

Universidad de las Ciencias Informáticas

Facultad # 3



Título: Solución informática para el control de las
Reservas Materiales

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

Autor (es): Yanelis Nuñez Arcia

Enmanuel Riverón Frías

Tutores: Ing. Javier Ramírez Hernández

Ing. Lester González Gutiérrez

Ing. Yuriel Martínez Ordaz

Ciudad de La Habana, Junio, 2011

Año 53 de la Revolución.

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de la **Solución informática para el control de las Reservas Materiales** y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Enmanuel Riverón Frías

Yanelis Nuñez Arcia

Ing. Javier Ramírez Hernández

Ing. Lester González Gutiérrez

Ing. Yuriel Martínez Ordaz

RESUMEN

El uso de herramientas de software para la realización de la gestión y la informatización de procesos en las diferentes entidades, ha conllevado a la obtención de mejores resultados económicos, dada la optimización de los procesos y el mejor aprovechamiento de las inversiones y recursos destinados a éstas. Como consecuencia de la gran cantidad de información que se genera es necesaria la utilización de un sistema de gestión que contemple el tratamiento de las reservas materiales y permita tanto administrar como controlar los recursos del país.

El objetivo principal que persigue este trabajo de diploma es informatizar los distintos procesos de reserva en las entidades cubanas, como parte de los procesos del control de inventarios del Sistema Integral de Gestión (Cedrux), garantizando una herramienta poderosa capaz de optimizar estos procesos de forma tal que satisfagan las necesidades del cliente. Para lograrlo fue necesario estudiar la metodología de desarrollo del software a utilizar, lenguajes de modelado para generar los artefactos y las herramientas. Además se analizó la documentación que fue entregada por los clientes, con el fin de comprender los procesos de negocio y una posterior captura de requisitos, logrando así, que los servicios brinden mayor funcionalidad y seguridad, garantizando la calidad del sistema y así obtener resultados positivos.

PALABRAS CLAVE

Procesos, Reserva, Software.

ÍNDICE

| | |
|--|----|
| INTRODUCCIÓN | 1 |
| CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA | 6 |
| 1.1 Introducción | 6 |
| 1.2 Reserva | 6 |
| 1.3 Soluciones informáticas que realizan el proceso de Reserva..... | 7 |
| 1.3.1 Soluciones informáticas Internacionales..... | 7 |
| 1.3.2. Solución informática nacional..... | 9 |
| 1.4 Valoración del estado del arte..... | 9 |
| 1.5 Modelo de desarrollo | 10 |
| 1.6 Marco de trabajo..... | 11 |
| 1.6.1 Herramientas | 11 |
| Sistema Gestor de Base de Datos (SGBD) | 12 |
| 1.6.2 Tecnologías..... | 12 |
| 1.7 Lenguajes de modelado y desarrollo | 14 |
| 1.7.1 Lenguajes de modelado..... | 14 |
| 1.8 Herramienta de diseño..... | 15 |
| 1.9 Patrones de diseño | 16 |
| 1.10 Técnicas para la captura de requisitos | 18 |
| 1.11 Conclusiones parciales..... | 19 |
| CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA | 20 |
| 2.1 Introducción | 20 |
| 2.2 Objeto de estudio..... | 20 |
| 2.2.1 Descripción de los procesos del negocio | 20 |
| 2.2.2 Objeto de automatización | 21 |
| 2.2.3 Información que se maneja..... | 23 |
| 2.2.4 Propuesta del sistema | 24 |
| 2.3 Modelación del negocio | 24 |
| 2.3.1 Diagramas de proceso del negocio (BPMN) | 25 |
| 2.3.2 Especificación del proceso Tratamiento de la Reserva | 25 |
| 2.3.3 Validación de procesos del negocio..... | 29 |
| 2.4 Requerimientos y funcionalidades | 29 |
| 2.4.1 Aplicación de las técnicas de validación de requisitos | 29 |
| 2.4.2 Listado de Requisitos Funcionales Identificados..... | 30 |
| 2.4.3 Especificación de Requisitos Funcionales | 31 |
| 2.4.4 Validación de los Requisitos Funcionales | 34 |
| 2.4.5 Requisitos No Funcionales | 34 |
| 2.5 Conclusiones parciales | 37 |
| CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA..... | 38 |
| 3.1 Introducción..... | 38 |
| 3.2 Análisis | 38 |
| 3.2.1 Modelo de clases del análisis | 38 |
| 3.3 Arquitectura | 38 |
| 3.3.1. Estilos arquitectónicos | 39 |
| 3.4 Aplicación de los patrones de diseño | 42 |

| | | |
|--|--|----|
| 3.5 | Modelo de clases del diseño..... | 43 |
| 3.4.1. | Mecanismos de diseño | 44 |
| 3.6 | Diagramas de secuencias..... | 48 |
| 3.6 | Diagrama de clases del diseño Web | 49 |
| 3.7 | Diseño de la Base de Datos..... | 50 |
| 3.7.1. | Modelo lógico de datos | 50 |
| 3.7.2. | Modelo físico de datos | 51 |
| | Figura 22 Modelo físico de datos..... | 51 |
| 3.7.3. | Descripción de las tablas | 51 |
| 3.8 | Conclusiones parciales | 53 |
| CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA..... | | 54 |
| 4.1 | Introducción | 54 |
| 4.2 | Implementación | 54 |
| 4.2.1 | Diagrama de despliegue | 54 |
| 4.2.2 | Diagrama de componentes | 55 |
| 4.3 | Prueba | 55 |
| 4.3.1 | Pruebas de unidad..... | 55 |
| 4.4 | Validación del modelo de diseño utilizando métricas | 63 |
| 4.4.1. | Tamaño operacional de clase (TOC) | 64 |
| 4.4.2. | Relaciones entre clases (RC) | 64 |
| 4.4.3. | Profundidad de herencia (PH)..... | 65 |
| 4.4.4. | Número de Operaciones Redefinidas para una clase hija (NOR) | 65 |
| 4.4.5. | Resultados de los instrumentos de evaluación de las métricas..... | 65 |
| 4.4.6. | Matriz de cubrimiento de los parámetros de calidad evaluados con las métricas propuestas | 69 |
| 4.5 | Conclusiones parciales | 71 |
| CONCLUSIONES GENERALES..... | | 72 |
| RECOMENDACIONES | | 73 |
| BIBLIOGRAFÍA | | 74 |
| GLOSARIO DE TÉRMINOS GENERAL | | 78 |

INTRODUCCIÓN

El uso de la informática como herramienta, es vital para el desarrollo y la gestión de entidades que marcan la diferencia a la hora de resolver determinados problemas, que afectan el avance de la economía de aquellos países que se encuentran inmersos en la evolución tecnológica. La informatización de procesos y servicios ha sido un factor esencial en el reestructuramiento de métodos en sistemas informatizados, logrando disponer de la gestión de datos con precisión a la hora de tomar decisiones importantes, que posibilitan la mejora para la optimización de los resultados en Sistemas de Gestión, con el propósito de obtener un mejor funcionamiento, calidad y precisión en las tareas que se realizan.

Las entidades cubanas destinan esfuerzo al control de los medios materiales sobre los cuales trabajan. Por estas razones utilizan sistemas logísticos entre los que se encuentran el Stock Empresarial Mistral Caribe, Versat Sarasola, SISCONT 5, RODAS XXI, entre otros. Estos recursos son almacenados en depósitos de cada entidad, donde se encuentran bienes independientes al resto de los productos, llamados Reservas Materiales que no entran en el proceso del control habitual.

El proceso del Tratamiento de las Reservas Materiales, forma parte de las actividades para garantizar el desarrollo normal de la economía, la vida de la población, mantener la capacidad de resistencia del país, asegurar la realización de las acciones combativas, así como mantener la seguridad y el orden interior en situaciones excepcionales, garantizando el restablecimiento de la nación ante cualquier desastre.

Las Reservas Materiales son productos almacenados, con el objetivo de ser utilizados cuando el país lo necesite. Estos productos forman parte del proceso de inventario. Los Inventarios son bienes tangibles utilizados para la venta en el curso ordinario del negocio o para ser consumidos en la producción de bienes o servicios para su posterior comercialización. Lo conforman materias primas, productos en proceso y terminados o mercancías para la venta, los materiales, respuestos y accesorios que son consumidos en la producción de bienes fabricados para la venta o en la prestación de servicios; empaques, envases y los inventarios en tránsito. La base de toda empresa comercial es la compra y venta de bienes o servicios; de ahí la importancia del manejo del inventario por parte de la misma. El control de inventarios se encarga de regular de forma adecuada las existencias en los almacenes de herramientas, materias primas, productos en proceso y terminados; protegiendo a la empresa de costos innecesarios por acumulación o falta de existencias en el almacén. Este control de inventario, es la base del control organizado y del seguimiento preciso de la evolución de los recursos de una entidad. Se obtiene de manera directa y se emplea formularios y/o dispositivos electrónicos para la captura de la información, en este caso sería recomendable utilizar sistemas ERP.

Los sistemas de Planificación de los Recursos Empresariales (ERP) pueden ser vistos como soluciones para tecnologías de la información que permiten integrar los procesos de competencias de las empresas.

Estos productos son modulares en estructura y comprenden diversas áreas en su logística, como planificación financiera, ventas, procesamiento de órdenes, producción y planificación de los recursos humanos y materiales. Las organizaciones pueden implementar uno o varios módulos al mismo tiempo y realizar seguimiento para la implementación de otros módulos proyectados a futuro. (DeepInder, 2004)

Los sistemas ERP pretenden garantizar el éxito y la planificación de los negocios, en aras de obtener beneficios en su implementación ya sea de forma operacional o estratégica. La evidencia de estudios realizados en la actualidad, indica que contar con un sistema ERP mejora la información en las entidades y la integración de operaciones en general. Las organizaciones tienden a tener un alto desarrollo financiero con respecto al valor del mercado que aquellas que no han invertido en este tipo de sistemas. (Shang, 2000)

El sistema ERP, denominado “Cedrux”, es una solución informática de software, que se construye especialmente adaptado al control económico en las condiciones actuales del país.

Cedrux, presenta varios subsistemas, entre los cuales se encuentra el de Inventario, encargado de llevar la gestión de los recursos materiales en las entidades cubanas. Este subsistema tiene entre sus procesos la recepción y la transferencia entre áreas, que junto al proceso de facturación del subsistema Facturación, están centrados la mayoría de los procesos del control de las Reservas Materiales.

La gestión de información del proceso para las Reservas Materiales como parte de los procesos de control de inventarios, se dificulta en el momento de realizar operaciones contables que se necesitan para su control y almacenamiento. Constituye una estrategia tomada por el país con el fin de lograr un equilibrio de la economía en caso de existir una situación excepcional. Debido a lo anteriormente planteado, la información en cuanto al tema de las Reservas es escasa, puesto que es confidencial. Solo puede ser accedida por el INRE y el especialista de la Reserva, de esta forma se evita los posibles riesgos de que el enemigo tenga conocimiento de su existencia y pueda acometer acciones que perjudiquen su correcto almacenamiento.

Por la importancia que tienen estas Reservas, se realizó un análisis de las actividades llevadas a cabo en éste proceso, lo que reveló un grupo de deficiencias. Estas surgieron a través de una entrevista realizada al Funcional asignado a la Línea de Logística, debido a la poca bibliografía existente sobre el tema. (Ver Anexo 1). El análisis arrojó los siguientes resultados:

- La mayoría de los procesos y subprocesos de la Reserva no están informatizados, por lo que no hay una opción que brinde este servicio. Debido a lo anteriormente planteado, se necesita un estudio para llevar a cabo cualquier operación.
- La reserva estatal y la reserva FAR (Fuerzas Armadas Revolucionarias), son áreas dentro del mismo almacén de inventario corriente, lo que genera un inadecuado tratamiento de las operaciones contables.

- Se dificulta el control de los niveles de acumulación y de la rotación de lotes, posibilitando que ocurran vencimientos de los mismos.
- Se transcriben todas las facturas y modelos generados a un modelo oficial de forma manual, atentando de esta forma la integridad y disponibilidad de estos datos.
- El especialista debe tener un conocimiento previo y una basta experiencia, debido que los procesos necesitan de una ardua investigación para su desempeño.
- Se dificulta el control de los niveles de acumulación y de la rotación de lote, posibilitando que ocurran vencimientos en los mismos.

La situación problemática anteriormente planteada, consta como **problema a resolver**: Carencia de una adecuada gestión del control de las Reservas Materiales.

Para darle solución al problema existente se propone como **objeto de estudio**: Los procesos para el control de inventarios.

Teniendo en cuenta el problema a resolver se define como **objetivo general** del trabajo de diploma: Desarrollar la solución informática para el control del tratamiento de las Reservas Materiales como parte de los procesos de control de inventario.

Para lograr el objetivo general se proponen los siguientes **objetivos específicos**:

- Determinar el marco teórico de la investigación para el control de los procesos del tratamiento en las Reservas Materiales.
- Analizar, diseñar e implementar la solución del sistema para el control del tratamiento de la Reserva como parte del sistema de control de inventario.
- Validar la solución propuesta.

Por consiguiente, el **campo de acción** de este trabajo está enmarcado en: los procesos para el control de las Reservas Materiales.

Se plantea como **idea a defender**: El desarrollo de la solución para el control del tratamiento de las Reservas Materiales como parte de los procesos de control de inventarios de Cedrux permitirá una adecuada gestión del control de las Reservas Materiales.

Para lograr los objetivos específicos de este trabajo, se propone un desglose de las **tareas de investigación**:

- Análisis del comportamiento de control de los procesos del tratamiento de las Reservas Materiales a nivel mundial.
- Caracterización de las tecnologías, lenguajes y herramientas propuestas para el desarrollo de la aplicación.

