar el año. | |
| Actor | Contador Principal. | |
| Resumen | El caso de uso se inicia cuando el Contador Principal solicita cerrar el año. | |
| Referencias | R30. | |
| Precondiciones | Debe estar cerrado el mes. | |
| Flujo Normal de Eventos | | |

| Acción del Actor | Respuesta del Sistema |
|--|--|
| 1. El Contador solicita hacer cierre de año. | 1.1 El sistema muestra los formularios correspondientes. |
| 2. El Contador llena los formularios | 2.1 El sistema valida los datos. |
| | 3. El sistema valida que se haya realizado el Cierre del Mes y proporciona Listados de los productos que no se hayan chequeado y de los que no hayan tenido movimientos en el año. |
| | 4. Posibilita borrar del Tarjetero todos los productos cuya existencia=0.00 y pasa los Movimientos del Año para un Histórico (no está en el módulo actual). |
| | 5. Limpia las tablas de Movimientos y pone en cero el campo en el fichero de consecutivos para los No. de Documentos automáticos por Almacén-Documento y borra las prefacturas de más de 60 días de creadas. |
| | 6. Cambia la fecha para el 1ro. / Enero del siguiente año. |
| Prioridad | Crítico. |

Tabla A3. Descripción de los Casos de Usos del Sistema.

Anexo 4: Sistema representado en Paquetes.

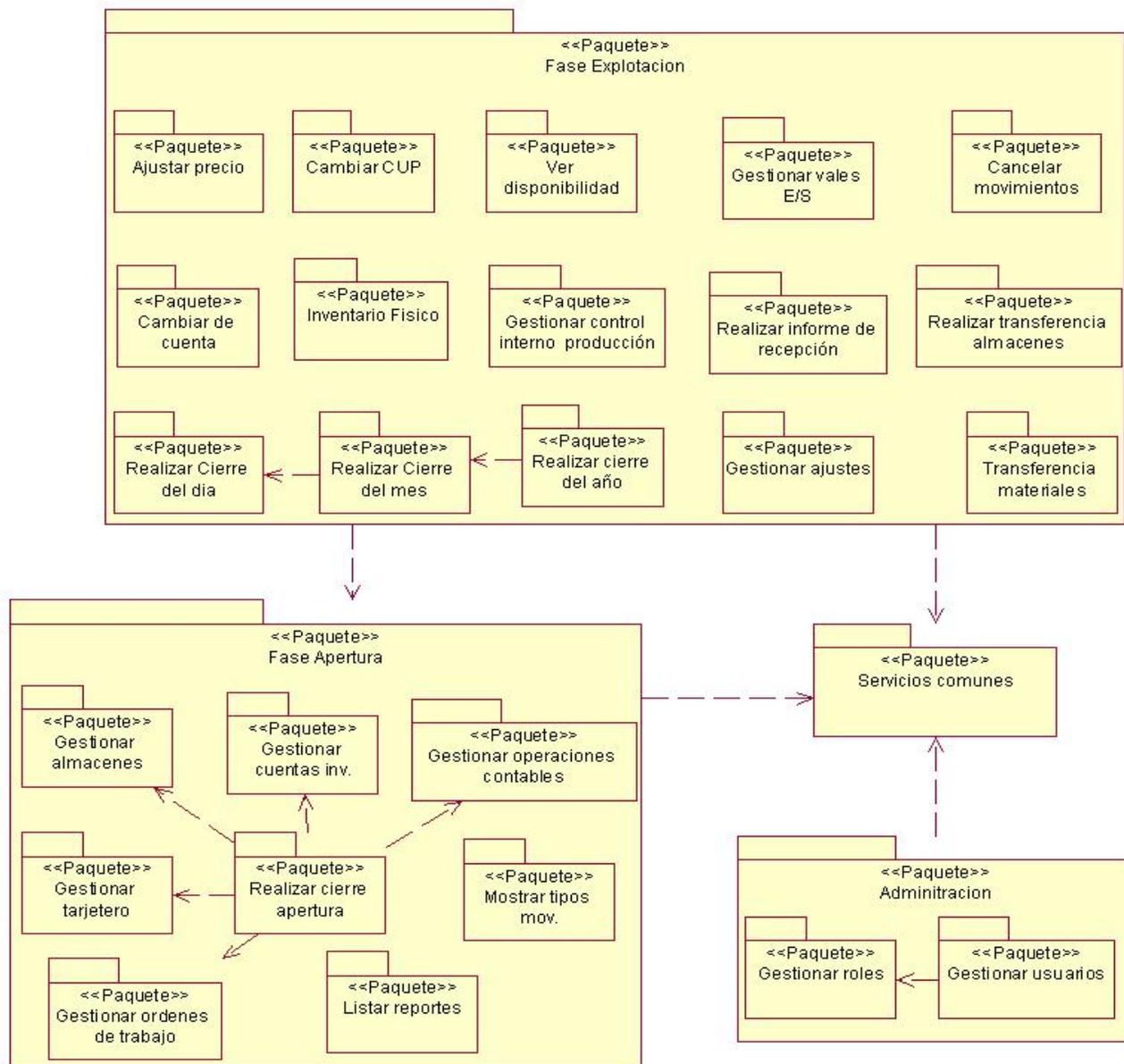


Fig. A4. Sistema de Empaquetado

Anexos 5: Diagrama de Clases de Análisis de los CU Críticos

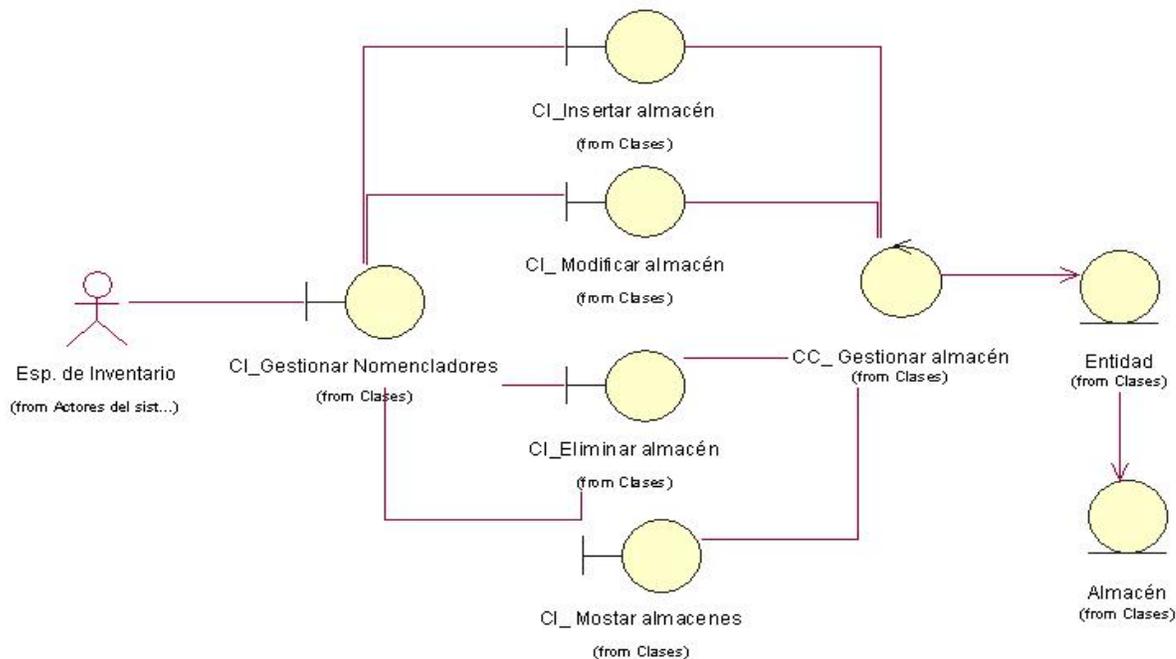


Fig. A5.1 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Gestionar almacenes.

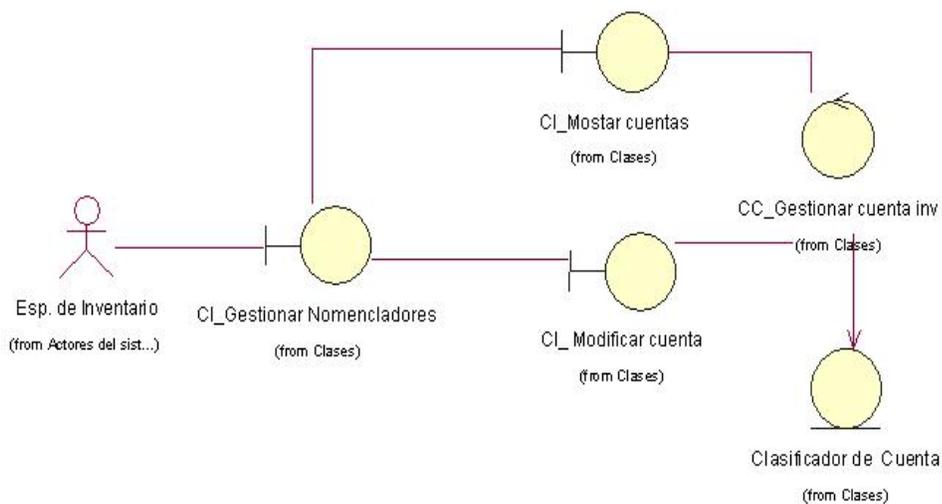


Fig. A5.2 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Gestionar cuentas de inventario.

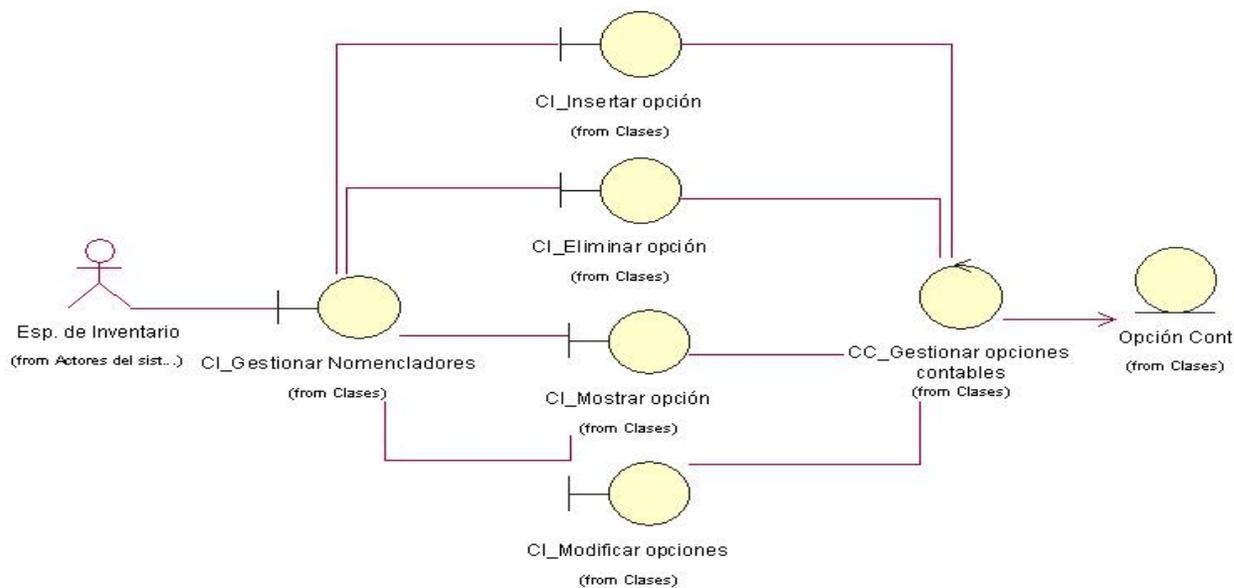


Fig. A5.3 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Gestionar opciones contables.

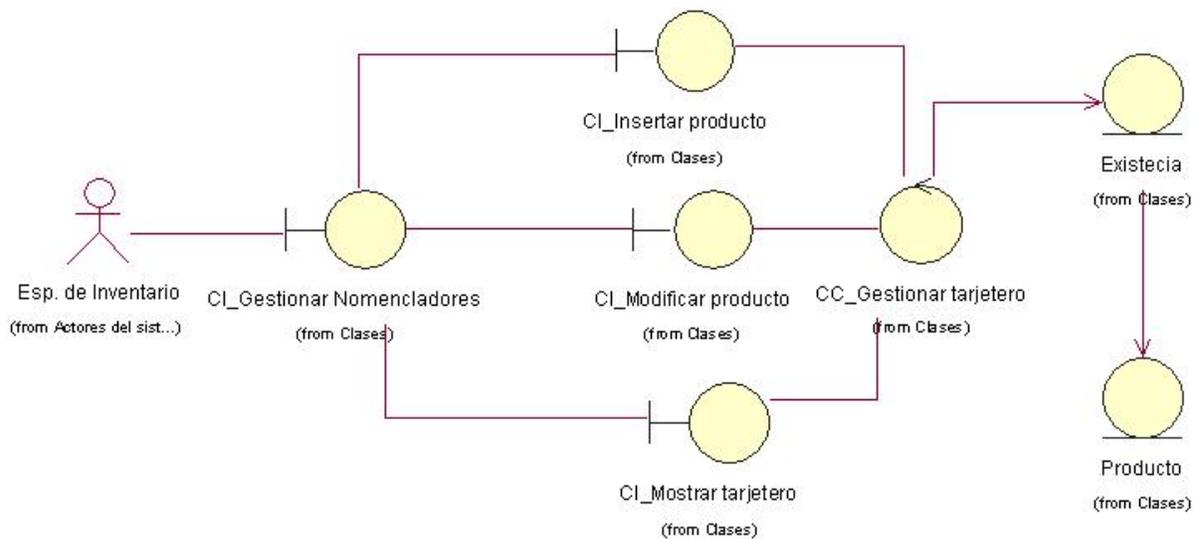


Fig. A5.4 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Gestionar tarjetero.

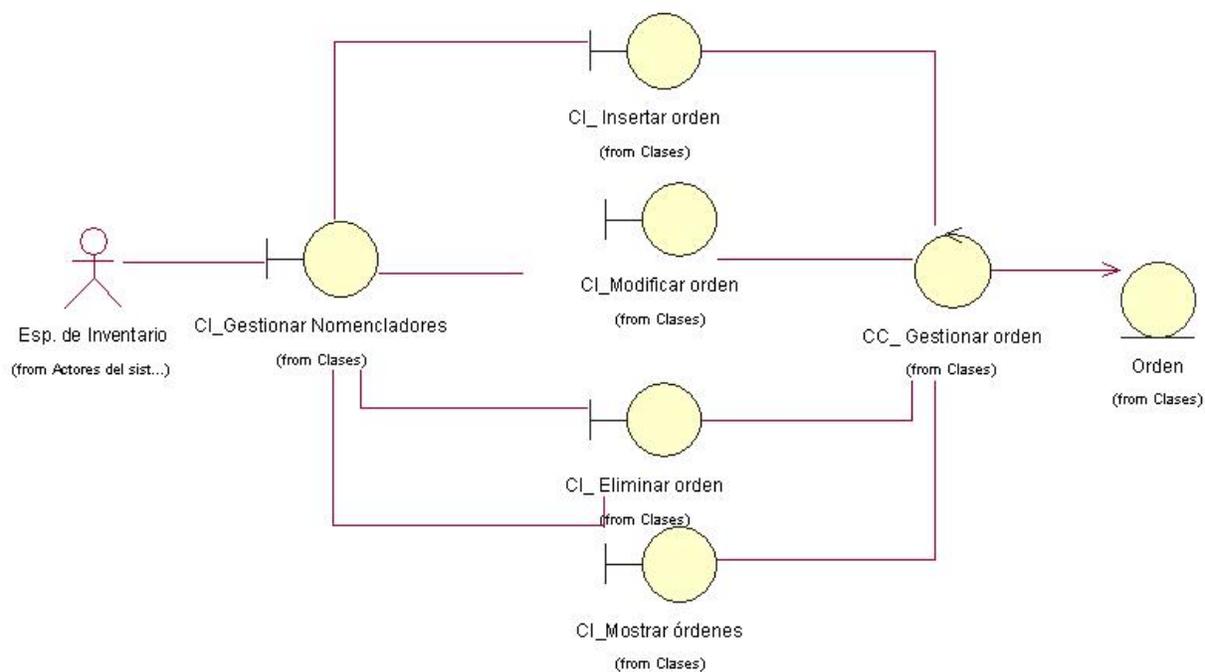


Fig. A5.5 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Gestionar órdenes de Trabajo

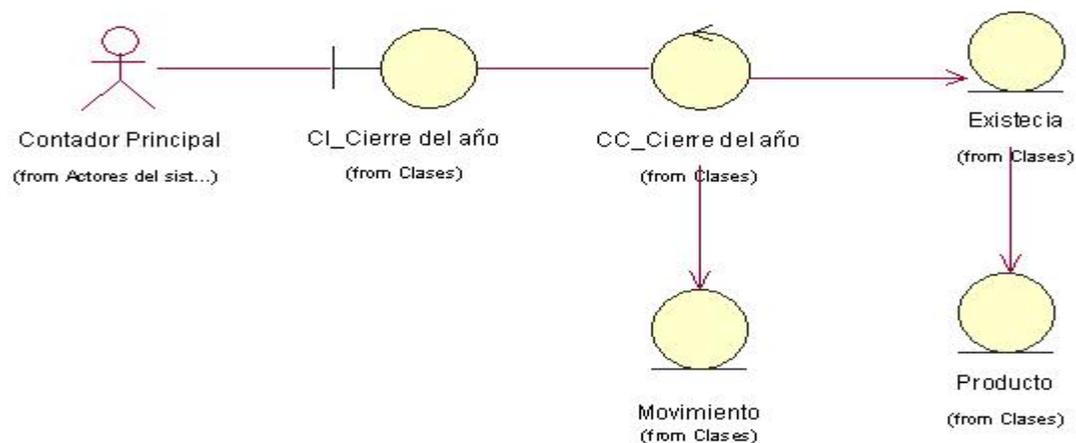


Fig. A5.6 Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Cierre del año

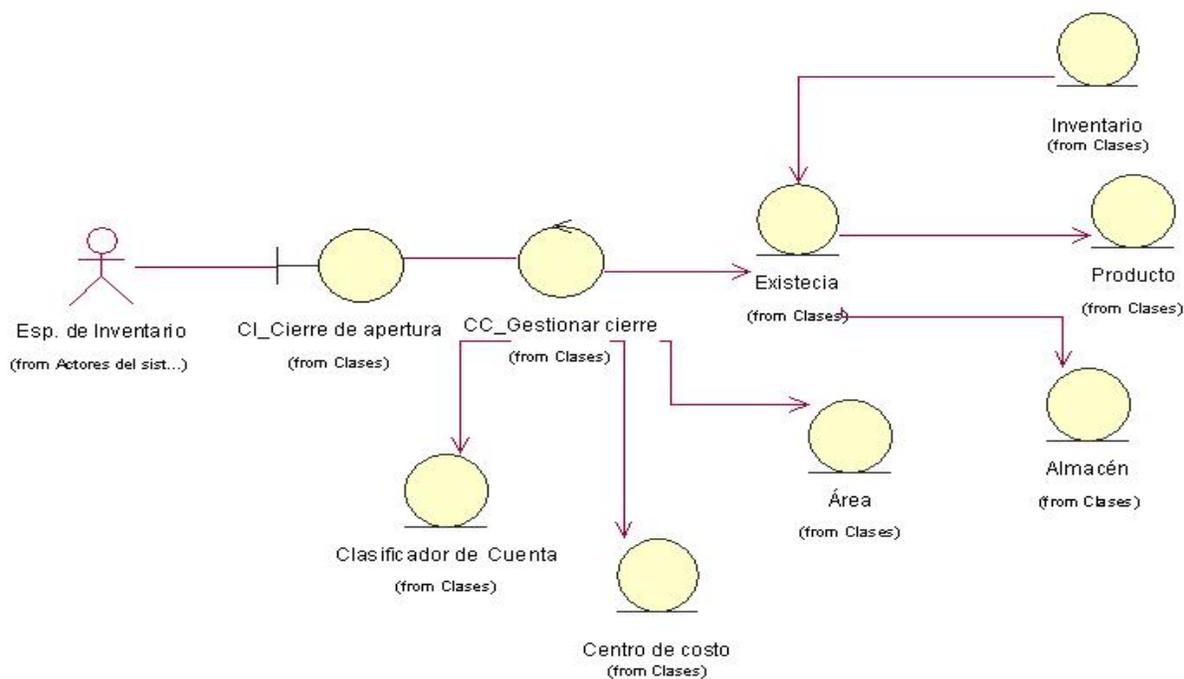


Fig. A5.7 Diagrama de Clases de Análisis. Paquete Fase de Apertura. Subpaquete Cierre de apertura.

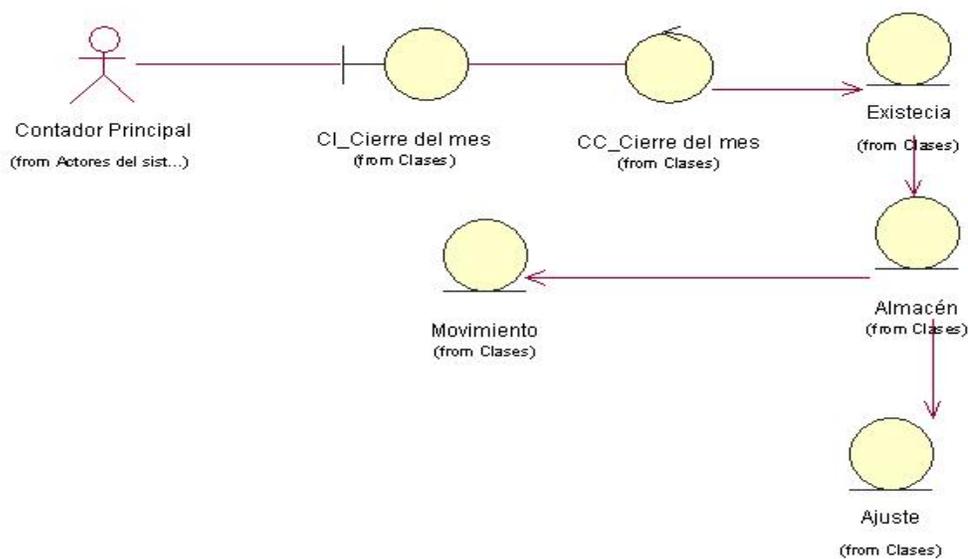
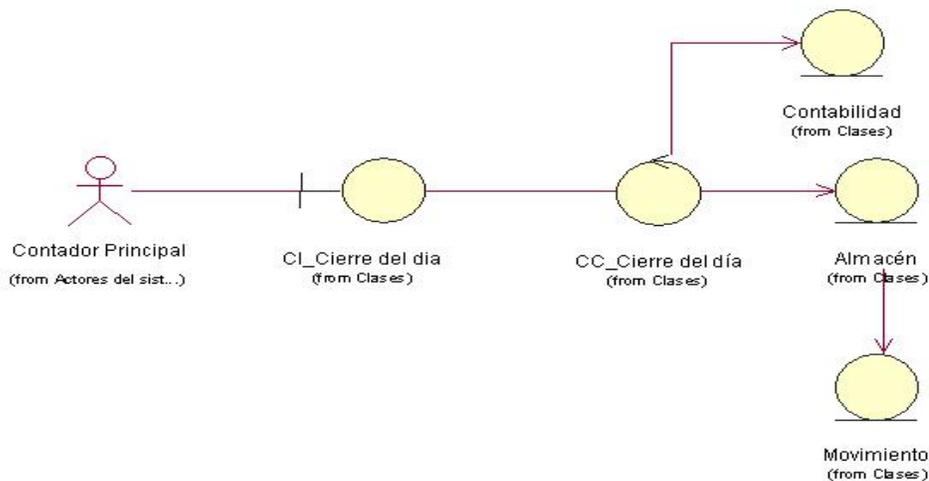


Fig. A5.8. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Cierre del mes.