- Análisis de la solución del sistema para el control del tratamiento de las Reservas Materiales como parte del sistema de control de inventario.
- Diseño del sistema para el control del tratamiento de las Reservas Materiales.
- Evaluación de los componentes existentes en el sistema de control de inventario que pueden reutilizarse.
- Implementación de las interfaces de los componentes como parte del sistema de control de inventario.
- Implementación de la capa de negocio como parte del sistema de control de inventario.
- Implementación de la capa de acceso a datos como parte del sistema de control de inventario.
- Implementación de las validaciones y excepciones de los componentes del sistema de control de la Reserva como parte del sistema de control de inventario.
- Implementación de los servicios incluidos dentro de sus responsabilidades que se necesiten para la implementación de otros módulos.
- Validación del diseño de los componentes obtenidos de la solución a través de métricas.
- Realización de pruebas de unidad a los componentes obtenidos.

Durante el desarrollo de la presente investigación se emplearon diferentes procedimientos que permitieron dar cumplimiento al objetivo propuesto. Se utilizaron **métodos teóricos** como el **Analítico - Sintético** que permitió analizar las características principales del proceso para el tratamiento en las Reservas Materiales y definir las relaciones que hay entre ellas, logrando sintetizar los elementos más importantes que se relacionan y el **Histórico Lógico** que posibilitó realizar un estudio a nivel nacional e internacional de sistemas similares al que se quiere implementar. Sustentó además el desarrollo de este trabajo el **método empírico Entrevista**, utilizado para la descripción del negocio asociado al proceso de control de Reservas Materiales, la obtención de los requisitos funcionales de acuerdo al negocio descrito y el diseño de interfaces teniendo en cuenta las exigencias del cliente (Ver Anexos 1, 2 y 3).

Este documento está estructurado en cuatro capítulos:

Capítulo 1: **Fundamentación teórica**. En este capítulo se presentan temas que integran la fundamentación teórica de la investigación a realizar, se hace necesario el análisis y revisión de las fuentes bibliográficas relacionadas con el tema en cuestión. Se realiza un análisis sobre las principales tecnologías, metodologías y herramientas que se utilizan en la actualidad para este tipo de aplicación.

Capítulo 2: **Características del sistema**. En este capítulo se describen las principales características que debe tener el sistema, mediante el modelamiento de los procesos de negocio y la definición de los requisitos de software con los que debe cumplir el sistema. Se expone una breve descripción de la solución propuesta.

Capítulo 3: **Análisis y diseño del sistema.** En este capítulo se desarrollan parte de los artefactos propuestos en la fase de Modelación definida en la metodología de software aplicada. Se realiza el diseño de los componentes a implementar y se hace referencia a la arquitectura que sustenta los módulos de software. Culmina la fase de Modelación, obteniendo los artefactos restantes de la fase. Se abarca todo lo concerniente con la implementación del sistema.

Capítulo 4: **Prueba y Validación.** Se realizan pruebas a la aplicación, en aras de garantizar su calidad, aplicando un grupo de métricas de validación del diseño e implementación. Se utilizan los instrumentos de evaluación de las métricas: Tamaño Operacional de la Clase (TOC), Relaciones entre Clases (RC), Profundidad de Herencia (PH) y Número de Operaciones Redefinidas en las clases hijas (NOR).

Este documento posee, además, Introducción, Conclusiones parciales para cada capítulo, Conclusiones generales, Recomendaciones, Bibliografía, Referencias bibliográficas, Glosario de términos y Anexos.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En el presente capítulo se realizará una descripción de los conceptos asociados a las Reservas Materiales. Se hará referencia a las soluciones informáticas a nivel nacional e internacional, que dentro de sus procesos brinden alguna opción que permita el tratamiento de las Reservas Materiales. Además se exponen diferentes procedimientos que posibilitarán el correcto control del proceso de Reservas Materiales y se explicarán las características de las herramientas, tecnologías y metodologías utilizadas para la creación de la aplicación.

1.2 Reserva

En la vida diaria se pueden dar casos excepcionales, tal es el caso de los fenómenos naturales, los cuales son muy frecuentes por la ubicación que tiene Cuba, otra situación pudiera ser, el brote de una epidemia o simplemente la guerra. Para esto cada país debe estar preparado y tratar de tener planificados algunos materiales, para mantener el equilibrio de la economía en estos casos. Para lograr este equilibrio, es de vital importancia que en cada entidad cubana que lleve a cabo el control de las Reservas exista un stock¹ de reservas materiales.

Según el Artículo 2 del Reglamento del Sistema de Reservas Materiales 23/06/2006, “El Estado cubano dispone la creación de las Reservas Materiales para garantizar el desarrollo y normal funcionamiento de la economía, así como durante las situaciones excepcionales, la vida de la población, mantener y elevar la capacidad de resistencia del país, asegurar la realización de las acciones combativas, mantener la seguridad y el orden interior, prevenir y restablecer las consecuencias derivadas de desastres y continuar la actividad económico-productiva.” (Ministro, 2006)

Las **Reservas Materiales** constituyen los productos y bienes reservados para garantizar el desarrollo y normal funcionamiento de la economía durante situaciones excepcionales.

De acuerdo con su destino las Reservas Materiales se integran por:

- **Reserva Movilizativa:** Constituye el conjunto de recursos materiales acumulados por los órganos y organismos estatales, las entidades económicas e instituciones sociales, destinados a asegurar la continuidad de la producción y los servicios y la satisfacción de las demandas de la lucha armada y la economía, las necesidades de la población, además de las que aseguren el cumplimiento de misiones de la Seguridad del Estado y el Orden Interior durante el estado de Guerra o la Guerra.

¹ **Stock** se usa con el sentido de existencias o reserva de alguna cosa disponible para un uso futuro. (Diccionario, 2008)

- **Reserva Estatal:** Se compone por recursos materiales que tienen el objetivo de garantizar el desarrollo y normal funcionamiento de la economía nacional, atenuar dificultades imprevistas por desastres naturales, fortalecer la capacidad defensiva del país y brindar ayuda internacionalista.
- **Reserva FAR:** Es creada por las Fuerzas Armadas Revolucionarias las cuales determinan su designación, composición, escalonamiento, normas, ubicación y facultades para su empleo, sobre la base de las necesidades de la lucha armada y en correspondencia con las misiones planteadas a las unidades regulares y a las Milicias de Tropas Territoriales. (Ministro, 2006)

1.3 Soluciones informáticas que realizan el proceso de Reserva

Los ERP, son programas de gestión administrativa con herramientas de planificación. Estos cumplen con la rutina comercial diaria de la empresa y la de proveer mecanismos analíticos y proyectivos, de manera que la dirección pueda establecer políticas y observar su comportamiento en el tiempo. A su vez, la multiplicidad de funciones y la flexible capacidad de configuración los convierten en sistemas administrativos centrales y únicos, donde el personal registrará todas las operaciones de gestión sin necesidad de herramientas informáticas adicionales. (Isis., 2010) El tratamiento de la Reserva, se encuentra presente en estos sistemas de planificación de recursos empresariales.

Es importante que una entidad tenga en cuenta varios aspectos a la hora de tomar la decisión de elegir un ERP, entre esos aspectos se encuentran:

- Ponderar la relación costo-beneficio, pues no es factible que el ERP elegido cause más gastos que los beneficios que aporta.
- Tratar de no utilizar un software propietario, para lograr la independencia tecnológica.
- Reducir la necesidad de introducir datos manualmente, eliminando posibles fuentes de errores.

En el siguiente acápite se hace referencia a las distintas soluciones informáticas en el ámbito nacional e internacional, teniendo en cuenta que cada país tiene sus propias especificaciones en cuanto al control de las Reservas Materiales, debido a que constituyen estrategias tomadas para fomentar la economía en caso de que ocurra una situación excepcional. Es importante aclarar, que existe escasa bibliografía a consultar en cuanto al tema de las Reservas Materiales, porque sus procesos constituyen un tema delicado y con un alto nivel de confidencialidad. Esto se debe a la seguridad que conviene tener sobre las mismas, evitando posibles acciones que profanen su correcto tratamiento, almacenamiento y control. (Ver Anexo 2)

1.3.1 Soluciones informáticas Internacionales

El **ERP Manager** brinda diferentes prestaciones acorde a las diferentes necesidades y presupuestos de cada empresa. Cuenta con depósitos o almacenes con identificación de "tipo" - inventario, control de

calidad, reserva y producción, siendo esta una de sus generalidades. En la producción ingresa órdenes de productos de manera manual o automáticamente desde el plan de producción con reserva de materiales. Utiliza tecnología de base de datos Microsoft SQL Server (Isis, 2010). Siendo esta una de las desventajas que presenta el sistema, pues Microsoft SQL SERVER consume una enorme cantidad de memoria RAM para la instalación y utilización del software.

El MRP (software de gestión industrial), es un software integrado al ERP Manager, permitiendo administrar materias primas, materiales, mano de obra, chequear procesos, establecer fórmulas de producto, generar planes de producción, asignar costos y evaluar requerimientos de equipos, personas y artículos para cumplir con sus metas industriales. Sus avanzadas herramientas de trazabilidad lo convierten en un sistema indispensable para empresas certificadas bajo normas de calidad ISO². Puede venderse solo o acompañado de versiones mayores de ERP Manager. Algunos de los elementos que integran el software MRP de gestión de costos industriales son:

- Plan de producción: Integrado por el plan maestro de producción (MPS - Master Production Schedule) que calcula, en base a los pedidos de los clientes y los pronósticos de demanda (tanto externa como interna - p.ej. generar stock de semielaborados a utilizar en producciones futuras), cuáles serán los productos y en qué cantidades deberán fabricarse para satisfacer las necesidades. El plan maestro indica también las fechas en que deberá estar terminada la producción para cumplir con el objetivo. El plan de producción se actualiza periódicamente mediante minutas de avance del proceso y genera los comprobantes de reserva y retiro de materiales de los respectivos depósitos.
- Seguimiento y control de la gestión de producción: Emite órdenes de producción con reserva de materiales o utilización directa de los mismos a través de vales de consumo
- En cuanto a la consulta de órdenes de producción se tienen: Materiales reservados por órdenes de producción, vales de consumo asignados y producciones parciales asociadas. (Isis, 2010).

El **ERP Softland** Chileno es un software, que lleva a cabo en el módulo de Inventario y Facturación, el control de Reservas, Consignaciones y Consumos, gestionando un control de los productos en reserva, ya sea de los clientes o internos de la empresa para centros de costos, órdenes de trabajo, entre otros. Para administrar las bodegas, clasifica con diferentes conceptos los movimientos de entrada y salida. Por ejemplo, controla las entradas y salidas de productos en reservas (por cliente, Orden de Trabajo, Orden de Producción), productos en consignación, los despachos por facturar, las ventas efectuadas y por despachar y los consumos internos hacia los Centros de Costo. (Softland, 2009). Sus productos, servicios y soporte de alta calidad están orientados a satisfacer las necesidades locales a más de 35.000 clientes

² ISO: International Standardization Organization

en 19 países de Latinoamérica. Presenta como desventaja su trabajo sobre base de datos SQL Server de Microsoft y el número de usuarios concurrentes a la base de datos está limitado. (Softland, 2009)

1.3.2. Solución informática nacional

De acuerdo a la entrevista realizada al Funcional Jorge Reyes Estévez, asignado a la línea de Logística (Ver Anexo 2), Cuba cuenta con el **Stock Empresarial Mistral Caribe**, que es el sistema que le brinda cierto tratamiento a la Reserva. Este sistema se caracteriza por ser una aplicación de escritorio, que muestra listados con la información de los recursos de la Reserva que se encuentran almacenados, posibilitándole al especialista la toma de decisiones. No informatiza ningún proceso de la Reserva, estos se realizan de forma manual, atentando con la integridad y persistencia de los datos.

El objetivo fundamental del Stock Empresarial Mistral Caribe es la disminución de los niveles de stock para reducir los recursos inmovilizados e incrementar la liquidez en la empresa. Para entidades con alcance nacional, posee herramientas de comunicaciones que crean en tiempo real, almacenes virtuales a diferentes niveles, con las existencias y movimientos de sus dependencias, permitiendo organizar el proceso de compras, distribución y pedidos. La consulta de esta información se realiza sobre sitios web ajustándose a los procedimientos internos de cada cliente. (S.A., 2011)

Hasta el momento ésta es la única solución que se conoce con respecto al control de Reservas, a causa de la escasa bibliografía existente sobre el tema y a la alta confidencialidad que tienen las mismas. Por este nivel de confidencialidad el enemigo no ha podido atentar contra el orden interior durante la ocurrencia de un caso excepcional en el transcurso de los años.

1.4 Valoración del estado del arte

Después del estudio realizado a algunos sistemas internacionales, se pudo comprobar que presentan inconvenientes teniendo en cuenta la utilización de software propietario y de algunas herramientas privativas, por este motivo su uso no es recomendable. La economía cubana necesita un software que cumpla con las expectativas de lograr la independencia tecnológica, que posibilite el correcto control a los procesos de las Reservas Materiales y que sea adaptable a cada entidad que lleve a cabo los procesos antes mencionados. A pesar de que en el país se utiliza el software Stock Empresarial Mistral Caribe con funcionalidades que le dan cierto tratamiento a los procesos de la Reserva, no informatiza algunos como es el caso de la rotación de lotes, solicitud de cambio de surtido, pago de deudas, entre otros.

Por el rigor y responsabilidad que debe tener este proceso, se decidió vincularlo al desarrollo de Cedrux. De esta forma se agruparían todos los procesos y subprocesos que no están informatizados, permitiendo la gestión de los materiales de forma uniforme. También se aprovecharía la experiencia de los procesos de Inventario y Facturación que ya están implementados en la Gestión de Inventario.

1.5 Modelo de desarrollo

La dirección del equipo de trabajo ERP-Cuba, para la construcción del Sistema Integral Cedrux, decidió utilizar el modelo de desarrollo propuesto por la UCID (Unidad de Compatibilización, Integración y Desarrollo de Software para la Defensa), basándose en los resultados obtenidos después de haber aplicado este modelo. La línea de Logística, que es la encargada del desarrollo del componente de Inventario, donde se pretende ubicar la solución del presente trabajo de diploma, como parte de la solución de Cedrux, también se acoge al modelo de desarrollo propuesto.

La descripción de éste modelo de desarrollo, se encuentra en el documento titulado: Proceso de desarrollo y Gestión de Proyectos de Software. A través del cual se describe la secuencia de actividades de alto nivel para la construcción y desarrollo de soluciones. Lográndolo con la combinación entre los modelos basado en Componentes, el Iterativo y el Incremental. Se emplearán las técnicas de prototipado, con el objetivo de desarrollar una definición mejorada de los requisitos del usuario para el sistema. (Unidad de Compatibilización, 2009)

El desarrollo iterativo e incremental, es un enfoque que está compuesto por iteraciones, cuyo objetivo es entregar una parte del sistema parcialmente completo, probado, integrado y estable. Todo el software es integrado en cada entrega de cada iteración hasta obtener el producto de software completo en la última iteración. En cada iteración se obtiene como resultado un incremento.

El desarrollo basado en componentes permite alcanzar un mayor nivel de reutilización de software, posibilita que las pruebas sean ejecutadas, probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados. Cuando existe un débil acoplamiento entre componentes, el desarrollador es libre de actualizar y/o agregar componentes según sea necesario, sin afectar otras partes del sistema. Dado que un componente puede ser construido y luego mejorado continuamente, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo. (Unidad de Compatibilización, 2009)

Dentro de los principales procesos definidos por el modelo de desarrollo antes mencionado se encuentran:

- Planificación.
- Gestión de los Riesgos del Proyecto.
- Gestión del Equipo de Desarrollo.
- Gestión de la Calidad.
- Gestión de la configuración.

El ciclo de vida de Cedrux, basado en el modelo de desarrollo propuesto se basa en cinco fases secuenciales:

- Inicio.

- Modelación.
- Construcción.
- Explotación.
- Experimental.
- Despliegue.

Estas fases, permiten mejorar el planeamiento, ejecución y control del proyecto, se caracterizan por la conclusión y la aprobación de uno o más artefactos entregables siendo estos productos del trabajo, mensurables y verificables, correspondiendo algunos a los procesos de gestión de proyectos y otros al producto. (Unidad de Compatibilización, 2009)

1.6 Marco de trabajo

Sentadas las bases para dar la solución del presente trabajo, se decidió utilizar el marco de trabajo Sauxe, desarrollado en la UCI (Universidad de Ciencias Informáticas) por el equipo de Arquitectura. Este framework tiene su fundamento en el uso de tecnologías libres, entre las que se encuentran el lenguaje de programación PHP, el Gestor de Base de Datos Postgres SQL entre otros, de esta forma se cumplen con las expectativas que tiene el país de lograr la independencia tecnológica. Se basa en el uso de otros marcos de trabajo, como es el caso del ExtJS para la presentación, el Zend para la lógica de negocio y el Doctrine para el acceso a datos. Cuenta con una arquitectura en capas que a su vez presenta en su capa superior un MVC³. (Gómez Baryolo, y otros, 2010) Este marco de trabajo es el que se empleará para el desarrollo de la solución que se propone, teniendo en cuenta la idea de lograr la independencia tecnológica que necesita el país, utilizando tecnologías libres y herramientas que se mostrarán a continuación.

1.6.1 Herramientas

Mozilla Firefox

El Mozilla Firefox se utilizará como navegador web Mozilla Firefox 3.6.10. Es un navegador de código abierto, multiplataforma, con interfaz gráfica de usuario desarrollado por la Fundación Mozilla Firefox y cientos de voluntarios. Se basa en el motor XULRunner, desarrollado en su mayor parte utilizando el lenguaje XUL. Se ha convertido en el principal foco de desarrollo de dicha Fundación, incorporando bloqueo de ventanas emergentes, navegación por pestañas, marcadores dinámicos, soporte para estándares abiertos, búsqueda "Arrastra y suelta", gestor de descargas y un mecanismo para añadir funcionalidades mediante extensiones. (Firefox, 2009)

³ **(MVC).Modelo Vista Controlador** es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Tortoise SVN

Como sistema de control de versiones se utilizará el Tortoise SVN (Subversion) en su versión 1.6.5. Esta herramienta de entorno colaborativo, es un cliente gratuito de código abierto que maneja ficheros y directorios almacenados en un repositorio que recuerda todos los cambios que se hayan escrito en él. Permite obtener versiones antiguas de documentos para saber cuándo y cómo fue la última vez que se realizó el cambio y por quién fue realizado, entre las características que posee se encuentra su integración con el shell de Windows: que permitirá continuar trabajando con las herramientas que se conocen sin necesidad de instalar aplicaciones nuevas cuando requieran las funciones del control de versiones. Mediante los iconos de las carpetas se puede saber en qué estado se encuentra la información. Posee fácil acceso a los comandos de Subversion añadiendo su propio submenú al menú del explorador, posibilitando el acceso a los comandos de *Subversion*. (Subversion, 2004)

Sistema Gestor de Base de Datos (SGBD)

Un SGBD es un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Tienen el propósito de manejar de manera clara, sencilla y ordenada grandes volúmenes de datos que posteriormente se convertirán en información relevante para una organización.

Los SGBD deben hacerse transparente al usuario detalles acerca del almacenamiento físico de la información, así como ser capaces de evitar la duplicidad de registros y otorgar permisos a los usuarios y grupos de usuarios, con el fin de garantizar la seguridad de la información, además de ofrecer el menor tiempo de respuesta cuando el usuario la solicita. (datos, 2004)

Como Sistema Gestor de Base de Datos se empleará el **PostgreSQL** en su versión 8.3.3. Esta herramienta es un motor y servidor de base de datos relacional libre, liberado bajo la licencia BSD⁴. Es un sistema de gestión de base de datos objeto-relacional para múltiples plataformas. Posee una estabilidad y confiabilidad legendaria, nunca ha presentado caídas en varios años de operación de alta actividad. Presenta un buen soporte para *subselects*, *triggers*, vistas y procedimientos almacenados en el servidor, además tiene ciertas características orientadas a objetos. (PostgreSQL, 2011)

1.6.2 Tecnologías

ExtJS

El ExtJS es un framework que se utilizará para el desarrollo de la capa de presentación. Constituye un marco de trabajo completo y avanzado, que está basado completamente a la programación Orientada a

⁴ La **licencia BSD**: Es una licencia de software libre permisiva como la licencia de OpenSSL o la MIT License. Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

Objetos, permitirá usar librerías JavaScript para sus funcionalidades, permitiendo hacer un modelo de componentes, modelo de contenedores, capas, tablas, plantillas y vistas de datos. Presentando un juego de interfaces visuales agradables a la vista del usuario haciendo uso de Ajax⁵ a través de XML⁶ y es capaz de acoplarse fácilmente a los lenguajes de comunicación con la base de datos tales como PHP, Python, MySQL, entre otros.

Esta librería incluye:

- Componentes UI del alto performance y personalizables.
- Modelo de componentes extensibles.
- Un API fácil de usar.
- Licencias Open Source y comerciales.

Entre los componentes que esta librería ofrece se encuentran cuadros de diálogo, menús, tablas editables, layouts, paneles, pestañas y todo lo necesario para construir atractivos desarrollos al estilo de Web 2.0. Una de las grandes ventajas de utilizar ExtJS es que permite crear aplicaciones complejas utilizando componentes predefinidos así como un manejador de layouts similar al que provee Java Swing⁷. (Frederick, y otros, 2009)

Zend framework

El Zend framework se utilizará para la lógica del negocio. Es un marco de trabajo de código abierto para desarrollar aplicaciones web y servicios web con PHP5, utilizando el estilo MVC como base de su funcionamiento. Es fácilmente integrable a las aplicaciones debido a su composición y a que contiene diferentes clases de gran utilidad, como por ejemplo en la búsqueda dinámica de ficheros a incluir o utilizar. Brinda soluciones para construir sitios web modernos, robustos y seguros.

Aunque se pueden utilizar de forma individual, los componentes de la biblioteca estándar de Zend Framework conforman un potente y extensible *framework* de aplicaciones web al combinarse y ofrece un gran rendimiento y una robusta implementación MVC, una abstracción de base de datos fácil de usar, y un componente de formularios que implementa la prestación de formularios HTML, validación y filtrado para que los desarrolladores puedan consolidar todas las operaciones usando de una manera sencilla la interfaz orientada a objetos. (Esser, 2009)

⁵ **Ajax:** Acrónimo de Asynchronous JavaScript And XML (JavaScript y XML asíncronos), es una técnica de desarrollo web para crear aplicaciones interactivas.

⁶ **XML:** es un Lenguaje de Etiquetado Extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Es un formato que permite la lectura de datos a través de diferentes aplicaciones.

⁷ **Java Swing:** Paquete java para la generación de GUI (en español: interfaz gráfica de usuario) en aplicaciones reales de gran tamaño.

Doctrine

El Doctrine es el marco de trabajo que se empleará para la capa de acceso a datos. Es un mapeador de objetos relacionales (ORM) para PHP 5.2.3 que se basa en una potente capa de abstracción de bases de datos (DBAL)⁸. Un ORM u (Object Relation Mapper) es una técnica de programación que permite convertir datos entre el sistema de tipos utilizados en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, es decir, las tablas de la base de datos pasan a ser clases y los registros objetos que se pueden manejar con facilidad.

Una de sus principales características es la opción de escribir las consultas de base de datos en un objeto de propiedad orientada dialecto SQL⁹ llamada Doctrine Query Language (DQL), inspirado en Hibernate HQL. Esto proporciona a los desarrolladores una poderosa alternativa a SQL que mantiene la flexibilidad, sin necesidad de la duplicación de código innecesario. (Smith, 2009)

1.7 Lenguajes de modelado y desarrollo

Los lenguajes de modelado y desarrollo que se proponen, forman parte de los lenguajes establecidos por parte de la dirección del Proyecto ERP-Cuba para el desarrollo de Cedrux, van aparejados a las herramientas, tecnologías y marco de trabajo definido anteriormente.

1.7.1 Lenguajes de modelado

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software. Se emplearán para el desarrollo de la solución que se propone los siguientes lenguajes de modelado:

UML

El Lenguaje Unificado de Modelado o UML por sus siglas en inglés, es un lenguaje que permitirá modelar, construir y documentar los elementos que forman un sistema de software orientado a objetos. Es un lenguaje de modelado de propósito general que proporcionará una gran flexibilidad y expresividad a la hora de modelar sistemas. Implementará un lenguaje de modelado y una documentación común para todos los desarrolladores, independientemente del lenguaje utilizado para el desarrollo. Dividirá cada proyecto en un número de diagramas que representan las diferentes vistas del proyecto. Estos diagramas juntos son los que representa la arquitectura del proyecto. UML ayudará al usuario a avizorar los resultados antes de invertir en herramientas innecesarias, reduciendo el coste y el tiempo empleado en la construcción de las piezas que constituirán el modelo. (Ferré Grau, y otros, 2009)

⁸ **DBAL**: Data Base Abstraction Layer.

⁹ **SQL**: Lenguaje de consulta estructurado utilizado para acceder a bases de datos relacionales que permite realizar varias operaciones sobre estas.

BPMN

La Notación para el Modelado de Procesos de Negocio o BPMN por sus siglas en inglés es una notación común para cerrar la brecha de comunicación que frecuentemente se presenta entre el diseño de los procesos de negocio y la implementación de los mismos. Es un estándar de modelado de procesos de negocio, en donde se presentarán gráficamente las diferentes etapas del proceso del mismo. La notación ha sido diseñada específicamente para coordinar la secuencia de procesos y los mensajes que fluyen entre los diferentes procesos participantes.

BPMN se considera un estándar internacional muy utilizado, es independiente de cualquier metodología de modelado de procesos, crea un puente estandarizado para disminuir la brecha entre los procesos de negocio y la implementación de estos. Permitirá modelar los procesos de forma unificada y estandarizada. Su principal objetivo será proveer una notación estándar facilitando el entendimiento de las colaboraciones y transacciones de negocio para todos los involucrados e interesados del negocio. Entre estos interesados están los analistas de negocio (quienes definen y redefinen los procesos), los desarrolladores técnicos (responsables de implementar los procesos) y los gerentes y administradores del negocio (quienes monitorean y gestionan los procesos). (Bizagi, 2006)

1.8 Herramienta de diseño

Visual Paradigm

Se utilizará el Visual Paradigm como herramienta de modelado, que aunque es propietaria, la UCI a la cual pertenece el Cedrux, que es donde se pretende desarrollar la solución de este trabajo, posee una licencia para el tratamiento con la misma. Esta herramienta ofrece un entorno de creación de diagramas para UML y al mismo tiempo por sus características particulares en la modelación de procesos de negocio: Diagramas de Procesos de Negocio – Proceso, enfocado al negocio que generan un software de mayor calidad. Posee el uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación; capacidades de ingeniería directa (versión profesional) e inversa; modelo y código que permanece sincronizado en todo el ciclo de desarrollo; disponibilidad en múltiples plataformas y de múltiples versiones, para cada necesidad. Presenta licencia gratuita cuando es usada para el sistema operativo Linux.