. Fig. A5.9. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Cierre del día.

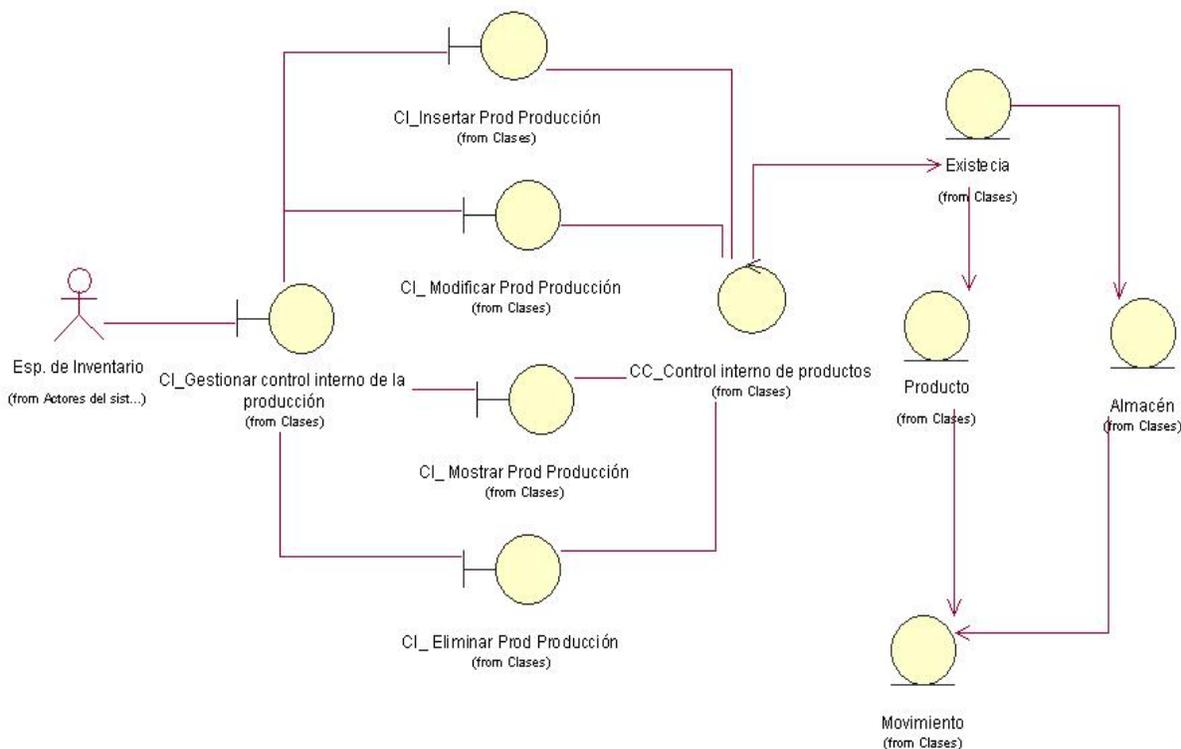


Fig. A5.10. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Gestionar control interno de productos.

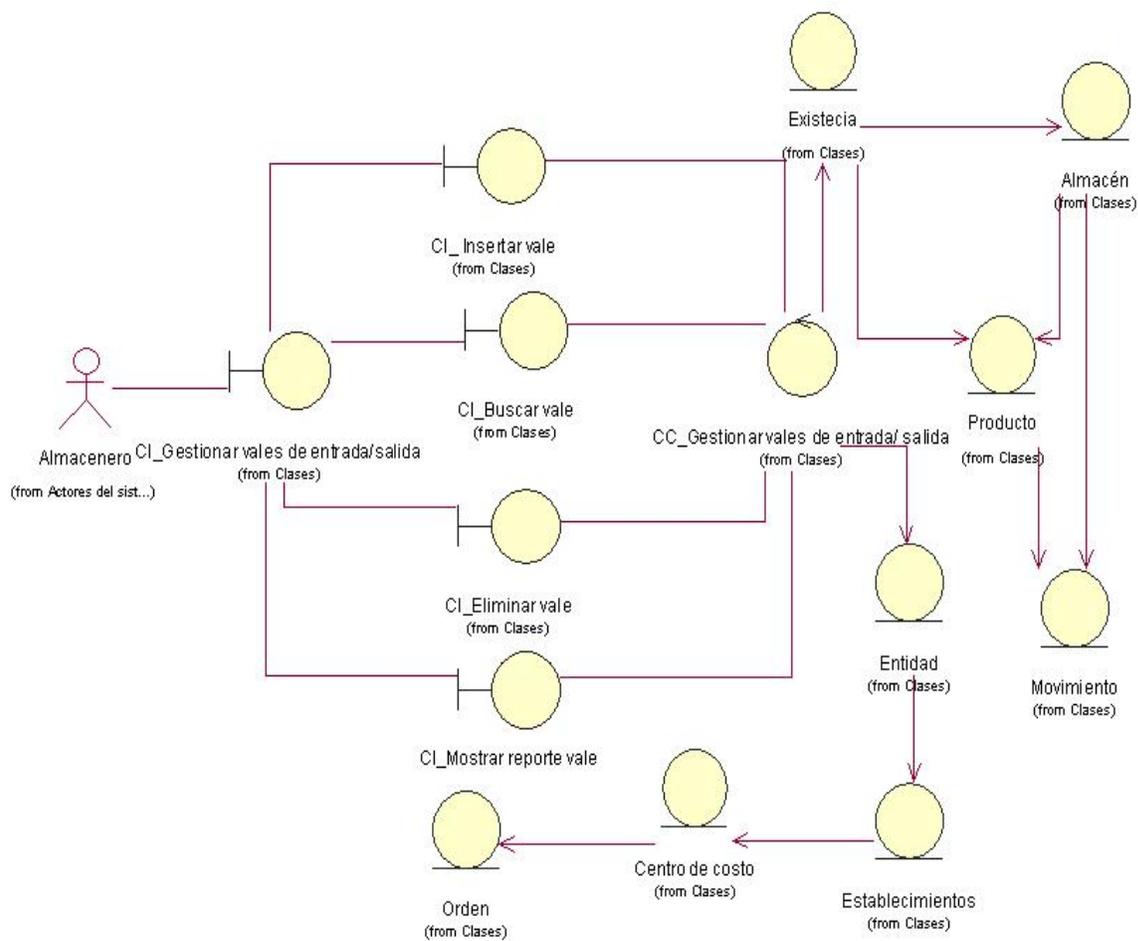


Fig. A5.11. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Gestionar vales de entrada / salida.

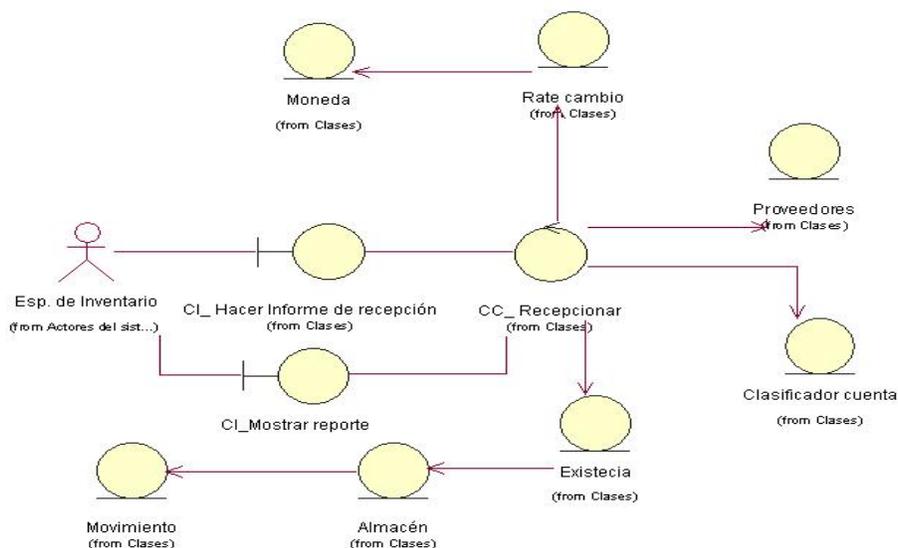


Fig. A5.12. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete Hacer informe de recepción.

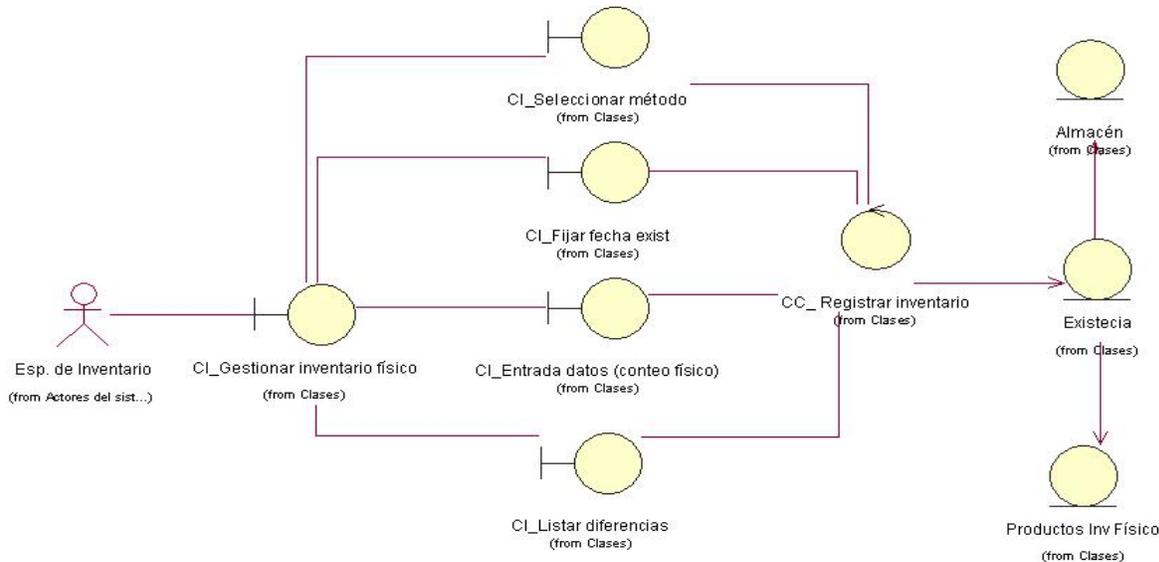


Fig. A5.13. Diagrama de Clases de Análisis: Paquete Fase Explotación. Subpaquete realizar inventario físico.

Anexo 6: Diagramas de Clases del Diseño de los CU críticos

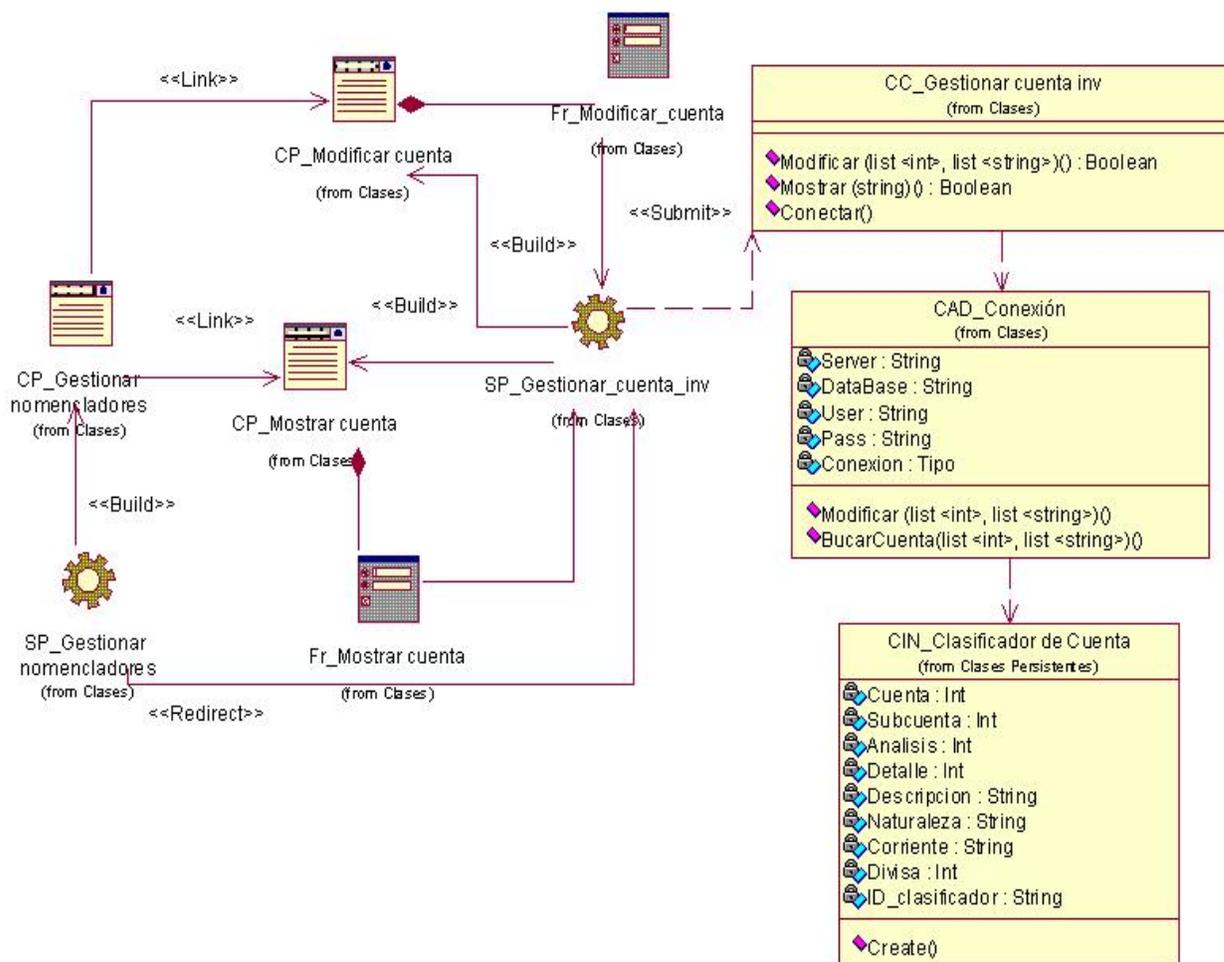


Fig. A6.1. Diagrama de Clases de Diseño: Paquete Fase Apertura, Gestionar cuentas de inventario.

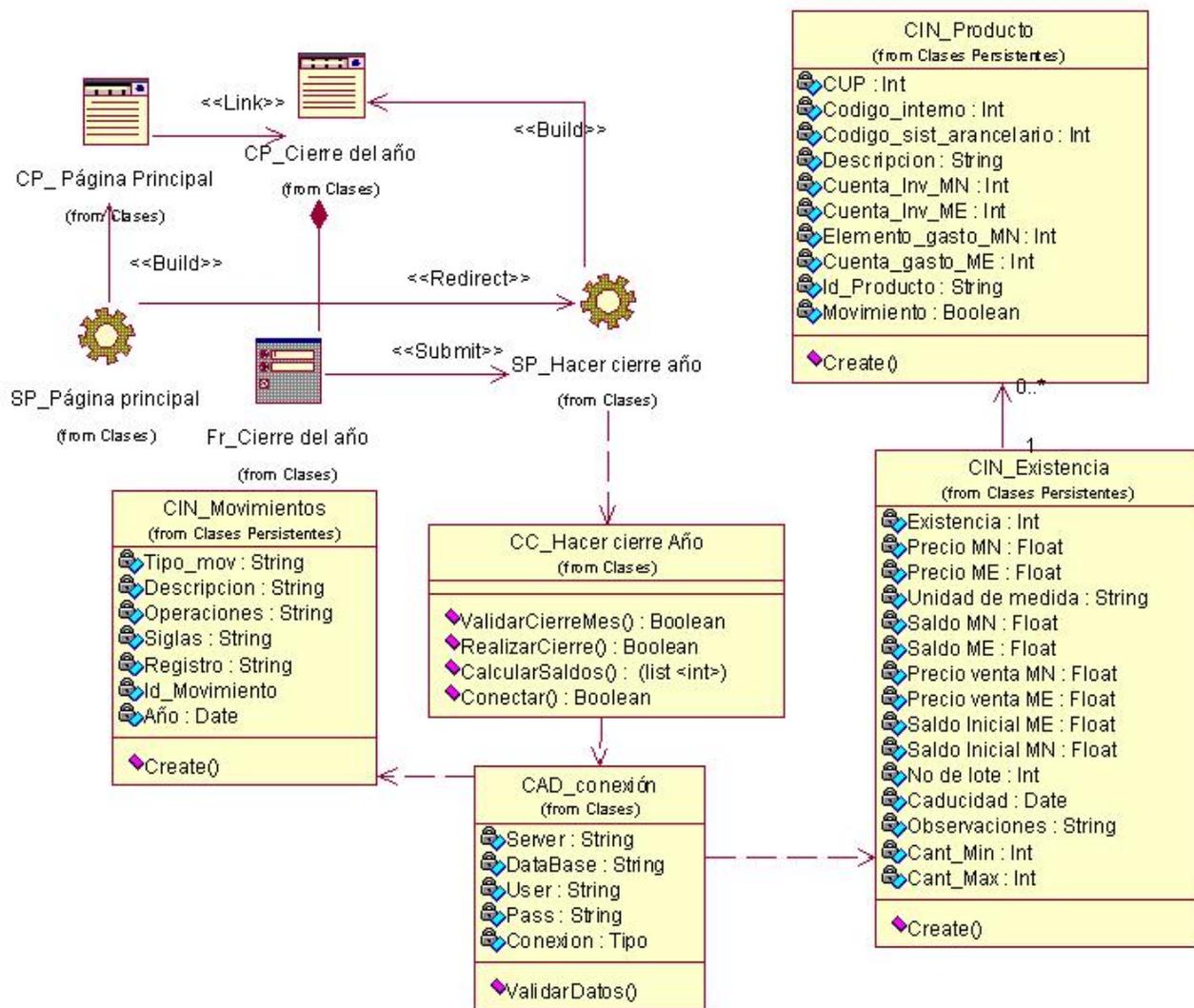


Fig. A6.2. Diagrama de Clases de Diseño: Paquete Fase Explotación, Cierre del Año.

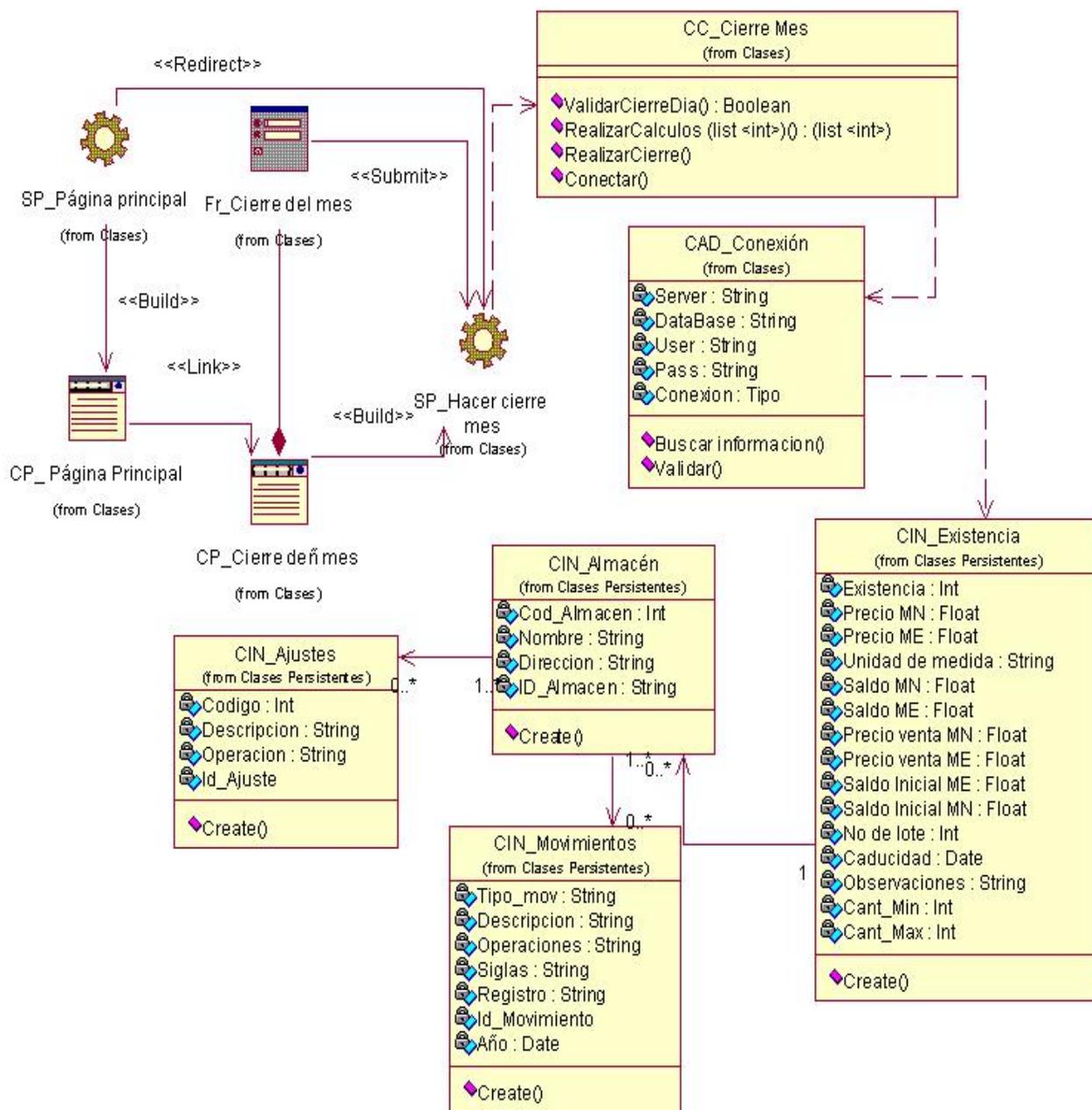


Fig. A6.3. Diagrama de Clases de Diseño: Paquete Fase Explotación, Cierre del Mes