Presenta un diseño centrado en casos de uso y proporciona a los desarrolladores de software una interfaz simple y amigable, con muchas opciones tales como: diversidad de idiomas, generación de código para varios lenguajes de programación posee facilidad para la instalación y actualización, así como compatibilidad entre sus ediciones. (Paradigm, 2005)

1.9 Patrones de diseño

Para el diseño se utilizarán patrones, que en términos generales se puede afirmar que un patrón es un cúmulo de información que proporciona respuesta a un conjunto de problemas similares en un contexto dado. Los patrones hacen la producción de software más resistente al cambio, establecen parejas problema-solución, ayudan a especificar interfaces, facilitan la reutilización del código y permiten una fácil comprensión debido a la documentación estándar que presentan. (Patrones, 2009)

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Dentro de estos patrones se encuentran:

- Los patrones **GRASP**: Patrones de Software para la Asignación General de Responsabilidad. En este caso las responsabilidades están relacionadas con las obligaciones de un objeto en cuanto a su comportamiento:

Conocer: Conocer los datos privados encapsulados, conocer los objetos relacionados, conocer las cosas que puede derivar o calcular.

Hacer: Hacer algo él mismo, como crear un objeto o hacer un cálculo, iniciar una acción en otros objetos, controlar y coordinar actividades en otros objetos. (Hurtado Bustamante, y otros, 2009)

GRASP destaca 5 patrones principales: Experto, Creador, Alta cohesión, Bajo acoplamiento, Controlador.

Experto: Este patrón es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para ejecutar la tarea. Refuerza el encapsulamiento y esto redundante en bajo acoplamiento. Está presente en la definición de las clases según las funcionalidades que realizan. (Hurtado Bustamante, y otros, 2009)

Creador: El patrón creador ayuda a identificar quién debe ser el responsable de la instanciación o creación de nuevas clases u objetos. La clase podrá crear la nueva instancia si y sólo si tiene en cuenta al menos uno de los siguientes criterios:

- Tiene la información necesaria.
- Usa directamente las instancias creadas del objeto.
- Almacena o maneja varias instancias de la clase.
- Contiene o agrega la clase.

La visibilidad entre la clase creada y la clase creadora es una de las facilidades que se deriva del uso de patrón y además conduce a un bajo acoplamiento, lo cual supone facilidad de mantenimiento y reutilización así como mayor claridad. (Hurtado Bustamante, y otros, 2009)

Alta cohesión: Indica que la información que almacena una clase debe ser coherente, de manera que todos sus métodos tengan un comportamiento bien definido.

- Las Clases se pueden reutilizar con mayor facilidad y flexibilidad.
- Una Clase con baja cohesión hace muchas cosas no relacionadas.
- Una Clase con alta cohesión hace lo que uno podría esperar que hiciera.

Si el sistema fallara por alguna razón es mucho más fácil encontrar responsabilidades si las Clases del sistema son cohesivas. (Hurtado Bustamante, y otros, 2009)

Bajo acoplamiento: Patrón evaluativo que asigna responsabilidades de modo que se mantenga un engranaje pobre entre las clases y objetos, reduce el impacto de los cambios y aumento de reutilización. (Hurtado Bustamante, y otros, 2009)

Controlador: Asigna la responsabilidad a una clase de manejar mensajes correspondientes a eventos en un sistema. Esta clase será la encargada de recibir los datos del usuario y enviarlos a las distintas clases según el método llamado a modo de intermediario entre una determinada interfaz y el algoritmo que la implementa. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación para aumentar la reutilización de código y a la vez tener un mayor control. (Hurtado Bustamante, y otros, 2009)

- Los patrones **GOF (Gang Of Four)**: Son 23 patrones que describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Están clasificados en patrones creacionales, estructurales y de comportamiento.

Los patrones creacionales se encargan de la creación de los objetos ayudando a que el sistema sea independiente de la creación, composición y representación de los objetos. Los patrones estructurales son los encargados de cómo las clases y objetos están compuestos para formar estructuras más grandes. Usan la herencia para componer interfaces u objetos en tiempo de ejecución. Los patrones de comportamiento plantean algoritmos y la asignación de responsabilidades entre objetos. Estos patrones no solo describen clases y objetos sino también describen la comunicación entre ellos. En la aplicación se utilizaron los siguientes patrones: (Olivares Rojas, 2009)

Fachada: Patrón estructural que permite proveer una interfaz unificada y sencilla como intermediaria entre un cliente y una interfaz o grupo de interfaces más complejas. Es utilizado en varios de los diagramas de clases realizados en el diseño. Un ejemplo de su uso es en los servicios, donde la relación existente entre las clases controladoras y los servicios permite acceder a métodos que no están implementados en el componente Inventario y que se encuentran, tanto en otros componentes pertenecientes al subsistema Logística, como en otros subsistemas externos. La interfaz utilizada como fachada sería el IoC (en inglés: Inversion of Control) que se explicará posteriormente. (Olivares Rojas, 2009)

Mediador: Patrón de comportamiento que coordina las relaciones entre sus asociados. Permite la interacción de varios objetos, sin generar acoples fuertes en esas relaciones. Es utilizado en la mayoría de los diagramas de clases siendo aplicado en el tratamiento de excepciones. (Olivares Rojas, 2009)

1.10 Técnicas para la captura de requisitos

A través de los años se ha optado por desarrollar algunas técnicas con el objetivo de ayudar a mejorar el levantamiento de información. Estas son manejadas por un equipo de clientes y desarrolladores que trabajan juntos para comprender el problema, proponer soluciones, negociar diferentes perspectivas o puntos de vista y especificar un conjunto básico de requisitos de la solución.

Para la captura de requisitos existen varias técnicas, las aplicadas en este trabajo de diploma son las explicadas a continuación:

Entrevista: se realizan entrevistas con el cliente (en este caso los funcionales) para tener una visión clara de cada funcionalidad que debe tener el sistema. La meta de una entrevista es enterarse de opiniones, sentimientos acerca del sistema, la organización y metas del sistema y personales del cliente. (Koch y otros, 2002)

Elaboración de prototipos: mediante la aplicación de esta técnica se procede a la generación de un “casarón” que presenta la forma de interfaz y simula algunas funcionalidades. Este método es útil porque permite observar las reacciones del usuario y sus habilidades. Se pueden incorporar sugerencias e innovaciones antes de entregar una primera versión completa. El sistema resultante será más cercano al usuario.

Lluvia o Tormenta de ideas: se realiza una reunión de varios interesados en la que todos ellos expresan sus ideas sobre el problema y su solución. (Zapata, y otros, 2004)

Talleres: consiste en la realización de reuniones con los involucrados en el desarrollo del sistema, pero estas son sesiones más intensivas y estructuradas donde se realiza una preparación previa y es dirigida por un experto (en este caso el funcional o jefe de la factoría de PDO). Su objetivo fundamental es detallar cada requisito y sirve como base fundamental para la posterior especificación de los requisitos.

La Revisión Técnica Formal (RTF)

Son reuniones del personal técnico (usuario final del sistema) con el objetivo de validar la especificación de requisitos. Su aplicación a los documentos de práctica permitirá detectar deficiencias, ambigüedades, omisiones, errores, etc., tanto de formato como de contenido. Suelen ser realizadas entre 3 y 5 personas y los especialistas funcionales deben ser independientes del equipo que ha realizado la especificación de requisitos.

Cada Revisión Técnica Formal (RTF) se lleva a cabo mediante una reunión y solo tiene éxito si esta es bien planificada, controlada y atendida.

1.11 Conclusiones parciales

En el presente capítulo se describieron los tipos de Reservas existentes, así como los conceptos asociados a los mismos. Se realizó un estado del arte a partir de las soluciones informáticas existentes en el ámbito nacional e internacional que dentro de sus procesos incluyen las Reservas Materiales, evidenciando la no existencia de un software que realice un control al proceso de la Reserva, de acuerdo con las especificaciones de las entidades cubanas. Por último se describieron las herramientas, tecnologías y el proceso de desarrollo propuesto por la dirección de ERP-Cuba, para el desarrollo de Cedrux.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En el presente capítulo se realizará un análisis de los procesos de la Reserva de Recursos Materiales para facilitar una mayor comprensión de los mismos y describir el negocio. Se conocerán los informes que se manejan y la propuesta del sistema, así como el modelado del negocio, presentando los artefactos resultantes correspondientes a este flujo, definido por el modelo de desarrollo utilizado. Para el mejor funcionamiento del sistema propuesto y desarrollo de software, se definirán las funcionalidades del sistema, analizando detalladamente los requerimientos de software y hardware mediante la captura de requisitos. Además se mostrarán todos los artefactos del flujo de trabajo realizado.

2.2 Objeto de estudio

2.2.1 Descripción de los procesos del negocio

La Reserva Movilizativa, Reserva Estatal y Reserva FAR¹⁰, son vistas como áreas dentro de un mismo almacén y no como un depósito, lo cual genera un inadecuado tratamiento de las operaciones contables. Las tres Reservas de Recursos Materiales tienen como objetivo: Mantener la calidad de los productos, velar porque estén almacenados en la temperatura adecuada, mantener las normas que se deben cumplir y por último y no menos importante, tiene como objetivo, evitar que los productos se venzan. Para lograr que se cumplan estos objetivos, es necesario realizar un grupo de acciones, las cuales son realizadas por los especialistas de las reservas, que no es más que las personas capacitadas para hacer cumplir las normas y objetivos de la reserva. Los especialistas de las reservas materiales disponen para su trabajo: Nomenclaturas, normas y niveles de acumulación aprobados de las reservas, cronograma de evaluación de vencimientos, especificaciones de control y almacenamiento además de los Planes de rotación anticipada (Sólo La Reserva FAR (RESEF)).

La cantidad de materiales con que cuentan las reservas Movilizativa y Estatal, son establecidos por el Instituto Nacional de Reservas Estatales (INRE) y son los que van a ser observados desde cerca por el especialista de la reserva, quien tiene como objetivo mantenerlos en buen estado y no permitir que se venzan. Cada una de las reservas existentes cuenta con una nomenclatura, por otra parte, cada uno de sus productos tiene sus propios atributos.

Los productos de las áreas de mercancías para la venta pueden pasar al área de retenidos si se detecta algún problema tanto de producción como de calidad, hasta que los especialistas lo solucionen esa situación. Si se demuestra que el producto está en mal estado y que puede ocasionar daños irreparables

¹⁰ Estos tipos de reservas están brevemente explicados en el Capítulo 1.

a la población se pasan al área de rechazados y si no, se vuelven al área de mercancías para la venta (Ver Figura 1).



Figura 1 Reservas Materiales. Su ubicación

Entre las Reservas Materiales y las entidades hay especificaciones, acciones o procesos como también se nombran, los cuales mantendrán una buena gestión de los productos. La mayoría de estos procesos y subprocesos no están informatizados y en estos procesos que no están totalmente informatizados, se necesita de un profundo análisis para llevar a cabo alguna operación, se hace difícil el control de los niveles de acumulación y de la rotación de lotes, posibilitando que ocurran vencimientos de los mismos. A esto se une que en el sistema actual no existe una opción donde estén agrupados todos los procesos y subprocesos que se llevan a cabo en el Tratamiento de las Reservas Materiales, además de que las Reservas Movilizativas y Estatales no son vistas como depósitos sino como áreas dentro de un mismo almacén, lo que genera un inadecuado tratamiento de las operaciones contables.

2.2.2 Objeto de automatización

Los procesos que serán objeto de automatización dentro del proceso de Inventario serán los de las Reservas Materiales. Ellos son:

- **Cambio definitivo de surtido por un producto similar en baja cobertura:** Acción que se realiza con productos con próximo vencimiento, pertenecientes a las Reserva Estatal o Reserva Movilizativa, que consiste en hacer una solicitud al Instituto Nacional de Reservas Estatales (INRE) donde se les solicita cambiar cierta cantidad de un producto que tiene alto riesgo de vencimiento, por otro producto o surtido similar que tenga baja cobertura. El producto con riesgo de vencimiento se le compra al INRE para que sea comercializado. El producto sustituto se le vende al INRE para que sea incorporado a la reserva.

- **Cambio temporal de surtido por un producto similar en baja cobertura con reposición del surtido original:** Acción que se realiza con productos con próximo vencimiento, pertenecientes a las Reserva Estatal o Reserva Movilizativa, que consiste en hacer una solicitud al INRE donde se les solicita cambiar cierta cantidad de un producto que tiene alto riesgo de vencimiento, por otro producto o surtido similar que tenga baja cobertura, con el compromiso de devolver ese producto en el plazo establecido por el INRE. El producto con riesgo de vencimiento se le compra al INRE para que sea comercializado. El producto sustituto se le vende al INRE para que sea incorporado a la reserva, hasta que se reponga el original.
- **Préstamo:** Solicitud que se realiza al INRE, para poner a disposición de la comercialización de productos que se encuentran en falta, con el compromiso de devolver cuando aumenta la disponibilidad en las área de Mercancía Para la Venta (MPV), pagando el gravamen¹¹ que se establece en caso de que el producto pertenezca a la Reserva Estatal. Para el caso de la Reserva Movilizativa, no se paga gravamen.
- **Pago de deuda:** Se aplica cuando existe alguna deuda con alguna de las reservas, provocada por un préstamo, un cambio de surtido temporal con reposición del surtido original o la rotación anticipada de los productos de las FAR. El pago se realiza con el mismo producto, uno similar, o por otro diferente, de mutuo acuerdo entre las partes.
- **Cumplimiento del plan:** Son movimientos de entrada a las áreas de reserva, para alcanzar el máximo a acumular establecido para cada producto.
- **Porcentaje de Acumulación:** Implica el porcentaje que se acumulará en las áreas de la Reserva Estatal y Reserva Movilizativa a partir de cada recepción recibida en las áreas de mercancía para la venta. Este valor se parametriza en el sistema.
- **Rotación de lotes:** Movimiento de lotes que se establece entre un área de mercancía para la venta y una de reserva, con el objetivo de evitar vencimientos de productos de las áreas de la reserva.