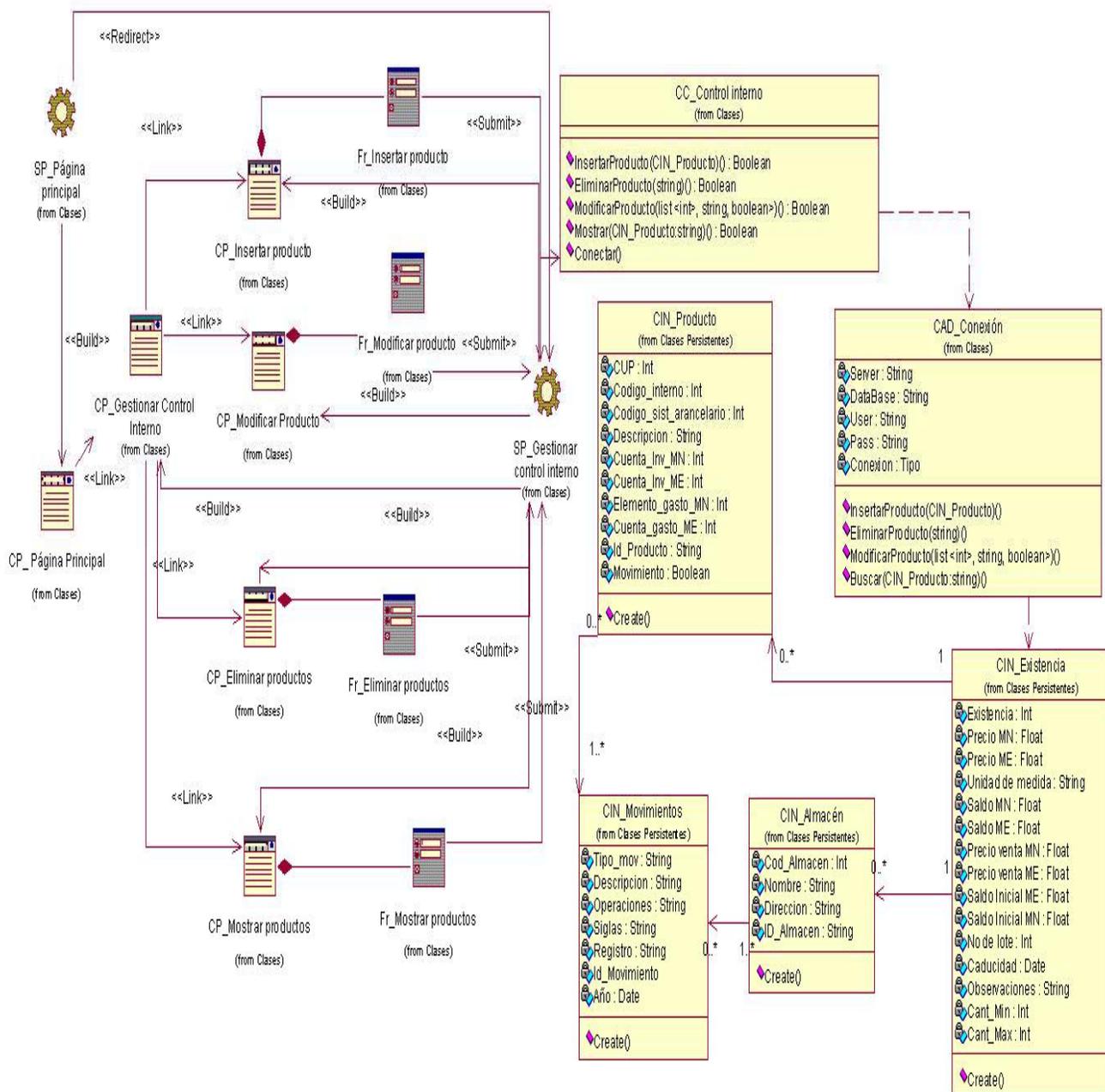


Fig. A6.4 Diagrama de Clases de Diseño: Paquete Fase Explotación, Principal, Control Interno de la Producción

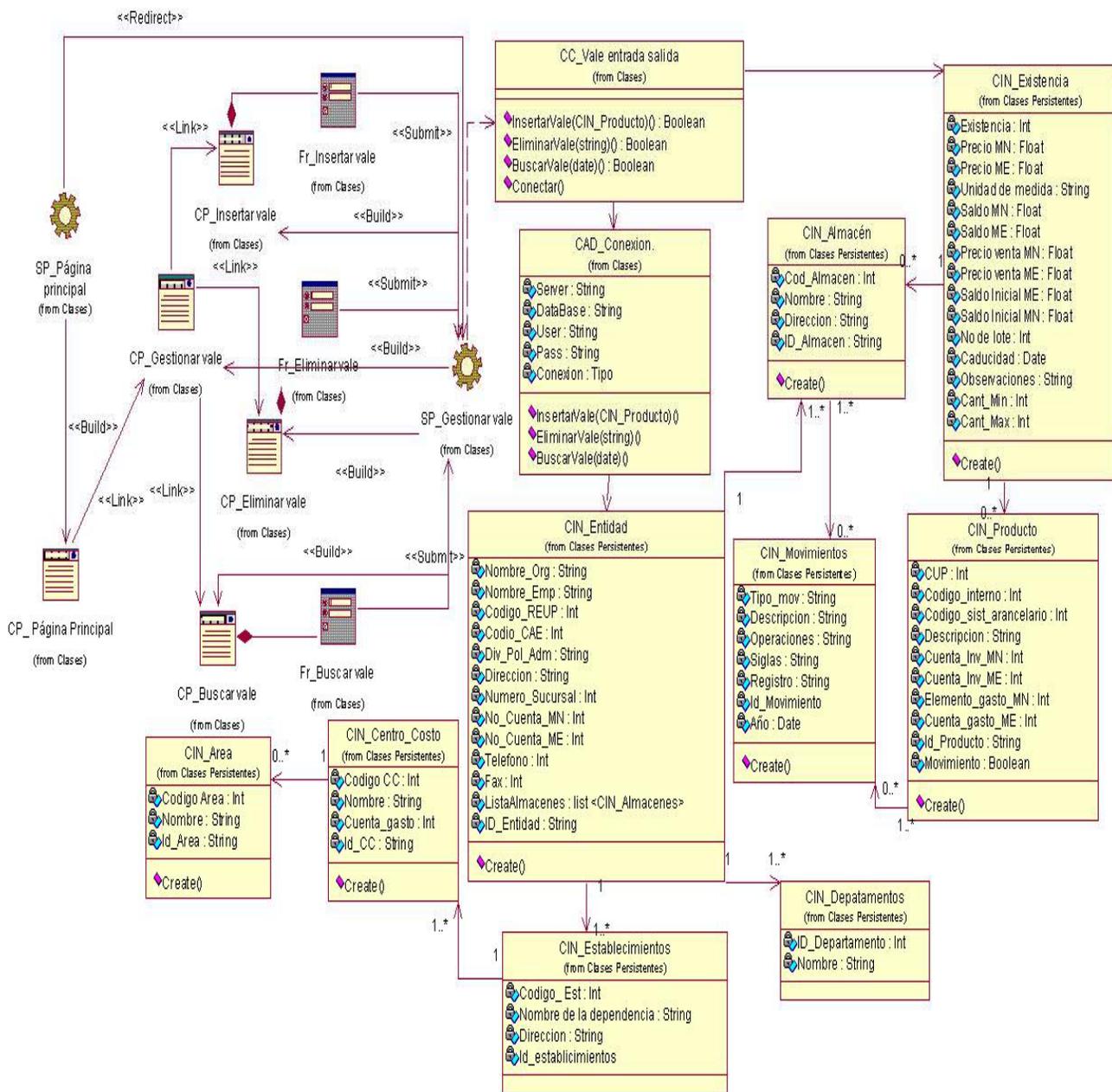


Fig. A6.5. Diagrama de Clases de Diseño: Paquete Fase Explotación, Principal, Gestionar Vales de Entrada/Salida

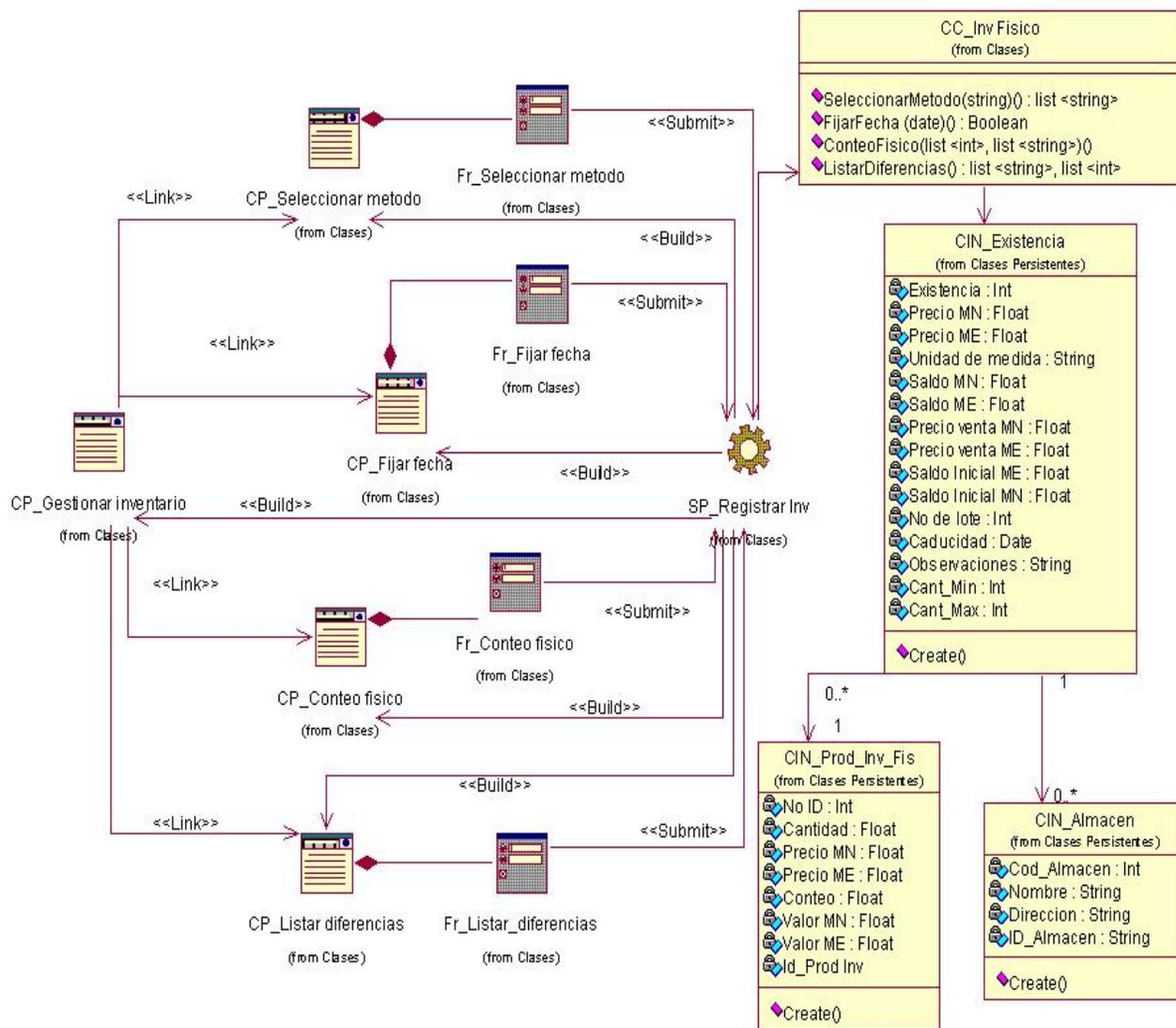


Fig. A6.6. Diagrama de Clases del Diseño: *Subsistema Fase de Explotación. Inventario Físico.*