Estos procesos se encuentran vinculados con los siguientes procesos que posee Inventario:

- Despacho.
- Tratamiento de la Reserva.
- Transferencia entre Áreas.
- Facturación.
- Recepción.

¹¹ **Gravamen:** Impuesto que debe pagar la entidad, por encima de los productos prestados.

Los distintos procesos de las Reservas Materiales, serán el centro para el funcionamiento de la gestión de los materiales de forma óptima en las distintas entidades donde se necesitará de dicha acción. El siguiente esquema muestra la relación que tienen los procesos necesarios para el buen funcionamiento del sistema y los tipos de reservas materiales.



Figura 2 Reservas Materiales. Procesos

Los elementos antes expuestos fomentan la correcta gestión del control de las Reservas Materiales, puesto que se tienen en cuenta las deficiencias que presentan actualmente todos estos procesos.

2.2.3 Información que se maneja

Estos procesos mientras son gestionados generan una serie de documentos, entre ellos se encuentran:

- **Recepción ciega:** Documento que refleja los datos de los productos a recepcionar, como también datos del proveedor. Los mismos se pueden obtener por mediación de facturas, transferencias, etc.
- **Informe de recepción (IR):** Documento que se obtiene a partir de una Recepción Ciega guardada, en este también se refleja todos los datos relacionados con el o los productos recepcionados.
- **Tarjeta de estiba:** Documento que refleja los datos generales de un producto, código CUP, descripción, unidad de medida, etc., así como todos los movimientos que ha tenido el producto dentro del almacén, tales como fecha del movimiento, tipo de movimiento, existencias antes y después de los movimientos realizados, etc.
- **Libro de lotes:** Documento que refleja para cada producto, los lotes que han sido recibidos, la fecha de fabricación, la de vencimiento y sus cantidades.

2.2.4 Propuesta del sistema

Una vez realizado un análisis de los principales problemas existentes en cuanto a las Reservas Materiales y teniendo en cuenta que no existe otro sistema que pueda dar solución a tales necesidades, se propone desarrollar un sistema que aporte soluciones reales y satisfactorias a las necesidades planteadas. Este sistema estará ubicado en cada entidad del país que lo necesite dándole la posibilidad de configurar las Reservas que tendrá y mientras las vayan seleccionando se habilitará su nomenclatura¹². La nomenclatura por su parte dará la opción de seleccionar la cantidad de productos para poder agregarle los datos que necesite realmente, hasta que no se tengan todos los productos escogidos con todos los datos no se marca al tipo de reserva como listo. Una vez que esté marcada la opción de listo se pueden realizar sobre esta reserva todas las acciones y procesos pertinentes para la gestión del control de los recursos. Estas acciones se agregarán como una opción del proceso de Tratamiento de la Reserva. El sistema debe ser fácil de usar y muy seguro. Además, las Reservas Estatales y Reservas FAR serán vistas como almacenes lógicos en lugar de área, en cambio la Reserva Movilizativa si se mantendrá como un área. En cada transacción que se realice deben generarse dos documentos: el de origen y el de destino, en el sistema debe captarse para cada tipo de reserva y para cada producto, el máximo que se debe almacenar, por otra parte deberá generar todos los documentos y anexos de la Reserva Movilizativa. La solución que se propone garantizará la confiabilidad, seguridad y funcionamiento del sistema.

2.3 Modelación del negocio

La modelación del negocio es utilizada para comprender el conjunto de procesos de negocio que tienen lugar dentro de una organización como paso previo para establecer los requisitos del sistema a desarrollar. En ella se describen los procesos de negocio, mediante el artefacto Especificación de procesos de negocio. El mismo tiene como objetivos:

- Lograr una mejor comprensión del problema que el software tiene que resolver.
- Entender la estructura y dinámica de la organización.
- Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- Asegurar que los clientes, usuarios finales y desarrolladores tienen un entendimiento común de la Organización.
- Derivar los requisitos del software necesarios para soportar la organización.

El modelado del negocio se hace más bien para derivar los requisitos a partir de él. Esto garantiza que el software que se desarrolle responda a las necesidades y las condiciones de la organización.

¹² En el epígrafe 2.2.1 se refiere a este tema. Reflejando que cada reserva tiene su nomenclatura así como cada producto tiene sus atributos.

2.3.1 Diagramas de proceso del negocio (BPMN)

Un proceso de negocio es un conjunto de actividades relacionadas que permiten crear un producto o servicio final a través de la transformación de uno o varios productos o servicios iniciales. (Real, 2009)

Un diagrama de proceso de negocio es una representación gráfica del proceso de negocio, en el caso del ERP, utilizando la Notación para la Gestión de Procesos de Negocio (Business Process Management Notation o BPMN).

El diagrama del proceso de negocio perteneciente al tratamiento de las Reservas Materiales, de forma general abarca todos sus procesos. Posibilitando la representación gráfica que especifica la relación entre los artefactos, eventos y roles que intervienen en el negocio. (Ver Anexo 4).

2.3.2 Especificación del proceso Tratamiento de la Reserva

| | |
|-----------------------------------|---|
| Objetivo | - Garantizar la integridad física y cualitativa del Sistema de Reservas Materiales, en evitación de acciones negligentes o infracciones de las disposiciones vigentes para el almacenamiento, la conservación, la rotación, el registro, el control y la protección de las Reservas Materiales, como un papel decisivo en la economía y la defensa del país. |
| Evento(s) que lo genera(n) | - Arribo de mercancía al almacén. - Aprobación de cambio de surtido por riesgo de vencimiento. - Aprobación de cambio de ubicación. |
| Pre condiciones | - Disponibilidad de los Planes de Acumulación de la reserva. - Disponibilidad de nomencladores, normas y niveles de acumulación. - Existencia de al menos una de las áreas de la reserva. - Conocer el inventario en las áreas de MPV. - Disponibilidad de cronograma de evaluación de vencimientos. - Disponibilidad de los planes de rotación de la reserva. |
| Marco legal | - Ley 75/94. Ley de Defensa Nacional. Capítulo XVI. Artículo 123. Reservas Materiales. - Gaceta Oficial No. 015 Extraordinaria de 6 de julio de 2006. Reglamento del Sistema de Reservas Materiales. - Resolución Conjunta MINSAP-MINBAS-INRE 31/07/2009. |
| Clientes internos | - PI. 2- Recepción - PI. 4- Transferencia entre áreas - PI. 6- Salidas - PI. 8- Inventario físico - PI.10- Cierre Contable |
| Clientes externos | - Facturación. |
| Entradas | - Nomencladores, Normas y Niveles de Acumulación. - Planes de Acumulación de la Reserva. - Cronograma de Evaluación de Vencimientos. - Nivel de Inventario de Productos en MPV. - Planes de Rotación de la Reserva. - Cambio de Surtido Definitivo por un Similar en Baja Cobertura. - Cambio de Surtido Temporal por un Similar en Baja Cobertura con Reposición del Surtido Original. |

- Cambio de Ubicación.

Flujo de eventos

Flujo básico Obtención de informes de recepción por movimientos de la reserva estatal

1. **Analizar parámetros de la Reserva.** El especialista de la reservas materiales analiza los nomencladores, normas y niveles de acumulación, los planes de acumulación de la reserva, los planes de rotación de la reserva, el cronograma de evaluación de vencimientos, las existencias en el almacén, los vencimientos probables y a partir de estos análisis, obtiene las especificaciones de la reserva. Estas especificaciones pueden ser para orientar acciones que deben realizarse durante un informe de recepción o pueden ser para acciones dirigidas por el especialista de la reserva. Si se va a actualizar la reserva por una recepción se ejecuta el flujo alternativo **1.a Actualizar la reserva durante un informe de recepción.**
2. Si no se va a actualizar la reserva por un informe de recepción, se verifica si se va a enviar informes a nivel superior. Si se va a enviar informes al nivel superior, se desarrolla el flujo alternativo **2.a Gestionar cambio de surtido o ubicación.**
3. **Enviar requerimientos de la reserva:** El especialista en reservas materiales envía los requerimientos de la reserva al técnico en gestión comercial.
4. **Recibir requerimientos de la reserva.** El técnico en gestión comercial recibe los requerimientos de la reserva.
5. **Si el requerimiento implica generar informes de recepción. Se ejecuta el subproceso Recepción.** En este subproceso se realizan los movimientos de la reserva estatal (préstamos y los cambios de surtido por riesgo de vencimiento temporal y definitivo) hacia áreas de mercancía para la venta que generan informes de recepción. **Si el requerimiento no implica generar informes de recepción se desarrollan los flujos alternos siguientes. 4.a Se ejecuta el subproceso Facturación, 4.b Se ejecuta el subproceso Transferencia entre Áreas.**
6. Se desarrollan los flujos paralelos **5.a Enviar documentos generados al almacén y 5.b Enviar documentos generados al especialista reservas materiales.**
7. **Recibir documentos de movimientos.** El Jefe de Área del Almacén recibe los documentos de los movimientos y los entrega a los manipuladores-despachadores para que los ejecuten.
8. **Ejecutar movimientos.** Los manipuladores despachadores ejecutan los movimientos y actualizan el inventario físico.
9. **Actualizar tarjetas de estiba.** Los manipuladores despachadores actualizan las tarjetas de estiba.
10. **Actualizar libro de lotes.** Los manipuladores despachadores actualizan el libro de lotes.
11. **Concluye el proceso.** Con la actualización del libro de lotes concluye el proceso.

Pos-condiciones

1. Quedan analizados los parámetros de la reserva
2. Quedan enviados los requerimientos de la reserva al Técnico de Gestión Comercial.
3. Quedan generados los informes de recepción y enviados al almacén.
4. Quedan actualizados el inventario físico, las tarjetas de estiba y el libro de lotes.

Salidas

1. Especificaciones para la Reserva para IR (Impreso)

2. Especificaciones para la Reserva Ejecución Dirigida (Impreso)
3. Cambio de Surtido Aprobado Definitivo por un Producto Similar (Impreso)
4. Cambio de Surtido Aprobado Temporal por un Producto Similar con Reposición del Surtido Original (Impreso)
5. Cambio de Ubicación (Impreso)
6. Rotación de Lotes (Impreso)
7. Préstamo (Impreso)
8. Cumplimiento del Plan o Nivel de Completamiento (Impreso)
9. Porcentaje de Acumulación (Impreso)
10. Pago de Deudas (Impreso)
11. Documentos Generados. Informe de Recepción (Impreso)

Flujos paralelos

5.a Enviar documentos al almacén.

1. **Enviar documentos generados al almacén.** El Técnico en Gestión Comercial envía los documentos generados al Jefe de Área del almacén.

Pos-condiciones

1. Quedan enviados al almacén todos los documentos generados.

Salidas

1. N/A

5.b Enviar documentos al especialista de Reservas Materiales.

1. **Enviar documentos generados al especialista de Reservas Materiales.** El Técnico en Gestión Comercial envía los documentos generados al Especialista de Reservas Materiales.
2. **Recibir documentos.** Después de una espera, el especialista de Reservas Materiales recibe una copia de los documentos generados
3. **Si los documentos recibidos se refieren a la Reserva Estatal. Enviar copia de los documentos al INRE:** El especialista de las reservas materiales envía una copia al INRE de los documentos generados. **Si los documentos recibidos se refieren a la Reserva FAR. Se desarrolla el flujo alternativo 5.b.3.a Enviar copia de documentos al MINFAR.**

Pos-condiciones

1. Quedan enviados los documentos generados al MINFAR y al INRE.

Salidas

1. Documentos generados al INRE (Impreso)
2. Documentos generados al MINFAR (Impreso)

Flujos alternos

1.a Actualizar la reserva durante un informe de recepción

1. **Realizar el Informe de Recepción, con especificaciones para la reserva.** indicadas por el especialista de la Reservas Materiales
2. **Si las especificaciones son para la Reserva Estatal. Realizar facturación al INRE.** El Técnico en Gestión Comercial factura al INRE **Si las especificaciones son para la Reserva Movilizativa. Se desarrolla el flujo alternativo 1.a1.a Realizar transferencias entre áreas.**
3. **Si existe la posibilidad de rotar lotes. Realizar la rotación de lotes.** El Técnico en

Gestión Comercial genera una notificación para la rotación de lotes.

4. Volver al paso 1 del flujo alterno 1.a.1.a Realizar transferencia entre áreas.

Pos-condiciones

1. Queda actualizada la reserva por el Informe de recepción con especificaciones.

Salidas

1. Factura Pago Deudas (Impreso, Digital).
2. Factura Cumplimiento del Plan (Impreso, Digital)
3. Factura % Acumulación (Impreso, Digital).

1.a1.a Realizar transferencia entre áreas.

1. **Realizar la transferencia entre áreas.** El Técnico en Gestión Comercial transfiere los productos entre las áreas involucradas.

Pos-condiciones

1. Queda actualizada la reserva por la realización de la transferencia.

Salidas

1. Transferencia entre áreas MPV->RESEM (Impreso, Digital).
2. Transferencia entre áreas.MPV<->RESEM (Impreso, Digital)

2.a Gestionar cambio de surtido o ubicación

1. **Enviar Informes de Posibles Vencimientos para Solicitud de Cambio de surtido.** El Especialista de Reservas Materiales envía a la entidad superior, los informes de los productos con próximo vencimiento que no se pueden rotar, para que se solicite al INRE autorización de cambio de surtido.
2. **Recibir Solicitudes aprobadas de Cambio de Surtido o Ubicación.** El especialista recibe del Organismo superior, las autorizaciones del INRE aprobadas de cambios de surtido o de ubicación.
3. Se ejecuta la actividad 2 del flujo básico

Pos-condiciones

1. Quedan gestionados los cambios de surtido y los cambios de ubicación.

Salidas

1. Informe Posibles Vencimientos para Solicitud de Cambio de surtido por riesgo de vence o ubicación (Impreso, Digital).
2. Informe con Solicitudes aprobadas de Cambio de Surtido o Ubicación (Impreso, Digital)

4.a Se ejecuta el subproceso Facturación

1. **Si el requerimiento implica generar facturas. Se ejecuta el subproceso Facturación.** En este subproceso se realizan los movimientos (pago de deudas, cumplimiento del plan y porcentaje de acumulación) que generan facturas de las áreas de mercancía para la venta hacia la Reserva Estatal y Reserva FAR

Pos-condiciones

1. Quedan generadas las facturas

Salidas

1. Facturas a la Reserva Estatal (Impreso, Digital).
2. Factura a la Reserva FAR (Impreso, Digital)

4.b Se ejecuta el subproceso Transferencia entre Áreas.

1. **Si el requerimiento implica generar Transferencias entra Áreas. Se ejecuta el subproceso Transferencias entre áreas.** En este subproceso se realizan los movimientos

(pago de deudas, cumplimiento del plan y porcentaje de acumulación) que generan facturas de las áreas de mercancía para la venta hacia la Reserva Estatal y Reserva FAR

Pos-condiciones

1. Quedan generadas las facturas

Salidas

1. Facturas a la Reserva Estatal (Impreso, Digital).
2. Factura a la Reserva FAR (Impreso, Digital)

Asuntos pendientes

- Posibles mejoras al proceso.
Dudas sobre el proceso.

Tabla 1 Proceso Tratamiento de la Reserva

2.3.3 Validación de procesos del negocio

Los procesos del negocio concerniente a la Reserva como parte de los procesos del control de Inventario, se encuentra validado por el especialista de la Unión Química Farmacéutica (QUIMEFA): Jorge Reyes Estévez, analista Funcional del Tratamiento de las Reservas Materiales y Especialista Informático de la entidad QUIMEFA, mediante el documento AVAL, el cual se puede encontrar en el Anexo 5.

2.4 Requerimientos y funcionalidades

Condición o capacidad que debe cumplir o poseer un sistema o componente del sistema para satisfacer un contrato, estándar, especificación u otro documento formalmente impuesto. Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen. Es de gran importancia para los desarrolladores y clientes, pues es una descripción completa del comportamiento del sistema a desarrollar.

2.4.1 Aplicación de las técnicas de validación de requisitos

La validación de requerimientos es una actividad muy importante, ya que un levantamiento de requisitos con errores que no se detecten a tiempo, además de no conducir a resultados inesperados provoca costos excesivos y gran pérdida de tiempo.

A continuación se aplican las técnicas definidas para la validación de requisitos.

Revisión Técnica Formal (RTF)

Los analistas del módulo una vez terminada las especificaciones de los requisitos realizaron la reunión de revisión, a la cual convocaron a la especialista funcional a cargo del módulo Inventario el Lic. Pedro J. Núñez Samé, al Dr. José Carlos Del Toro Ríos y a la Ing. Rolando Ramírez Concepción representante del UCID, los cuales firmaron y aprobaron las especificaciones descritas sin posteriores modificaciones.

La validación de requerimientos es una actividad muy importante, pues un levantamiento de requerimientos con errores que no se detecten a tiempo, además de no conducir a resultados inesperados provoca costos excesivos y gran pérdida de tiempo.

Prototipo de interfaz

Con el diseño de los prototipos de interfaz de los requisitos y la posterior aceptación de los usuarios finales, se logró que los desarrolladores y clientes tuvieran un mejor entendimiento acerca de las funcionalidades del futuro sistema.

2.4.2 Listado de Requisitos Funcionales Identificados

Los requerimientos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Los pertenecientes al proceso de la Reserva son:

RF1 Configuración de las Reservas Materiales.

1. Relacionador
2. Eliminar
3. Ayuda

RF2 Gestión de la nomenclatura de la reserva.

1. Producto
2. Imprimir
3. Confirmar

RF3 Porcentaje de acumulación durante un IR.

Aquí solo se le agregarán dos columnas al IR, que serán, el porcentaje de acumulación para la Reserva Estatal y porcentaje de acumulación para la Reserva Movilizativa.

RF4 Rotación de lotes durante un IR.

Cuando se le da clic al botón Contabilizar aparece esta funcionalidad y es la última del IR. Aquí están presentes los tres tipos de Reservas.

RF5 Productos en trámites de préstamo.

- **Listado de acumulación.**

Se marca un porcentaje en dependencia de la condición (Mayor, Menor, Igual), se especifica luego el tipo de reserva y el sistema debe devolver un listado con los productos que cumplan con esta condición. De esta forma el especialista tomará decisiones (Acumular, No acumular o simplemente no hace nada).

- **Niveles de acumulación.**

1. Modificar
2. Imprimir

RF6 Pago de deudas.

1. Buscar
2. Búsqueda avanzada
3. Imprimir
4. Lotes
5. Generar documento de pago

RF7 Solicitud de cambio de surtido.

1. Adicionar
2. Modificar
3. Eliminar
4. Imprimir
5. Confirmar
6. Ratificar
7. Producto
8. Cancelar estado
9. Aprobar
10. Ayuda

2.4.3 Especificación de Requisitos Funcionales

Para la especificación de requisitos se utiliza una plantilla que está definida en el expediente de proyecto del Centro de Informatización de la Gestión de Entidades (CEIGE). (Repositorio, 2011) Esta plantilla cuenta con secciones para describir claramente cada requisito, lo que debe suceder para que se ejecute la funcionalidad en el sistema. Describe además el flujo de eventos que da lugar a todo el proceso de ejecución del requisito.

A continuación se exponen las especificaciones de los requisitos: Configurar las Reservas Materiales y Gestionar la nomenclatura de la reserva, se describen estos requisitos por tener mayor complejidad de implementación y prioridad del negocio perteneciente al Tratamiento de la Reserva. Las demás descripciones se podrán consultar en el Anexo 5.

| | |
|---|---|
| Precondiciones | El usuario ha sido validado. Se debe haber activado el tab de Reservas materiales. |
| Flujo de eventos | |
| Flujo básico Configurar reserva material | |
| 12. | Se selecciona el tipo de reserva material que se va a utilizar. |
| 13. | Si el tipo de reserva seleccionada es de tipo Reserva Estatal y/o Reserva FAR, se selecciona el depósito que va a trabajar con el tipo Reserva Estatal y/o Reserva FAR. |
| 14. | Se selecciona el cliente para las operaciones de la reserva. |
| 15. | Se selecciona el proveedor para las operaciones de reserva. |

| | | |
|--|--|--|
| 16. | El sistema valida (ver validación 1) los datos introducidos. | |
| 17. | El sistema confirma el registro de los datos. | |
| 18. | Concluye el requisito. | |
| Pos-condiciones | | |
| 5. | N/A | |
| Flujos alternativos | | |
| Flujo alternativo 1.a El tipo de reserva seleccionada es Movilizativa | | |
| 1 | Se selecciona el área que se utilizará para la reserva Movilizativa. | |
| 2 | Volver al paso 5 del flujo básico. | |
| Pos-condiciones | | |
| 1 | N/A | |
| Validaciones | | |
| 1 | Se validan los datos según lo establecido en el Modelo conceptual <<Referencia al modelo conceptual en cuestión. | |
| Relaciones | Requisitos Incluidos | Se selecciona el cliente: Adicionar entidad cliente del nomenclador de entidad cliente; en la agrupación Gestionar entidad cliente del nomenclador cliente. Se selecciona el proveedor: Adicionar entidad suministrador del nomenclador de entidad suministrador; en la agrupación Gestionar entidad suministrador del nomenclador suministrador. |
| | Extensiones | N/A |
| Conceptos | Reservas Materiales | Visibles en la interfaz: reserva estatal reserva movilizativa reserva FAR Utilizados internamente: reserva estatal reserva movilizativa reserva FAR |
| | Cliente | Visibles en la interfaz: nombre de entidad cliente dirección entidad cliente código de entidad cliente tipo de cliente Utilizados internamente: cuenta bancaria de entidad cliente cuenta de la sucursal de entidad cliente NIT entidad cliente nombre cliente cargo cliente CI cliente |
| | Proveedor | Visibles en la interfaz: nombre de entidad suministrador dirección entidad suministrador código de entidad suministrador tipo de suministrador Utilizados internamente: cuenta bancaria de entidad suministrador cuenta de la sucursal de entidad suministrador NIT entidad suministrador |

| | |
|------------------------------|--|
| | nombre suministrador cargo suministrador CI suministrador |
| Depósito | Visibles en la interfaz: código de depósito descripción de depósito Utilizados internamente: N/A |
| Requisitos especiales | Son los requisitos no funcionales específicos para el requisito. |
| Asuntos pendientes | Posibles mejoras al requisito. |

Tabla 2 Especificación del requisito Configurar las Reservas Materiales

| | | |
|----------------------------|---|---|
| Precondiciones | Debe estar habilitada la configuración en la entidad para usar la reserva. Debe estar configurado al menos un tipo de reserva con la que va a trabajar la entidad. | |
| Flujo de eventos | | |
| Flujo básico | Listar nomenclatura de las reservas materiales | |
| 1 | El sistema muestra un listado de la nomenclatura (las nomenclaturas) de la reserva. Se muestran los siguientes datos: código descripción cantidad de producto listo | |
| 2 | Concluye el requisito. | |
| Pos-condiciones | | |
| 1 | N/A | |
| Flujos alternativos | | |
| Flujo alternativo | | |
| 1 | N/A | |
| Pos-condiciones | | |
| 1 | N/A | |
| Validaciones | | |
| 1 | N/A | |
| Relaciones | Requisitos Incluidos | N/A. |
| | Extensiones | N/A |
| Conceptos | Nomenclaturas | Visibles en la interfaz: código descripción cantidad de productos listo Utilizados internamente: código |

| | |
|------------------------------|--|
| | descripción cantidad de productos listo |
| Productos | Visibles en la interfaz: código descripción um precio de costo clave surtido % de acumulación plan norma margen vence rotación Utilizados internamente: código descripción um precio de costo clave surtido % de acumulación plan norma margen vence rotación |
| Requisitos especiales | Son los requisitos no funcionales específicos para el requisito. |
| Asuntos pendientes | Posibles mejoras al requisito. |

Tabla 3 Especificación del requisito Gestionar la nomenclatura de la reserva

2.4.4 Validación de los Requisitos Funcionales

La Validación de requisitos: Son Proceso que tiene como misión demostrar que los requisitos definen realmente el sistema que el usuario necesita.

Los requisitos del proceso de negocio concerniente a la Reserva como parte de los procesos de Inventario, se encuentra validado por el especialista de QUIMEFA, esto se puede encontrar en el Anexo 3 Entrevista 3. Los documentos del Centro de Soluciones de Gestión referentes al proceso de Inventario constituyen una constancia de la validación, identificación y especificación de los requisitos funcionales anteriormente descritos. (Repo, 2011)

2.4.5 Requisitos No Funcionales

Los requisitos no funcionales forman una parte significativa de la especificación. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto. Si se conocen las propiedades no funcionales, seguro, conveniente y agradable puede ser el software, un producto será bien

aceptado. Estos requerimientos tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, entre otros. (Díez González, 2006)

Los requisitos no funcionales con los que cuenta la aplicación son los mismos establecidos por el equipo de Arquitectura para el producto Cedrux. Esto se debe al éxito que ha tenido la etapa Experimental del Modelo de desarrollo definido por esta Unidad, puesto que el ERP-Cuba, está pasando por esta etapa en distintas entidades como por ejemplo: Rafael Trejo, Cubataxi entre otros. Estos requisitos se encuentran contemplados en la planilla Especificaciones complementarias perteneciente al Cedrux. (Repo1, 2011) A continuación se muestran los requisitos no funcionales con que debe cumplir la aplicación web perteneciente al control de las Reservas Materiales.

Confiabilidad (CON): En un ambiente de servidores, uno funcionando como centro de datos (servidor destino) y el resto como servidores locales (servidor fuente). Se realizaron réplicas de 50000 tuplas desde un servidor fuente hacia el servidor destino.

- Réplica online provocando fallos en la conexión.
- Réplica online con funcionamiento normal.
- Réplica offline mediante un fichero.
- Al configurar una nueva institución no se llega a la funcionalidad donde se va a definir un elemento que dependa de otro componente sin haber configurado el componente con antelación

Rendimiento (REN)/ Disponibilidad.

- Un usuario accede a cualquier interfaz de la aplicación y ésta debe estar disponible en un período de 0.1 a 0.2 segundos.
- Si 100 000 usuarios concurrentes acceden a la interfaz principal de la aplicación, ésta debe estar disponible para todos a la vez en un período de 0.5 a 1.0 segundos.
- Se solicita un recurso al sistema, que puede ser ligero (capacidad por debajo de 2 MB), medio (capacidad de 2 a 10 MB) o complejo (más de 10 MB de capacidad) en un servidor de 1 GB de memoria RAM y se recibe en un período de 0.1 a 0.7, de 0.8 a 2.0 y de 3.0 a 5.0 segundos respectivamente.
- Se intercambian datos con el sistema, ya sea mediante acciones de inserción, búsqueda, eliminación o modificación de datos, en un servidor de 1 GB de memoria RAM y se recibe la notificación de la acción realizada en un período de 0.1 a 0.7 segundos.