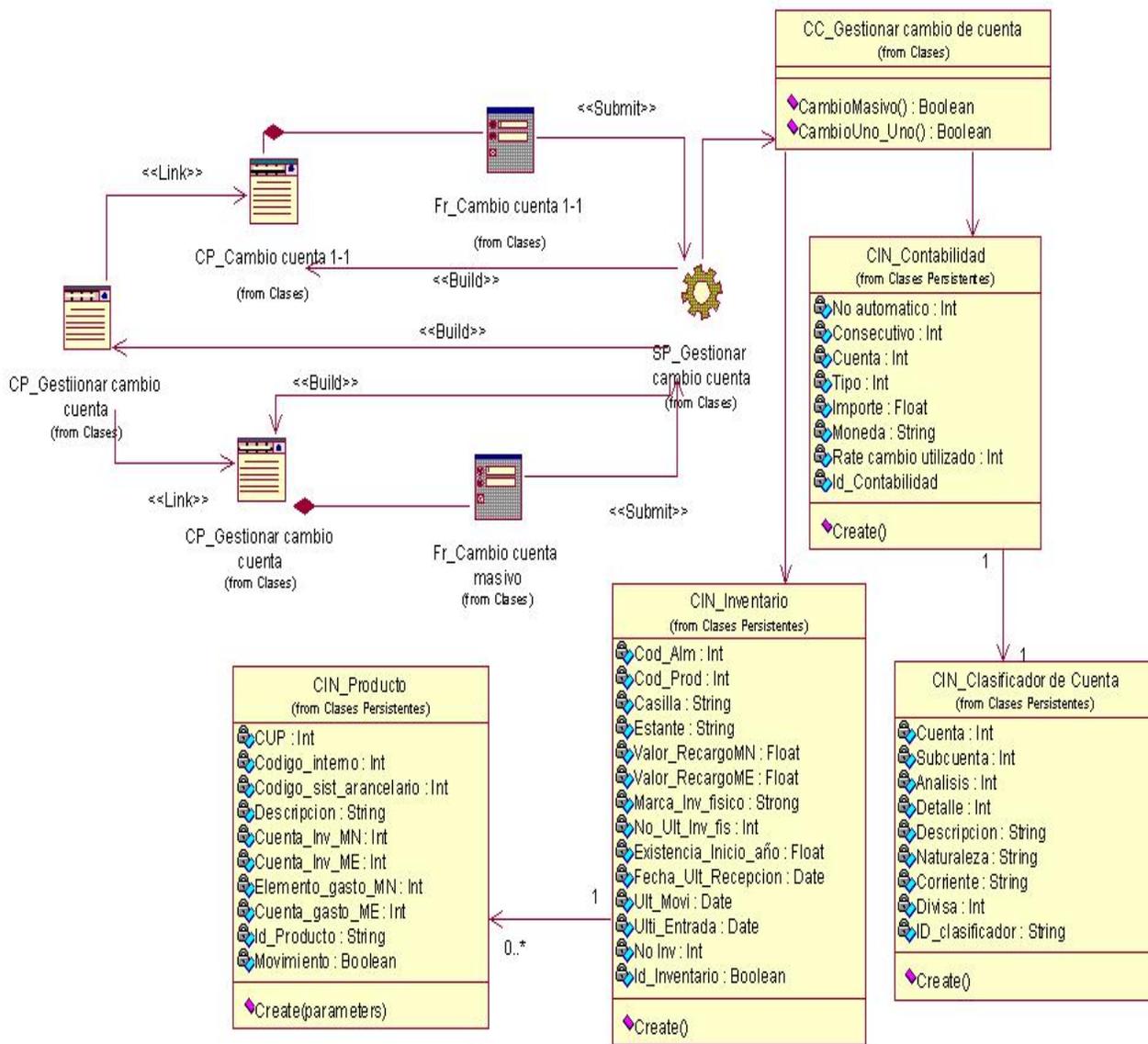


Fig. A6.7. Diagrama de Clases de Diseño: Paquete Fase Apertura, Gestionar cuenta de inventario.

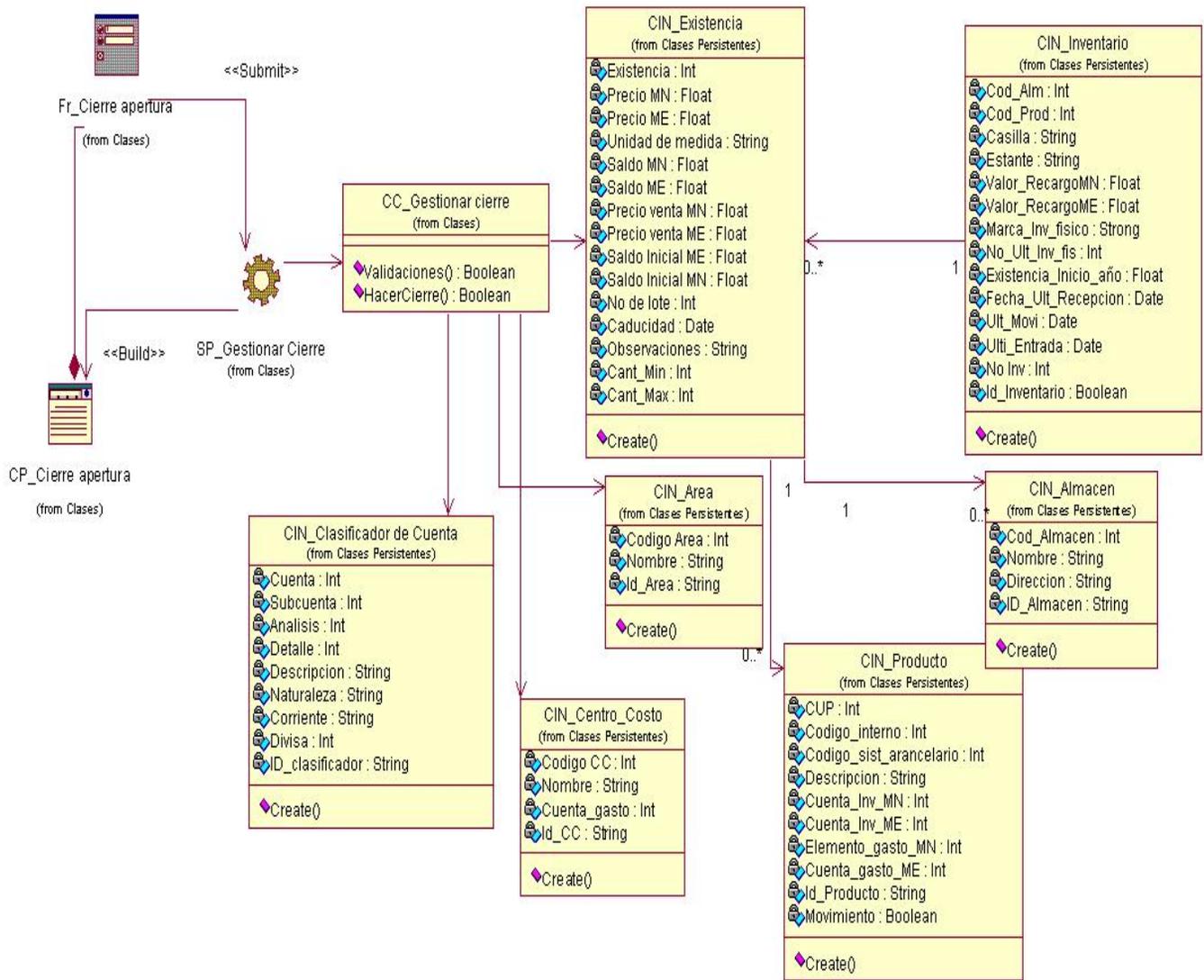


Fig. A6.8. Diagrama de Clases de Diseño: Paquete Fase Apertura, Cierre apertura.

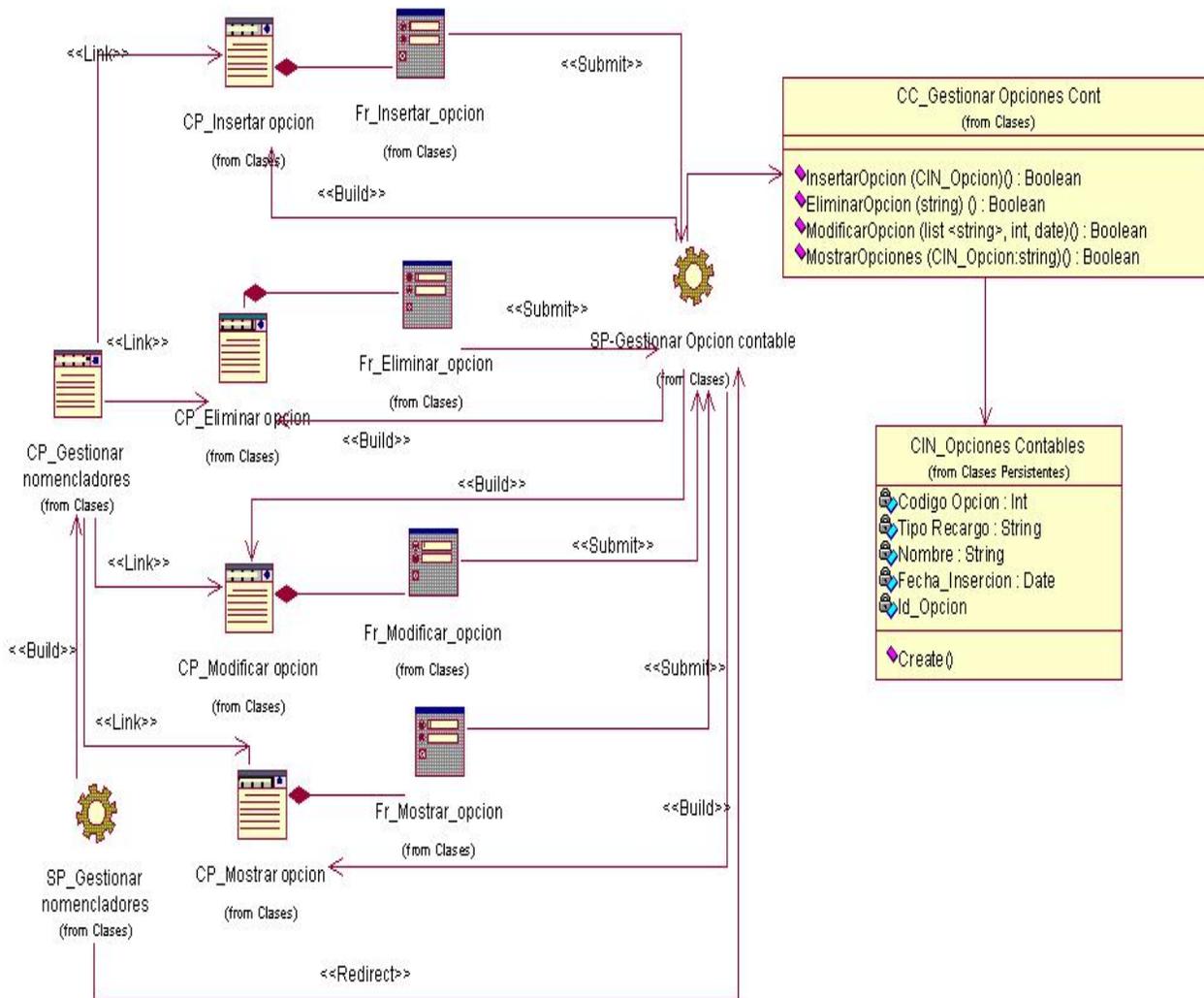


Fig. A6.9. Diagrama de Clases de Diseño: Paquete Fase Apertura, Gestionar opciones contables.