Seguridad (SEG)

- Se desea eliminar o modificar un elemento de la base de datos que está siendo utilizado por otro elemento que depende de él; el sistema no permite que este elemento sea eliminado.
- Dos usuarios acceden a la misma interfaz de modificar e intentan modificar la misma información simultáneamente, el sistema solo debe dejar que modifique uno de ellos y notifica al otro usuario que actualice para obtener la última versión de los datos.
- Se produce una excepción en los datos que no permite realizar la operación indicada por el usuario, el sistema mostrará un mensaje notificando el error pero solo con la información necesaria para el usuario, no con información del sistema.
- Al configurar una nueva institución no se llega a la funcionalidad donde se va a definir un elemento que dependa de otro componente sin haber configurado el componente con antelación.

Software (SFT)

El sistema requiere como componentes de software del lado cliente: Navegador Mozilla Firefox versión 2.2.

| Del lado del servidor de Aplicaciones requiere: | Del lado del cliente de BD: |
|---|---|
| <ul style="list-style-type: none"> • Un servidor web apache versión 2.0. • Sistema Operativo Ubuntu Server. | <ul style="list-style-type: none"> • Ubuntu Server. • PostgreSQL versión 8.3. |

Hardware (HDW)

Los componentes de hardware que se necesitan para el correcto funcionamiento de la aplicación son:

- Por el lado cliente:
 - Procesador: 1.40 GHZ
 - RAM: 256 MB (recomendado 512 Mb)
 - Tarjeta de Red: 1
- Del lado del servidor de aplicaciones:
 - Tarjeta de Red: 1
 - Procesador: 3.00 GHZ
 - RAM: 1GB
 - Disco duro: 160 GB
 - UPS: 1
 - Lector de CD: 1
- Del lado del servidor de Base de datos:
 - Tarjeta de Red: 1
 - Procesador: 3.00 GHZ
 - RAM: 1GB

Disco duro: 160 GB

UPS: 1

Lector de CD: 1 Prototipo de Interfaz

2.5 Conclusiones parciales

En este capítulo, se describió el negocio referente a la Reserva, mostrando las deficiencias que presentan actualmente las entidades cubanas. El empleo de las técnicas de captura de requisitos, permitió identificar, describir los requisitos funcionales y procesos del negocio, teniendo en cuenta lo descrito en el negocio, se identificaron los requisitos no funcionales del sistema.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

En este capítulo se hará referencia a la disciplina de Análisis y Diseño del sistema. Los artefactos generados, modelarán los principales procesos del sistema propuesto a desarrollar, en un lenguaje entendible para los programadores. Del análisis de esta etapa, depende en gran medida el logro de los objetivos finales, mientras que la calidad depende de la modelación clara y específica de los artefactos generados mediante el análisis. De igual manera se definirá en el presente capítulo la arquitectura y los patrones a utilizar.

3.2 Análisis

En el análisis, los requisitos son descritos a un nivel de detalle mínimo, dándole las funcionalidades a un grupo de objetos. Los requisitos son cada vez más refinados, teniendo en cuenta los objetivos que se quieren tener al final del desarrollo del producto.

3.2.1 Modelo de clases del análisis

Las clases del análisis, representan una abstracción de una o varias clases y/o subsistemas del diseño del sistema. Mientras que el modelo de análisis se usa para refinar los requisitos en la captura de requisitos y no es más que un paquete de clases del análisis. (Slideshare, 2011)

Con el modelo de análisis se pretende lograr tres objetivos fundamentales:

- Describir lo que requiere el cliente.
- Establecer una base para la creación de un diseño de software.
- Definir un conjunto de requisitos que se pueda validar una vez que se construye el software.

A continuación se muestra el Modelo de clases del análisis del proceso Configuración de las Reservas Materiales. Los demás se pueden consultar en el Anexo 6 Diagramas de clases del análisis.

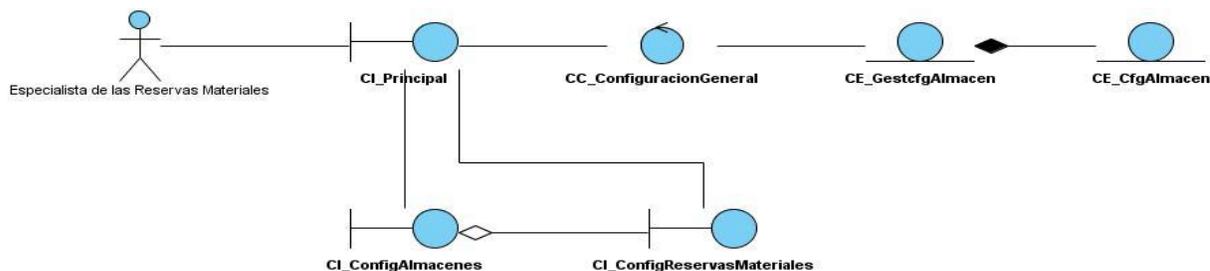


Figura 4 Diagrama de clases del análisis. Configuración

3.3 Arquitectura

La Arquitectura del Software es el diseño de más alto nivel de la estructura de un sistema, programa o aplicación y tiene la responsabilidad de:

- Definir los módulos principales
- Definir las responsabilidades que tendrá cada uno de estos módulos
- Definir la interacción que existirá entre dichos módulos:
- Control y flujo de datos
 - Secuenciación de la información
 - Protocolos de interacción y comunicación
 - Ubicación en el hardware

Aporta una visión abstracta de alto nivel, posponiendo el detalle de cada uno de los módulos definidos a pasos posteriores del diseño.

La Arquitectura del Software es la IEEE¹³ Std 1471-2000 que plantea: “La Arquitectura del Software es la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”. (Desarrolloweb.com, 2009)

Consiste además en el diseño de componentes de una aplicación (entidades del negocio), generalmente utilizando patrones de arquitectura. El diseño arquitectónico debe permitir visualizar la interacción entre las entidades del negocio y además poder ser validado, por ejemplo por medio de diagramas de secuencia. Un diseño arquitectónico describe en general el cómo se construirá una aplicación de software.

En otras palabras, la arquitectura va a ser la vista conceptual para que analistas, diseñadores y desarrolladores trabajen sobre una línea en común, para alcanzar los objetivos del sistema según las necesidades que existan. Es necesario hacer una buena definición de los estilos arquitectónicos, patrones y mecanismos de diseño.

3.3.1. Estilos arquitectónicos

Los estilos arquitectónicos: Sirven para sintetizar estructuras de soluciones, definen los patrones posibles de las aplicaciones, permiten evaluar arquitecturas alternativas con ventajas y desventajas conocidas ante diferentes conjuntos de requerimientos no funcionales. Este trabajo tendrá sus bases en una arquitectura multicapas, basada en componentes que junto a un MVC conforman un estilo híbrido. A continuación se explican cada uno de estos estilos arquitectónicos.

Arquitectura multicapas

Esta arquitectura brinda la ventaja de aislar definitivamente la lógica de negocios de todo lo que tenga que ver con el origen de datos, ya que desde el manejo de la conexión, hasta la ejecución de una consulta, la

¹³ IEEE: Corresponde a las siglas de *The Institute of Electrical and Electronics Engineers*, el Instituto de Ingenieros Eléctricos y Electrónicos, una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.

manejará la capa de Acceso a Datos. De este modo, ante cualquier eventual cambio, solo se deberá cambiar una capa en específico. Esta arquitectura está descrita en el Capítulo 1

1. Capa de presentación o capa de interfaz de usuario: es la forma de ofrecer al usuario un modo de interactuar con la aplicación. Las interfaces de usuario se implementan utilizando formularios, controles u otro tipo de tecnología que permita procesar y dar formato a los datos de los usuarios, así como adquirir y validar los datos entrantes procedentes de éstos.

2. Capa de lógica de negocio: es la parte más importante de una aplicación debido a que encapsula las responsabilidades y los procesos que en ella se comprenden, su funcionamiento basado en el uso de clases o componentes y a su vez compuestos estos por métodos que de una forma u otra establecen la relación entre los componentes de presentación y los componentes de acceso a datos. Para el acceso al subsistema de la capa de acceso a datos se presenta una interfaz denominada “Factoría típica”, clase que implementa el patrón para clases con un objetivo y funcionalidades similares, o sea todo flujo de información entre estas dos capas es a través de esta clase.

3. Capa de Acceso a Datos: permite conocer el formato de los datos que se intercambian a través de los componentes de acceso a datos y la capa lógica de negocio, permiten el acceso a los datos a través de los mismos. Además usan la tecnología de acceso a datos PDO e implementa un interfaz de programación para la gestión de los datos.

4. Datos: Contiene el almacenamiento de los datos, es decir todas las tablas de la Base de Datos.

5. Servicios: En esta última capa se encuentran todos los subsistemas que prestan y consumen servicios entre sí.



Figura 10 Arquitectura de tres capas (Galán Ramírez, y otros, 2008)

Arquitectura basada en componentes

En el presente acápite se puede observar como se realiza un componente a partir de la reutilización de los ya existentes. A continuación se muestra el diagrama de componentes perteneciente al componente de Reservas Materiales.

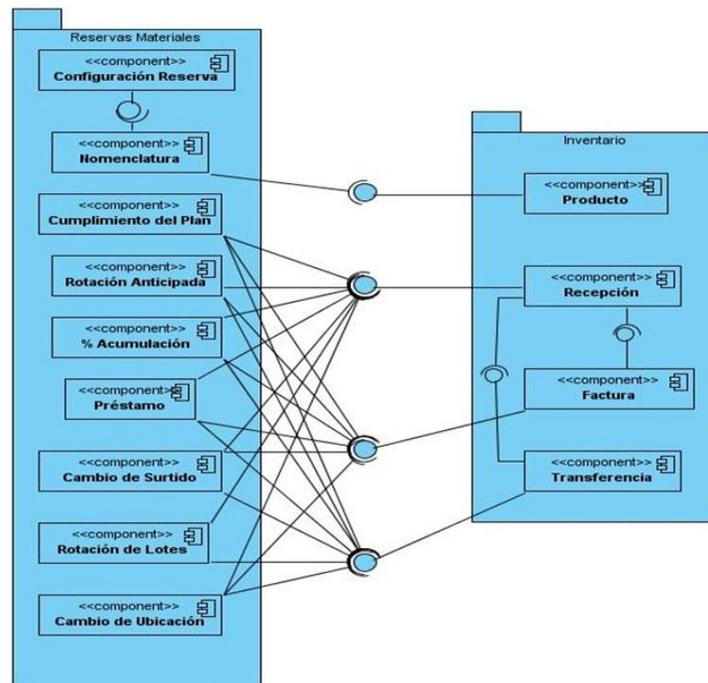


Figura 11 Diagrama de componentes. Reservas Materiales

Modelo-Vista-Controlador

El estilo arquitectónico MVC es utilizado para el desarrollo de aplicaciones Web con el fin de separar en tres componentes distintos la interfaz de usuario, la lógica de negocio y los datos persistentes, potenciando la flexibilidad y la adaptabilidad a futuros cambios. Por su parte:

El Modelo: Es la representación de la información que maneja la aplicación. Son los datos puros que puestos en un contexto del sistema son mostrados al usuario por medio del Controlador, proveen de información al usuario o a la aplicación misma.

La Vista: Constituye la representación del modelo en forma gráfica, disponible para la interacción con el usuario. En una aplicación web la "Vista " es la página HTML con contenido dinámico sobre el cual el usuario puede realizar operaciones.

El Controlador: Se encarga de responder a las solicitudes del usuario desde la Interfaz, manejando los diferentes eventos a través de las funcionalidades necesarias y la información perteneciente al Modelo.

Para el desarrollo del Sistema Integral de Gestión de Entidades CEDRUX se decidió trabajar con este patrón, evidenciándose en los *framework* definidos para cada una de las capas como parte del MVC de cada componente dentro de la aplicación, es decir: para la Vista: ExtJS-Framework, el cual es muy utilizado en el desarrollo de aplicaciones Web con tecnología AJAX, para el Controlador: Zend-Framework quien emplea específicamente el estilo Modelo- Vista- Controlador como base de su funcionamiento y

para agilizar el acceso a datos en el Modelo se utilizó Doctrine, un potente y completo sistema ORM (Mapeo Objeto Relacional). (Fernández González, y otros, 2010)

De forma general, la arquitectura utilizada es formada por un híbrido, ya que los estilos arquitectónicos descritos anteriormente no pueden verse de forma independiente, esto trae numerosas ventajas, dentro de ellas:

- Desarrollos paralelos: en cada capa.
- Aplicaciones más robustas debido al encapsulamiento.
- Mantenimiento y soporte más sencillo: es más sencillo cambiar un componente que modificar íntegramente una aplicación.
- 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.

3.4 Aplicación de los patrones de diseño

Los patrones utilizados fueron los patrones GOF y GRASP como se definieron en el Capítulo 1 del presente trabajo. Los patrones **GOF** están divididos fundamentalmente en tres grupos: estructurales, creacionales y de comportamiento. De ellos se utilizaron para la modelación del proceso de las Reservas:

Fachada: Es un patrón estructural que proporciona una interfaz unificada para un conjunto de interfaces de un subsistema. Define una interfaz de alto nivel que hace que el subsistema sea más fácil de usar. En el Cedrux los servicios están basados en el patrón fachada, permitiendo el acceso a métodos que se encuentran en otros componentes tanto dentro como fuera del subsistema de Inventario. Esto se debe a la relación que existe entre las clases controladoras y estos servicios. Entre las ventajas que brinda el patrón fachada se encuentran: que el usuario que lo va a consumir no necesita tener conocimientos del negocio interno del componente y facilita la portabilidad del componente.

Mediador: Es un patrón de comportamiento utilizado en el tratamiento de excepciones. En este trabajo su uso se basa en el diseño para validar la ocurrencia de errores al insertar elementos en la base de datos, este error es captado por las capas superiores, reenviando la excepción hasta la capa de aplicación donde dicho error se traduce al lenguaje del usuario.