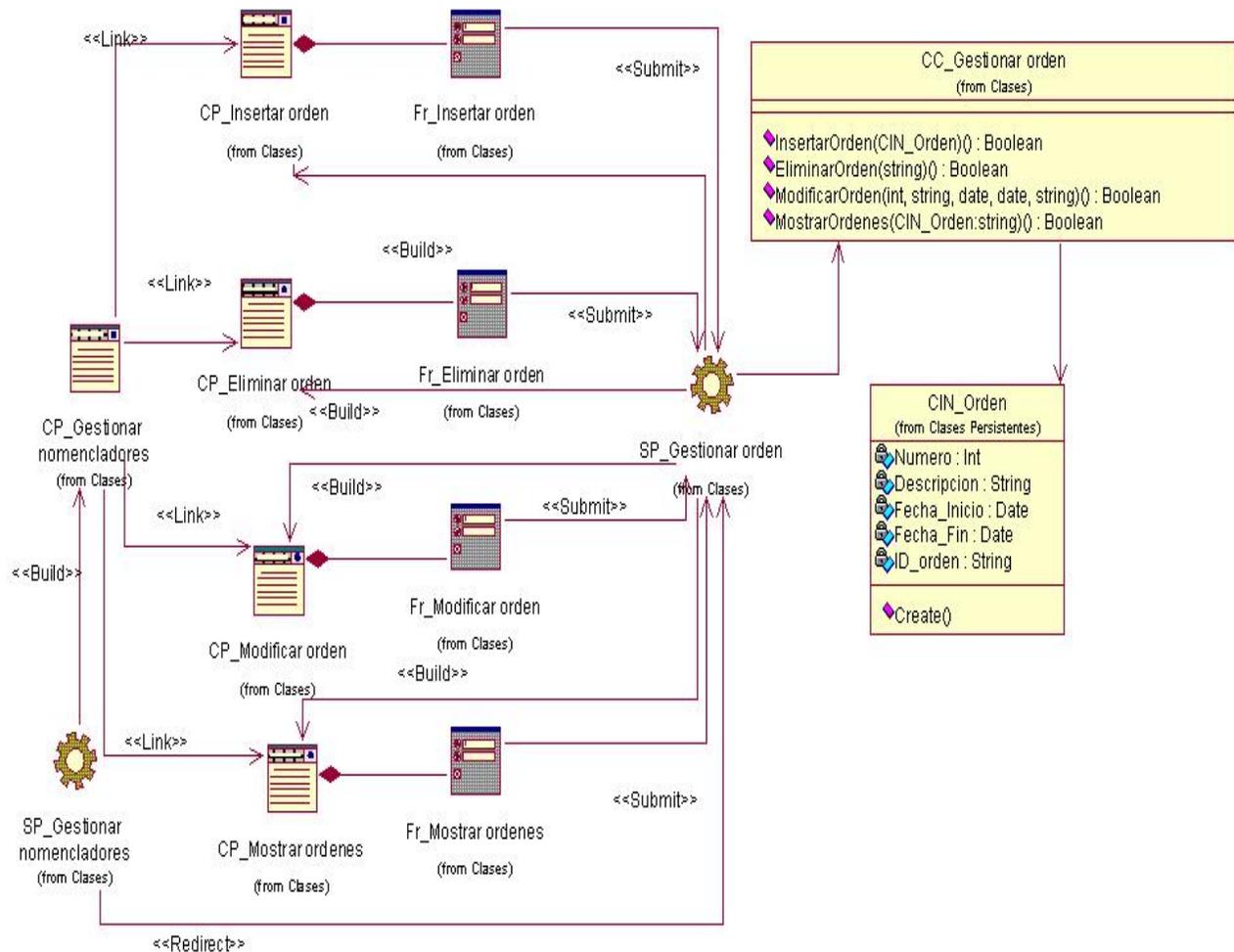


Fig. A6.10. Diagrama de Clases de Diseño: Paquete Fase Apertura, Gestionar ordenes de trabajo.

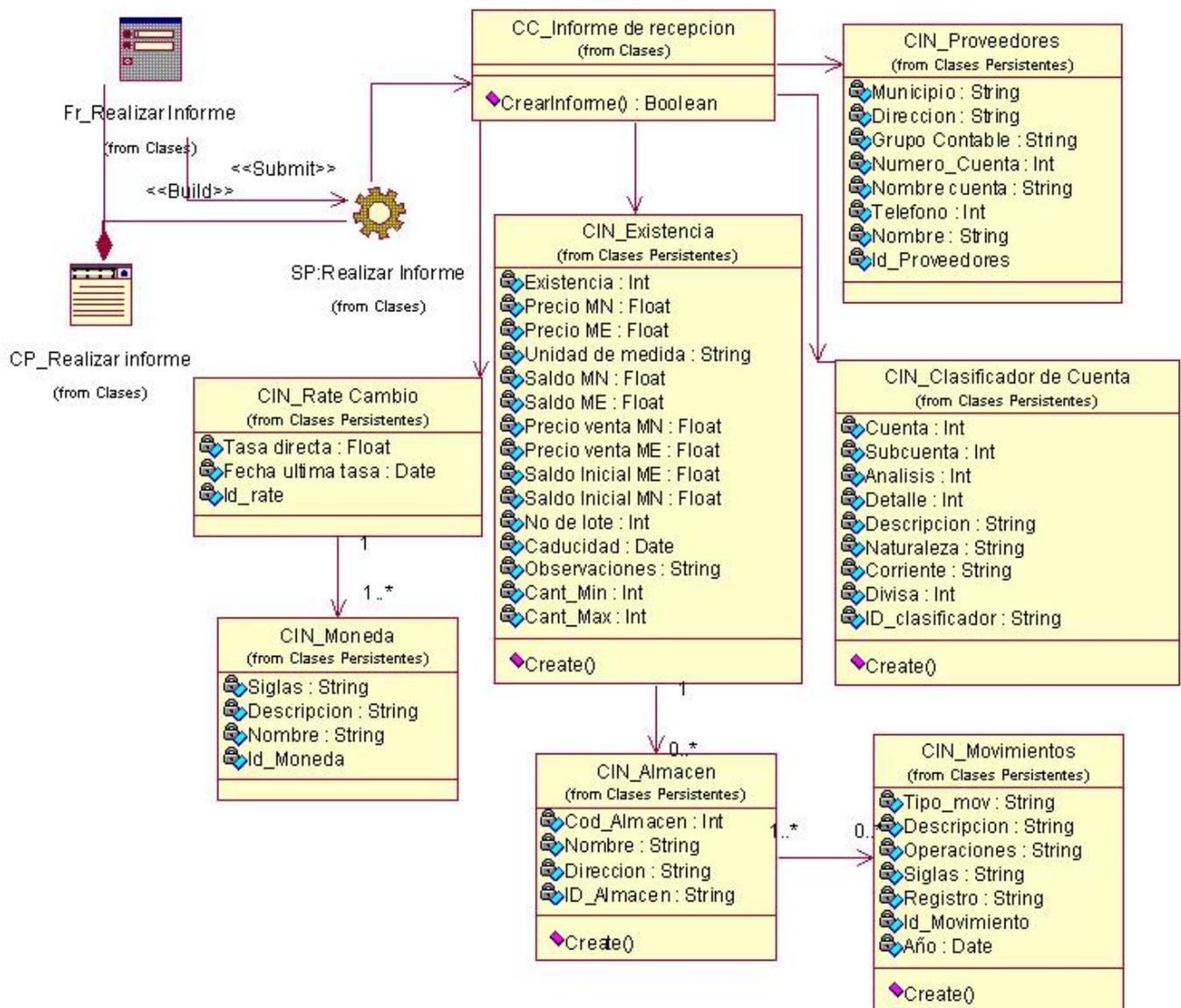


Fig. A6.11. Diagrama de Clases de Diseño: Paquete Fase Apertura, Realizar Informe de recepción.

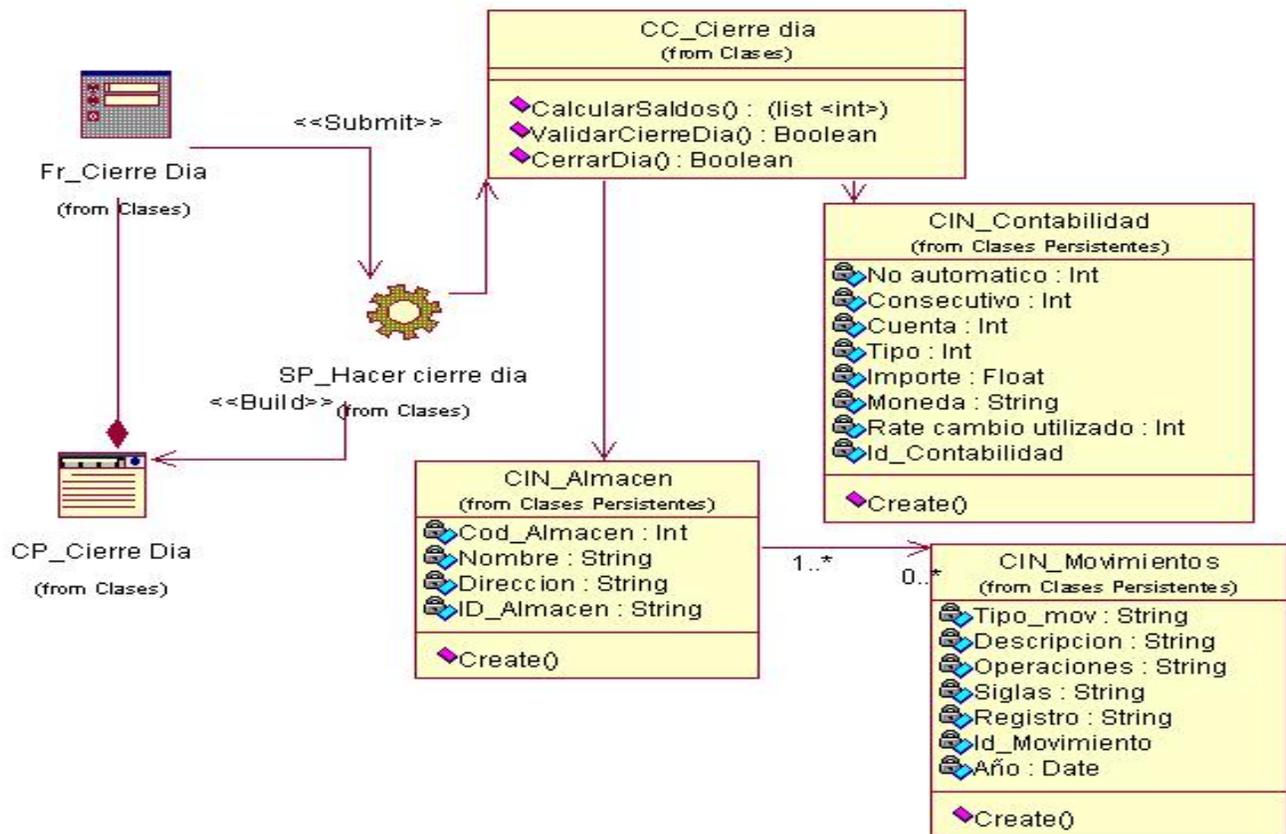


Fig. A6.12. Diagrama de Clases de Diseño: Paquete Fase Explotación, Cierre del Día.

Anexo 7 Diagrama de clases persistentes

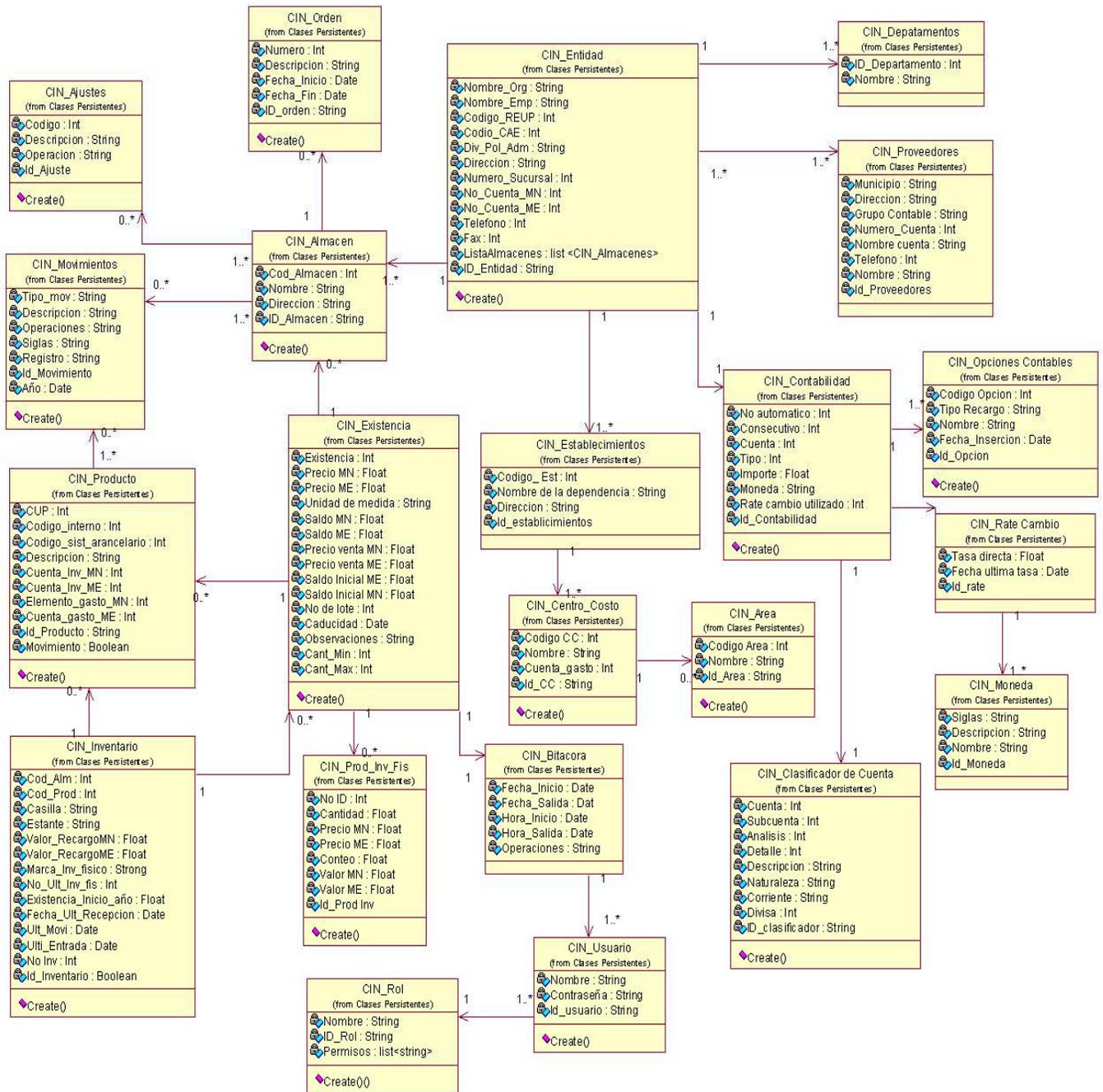


Fig. A7. Diagrama de Clases Persistentes.

Anexo 8 Modelo de Datos

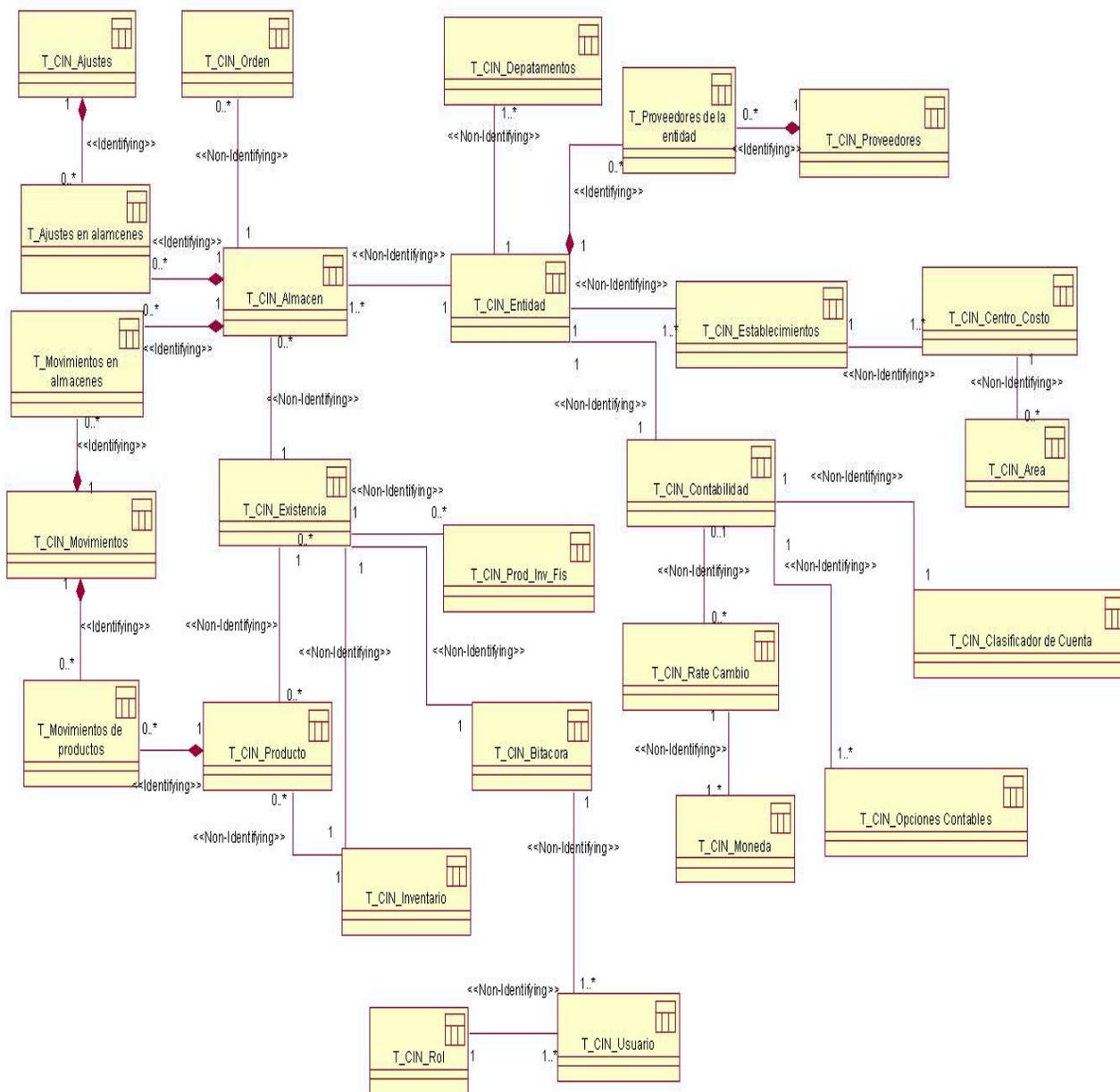


Fig. A8. Modelo de Datos.

Anexo 9 Modelo de Despliegue

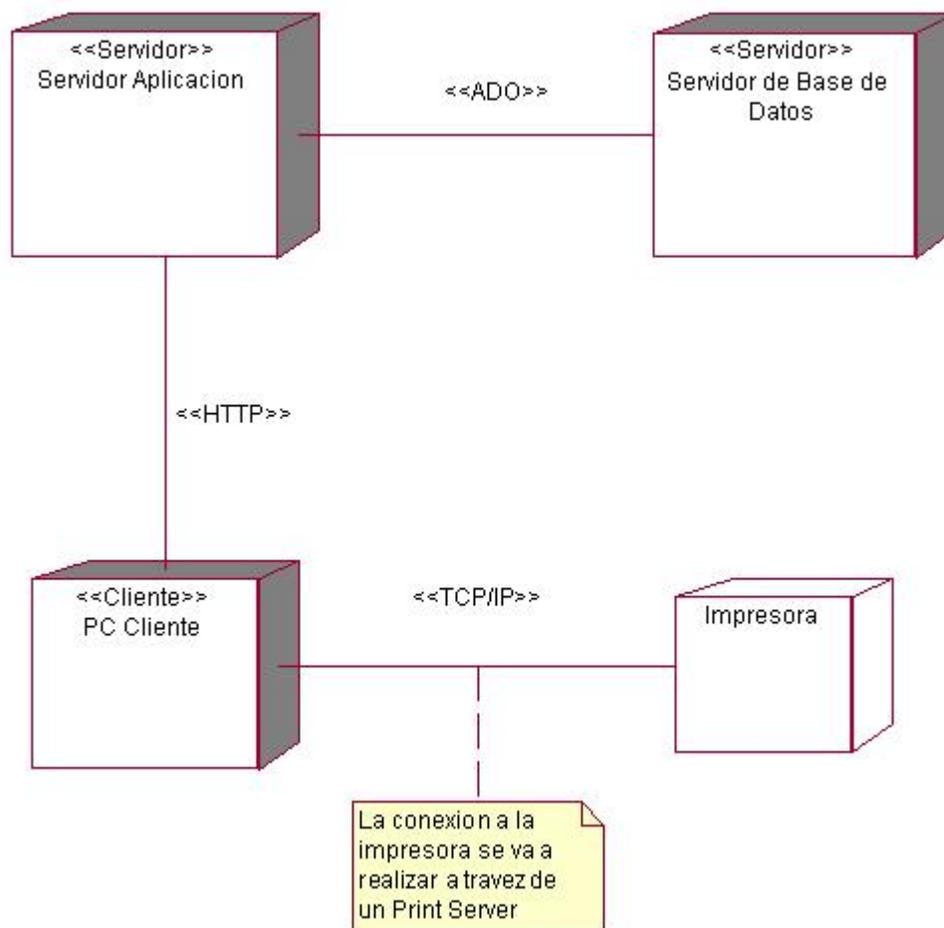


Fig. A9. Modelo de Despliegue.