Los patrones **GRASP** tienen como objetivo la descripción de los principios fundamentales del diseño de objetos para la asignación general de responsabilidades. Existen nueve patrones GRASP, de los cuales fueron utilizados:

Experto: Este se evidenció con el uso de clases que poseen responsabilidades específicas a cumplir de acuerdo con la información que manejan, el subsistema de inventario donde se incorpora la solución, cuenta con clases controladoras, modelos y de entidad que poseen funciones concretas de acuerdo con la

información que gestionan. Además, se modeló una clase entidad por cada tabla de la base de datos, posibilitando el trabajo específico y directo con el experto en la información.

Creador: El patrón de creación permitió que el diseño pudiera soportar bajo acoplamiento, mayor claridad, encapsulamiento y reutilización. Para lograrlo, las clases controladoras son las responsables de crear el objeto de las modelos, y estas a su vez son responsables de crearlo en las entidades.

Bajo Acoplamiento: Es un principio que asigna la responsabilidad de controlar el flujo de eventos del sistema a clases específicas. Esto facilita la centralización de actividades. El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión. Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema.

Alta Cohesión: En la perspectiva del diseño orientado a objetos, la cohesión es una medida de que determina la cota de las responsabilidades de una clase en cuanto a las relaciones y enfoques de la misma. Las clases con responsabilidades estrechamente relacionadas y que no realizan un trabajo enorme se caracterizan por tener una alta cohesión.

Controlador: Se utilizan cuando la aplicación es muy extensa, como es el caso del presente trabajo, ya que la solución está integrada al sistema Cedrux, brindando la posibilidad de tener varios controladores (Controllers) especializados en las funcionalidades de cada componente impidiendo la sobresaturación.

3.5 Modelo de clases del diseño

El diseño es el proceso de aplicar distintas técnicas y principios con el propósito de definir un producto con los suficientes detalles como para permitir su realización física. Con el diseño se pretende construir un sistema que respete los requisitos funcionales y no funcionales identificados y traducirlos de manera que represente al software. Permite producir un modelo o representación técnica del software que se va a desarrollar.

El diagrama de clases del diseño describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación.

Normalmente contiene la siguiente información:

- Clases, asociaciones y atributos.
- Interfaces con sus operaciones y constantes.
- Métodos.
- Información sobre los tipos de atributos.
- Navegabilidad.
- Dependencias.

(Visconti, y otros, 2009)

3.4.1. Mecanismos de diseño

El diseño es muy importante en la modelación de un sistema, para que haya una buena comprensión y comunicación entre el diseñador y el programador de forma completa, debido a que el tránsito de diseño a programación debe ser de forma efectiva para la obtención de un buen producto. Para lograr todo lo antes mencionado se propone utilizar mecanismos de diseño, que no es más que un artefacto de RUP, que agrupa un conjunto de clases del diseño, colaboraciones e incluso subsistemas del modelo del diseño que lleva a cabo requisitos comunes como persistencia, distribución, seguridad, y funcionamiento. (Galán Ramírez, y otros, 2008)

En el diseño de Cedrux, que es donde va a estar ubicado el componente de Reservas Materiales, propone utilizar dos mecanismos de diseño, uno para manejar la seguridad del sistema y el otro para la persistencia de los datos.

Mecanismos de seguridad

El sistema que se está implementando, utiliza el sistema de seguridad de Cedrux, como antes se había mencionado. Este sistema fue implementado bajo el uso de servicio Web, el cual está encargado del control de los accesos, autenticación y registro de los eventos que van ocurriendo simultáneamente. Se propone un mecanismo de seguridad que sirva a todas las aplicaciones que utilizan este servicio, para garantizar de esta forma la correcta funcionalidad de los requerimientos de este sistema.

El servicio Web, tiene muchos beneficios, entre ellos se destaca la posibilidad que tiene hacer intercambio de datos entre ellas, sin importar el lenguaje de programación en el que fueron desarrolladas o las plataformas sobre las que son ejecutadas. Por estas razones se propone el siguiente mecanismo de diseño para seguridad, basado en servicio Web.

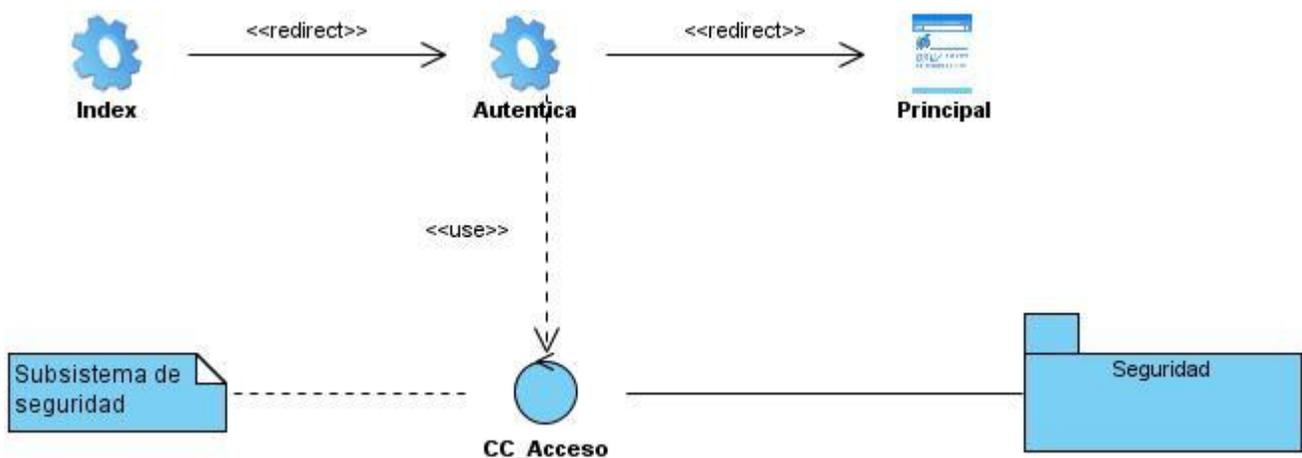


Figura 12 Mecanismo del diseño de seguridad.

Mecanismo de acceso a datos

El sistema que se está implementando, utiliza el mecanismo de acceso a datos de Cedrux. Este mecanismo se va a encargar de que el acceso a datos sea fácil y óptimo, siendo estos sus objetivos fundamentales. De esta forma se obtendrán diagramas más entendibles que crearán una referencia para los programadores y que estos se guíen por una única línea base. Se obtendrán a su vez la eficiencia y reutilización de los datos. La vista de este mecanismo (Ver Figura 13), muestra un conjunto de clases que se relacionan entre sí para dar paso a la manipulación de los datos desde su nivel más bajo, es decir, utilizando los objetos nativos brindados por el entorno de desarrollo PHP como son la extensión PDO que define una ligera interfaz, para el acceso a bases de datos, siguiendo así hasta la abstracción del acceso a datos a través de la clase controladora de la cual heredan las clases particulares del sistema. Se definió una clase que encapsulara las instancias de estos objetos, que está concebida para instanciar todas las tablas del sistema para utilizarlas de algún modo. A continuación se muestra una descripción más detallada de estas clases.

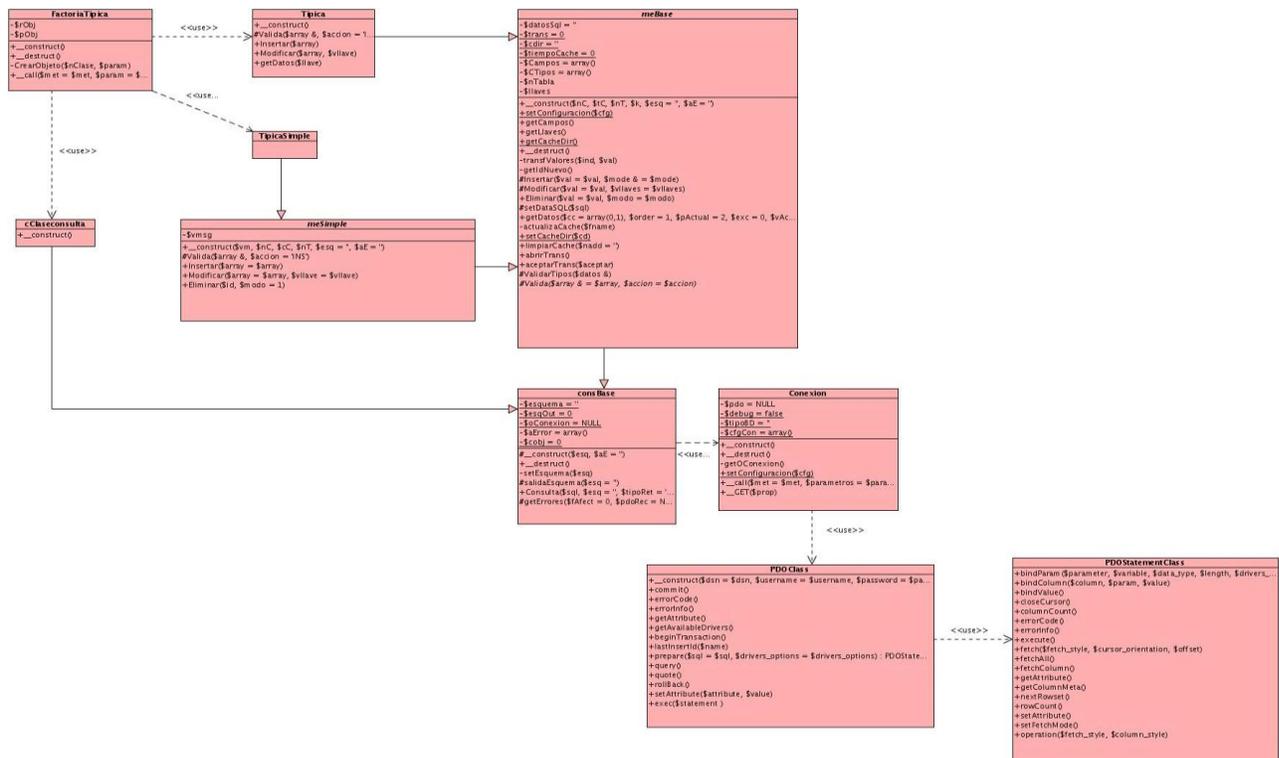


Figura 13 Mecanismo de acceso a datos.

Descripción de las clases

| Nombre de la clase | Descripción |
|--------------------|--|
| FactoriaTipica. | Implementa la interfaz del modelo de persistencia con el resto de los subsistemas. A través de esta clase se crean y se manipulan los objetos de las |

| | |
|----------------|--|
| | típicas simples, los nomencladores y las demás típicas. Es una puerta entre la capa de Acceso a Datos y la capa de Lógica de Negocio, para su creación se tomó en cuenta lo dictado por el patrón de diseño Factoría, el cual centraliza en una clase controladora la creación de objetos de un tipo determinado. Implementa un método de instanciación de clases típicas. |
| Típica | Representa a las clases típicas en general de la aplicación. Existe una típica para cada entidad de la base de datos. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway, que consiste en crear una instancia por cada tabla existente en la BD. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, (INSERT, DELETE, UPDATE). Hereda de la clase abstracta meBase. |
| TípicaSimple | Representa a las clases típicas para nomencladores simples. Para la implementación de esta clase se decidió aplicar el patrón de diseño Table Data Gateway. Sus métodos consisten en las operaciones básicas que se realizan sobre estas tablas, (INSERT, DELETE, UPDATE). Hereda de la clase abstracta meSimple. |
| cClaseconsulta | Representa a las clases consultas en general de la aplicación. Existe una clase consulta para cada entidad de la base de datos. Hereda de la clase abstracta consBase. |
| meSimple | Es la base para la implementación de las típicas que responderán a los nomencladores simples del modelo de persistencia dado. Redefine las operaciones básicas con la funcionalidad de Validación dada, Entidades cuya estructura responde al siguiente patrón: idALGO, ALGO, |

| | |
|----------|---|
| | <p>actual. Donde ALGO representa la descripción del atributo principal de los nomencladores clásicos, meSimple define las operaciones básicas que pudieran realizarse a una entidad (INSERT, DELETE, UPDATE) para los nomencladores simples. Hereda de la clase abstracta meBase.</p> |
| meBase | <p>Es la base para el resto de las que implementen funcionalidades para el trabajo con las entidades del sistema a implementar. Implementa las operaciones básicas que pudieran realizarse a una entidad (INSERT, DELETE, UPDATE). Hereda de consBase la operación de CONSULTA.</p> |
| consBase | <p>Es la base en toda la jerarquía de Acceso a Datos y es empleada para aportar contenido dinámico a las plantillas. Encapsula el objeto conexión. Implementa la operación de CONSULTA.</p> |
| Conexión | <p>Se encarga de establecer la conexión con el servidor de la Base de Dato a través de un objeto PDO de la librería de PHP. Se concibió aplicando el patrón Singleton el cual garantiza una única instancia para una clase y la creación de un mecanismo global (único) de acceso a dicha instancia.</p> |
| PDO | <p>Es un modelo de acceso a bases de datos para PHP. PDO brinda una capa de abstracción para el acceso a bases de datos desde PHP.</p> <p>El modelo de acceso a datos tiene una estrecha relación con los estándares de base de datos, concretamente los estándares ayudan a identificar grupos o tipos de entidades según el prefijo, por ejemplo: las entidades de datos tienen prefijo "dat_" y los nomencladores o clasificadores "nom_". Cuando alguien va a elaborar el modelo de acceso a datos, debe realizar un estudio de todas las</p> |

| | |
|--|---|
| | entidades con las que se va a relacionar. Esto le permitirá ir identificando patrones que posteriormente serían presuntas clases abstractas, ejemplo de esto es meSimple. |
|--|---|

El modelo de acceso a datos tiene una estrecha relación con los estándares de base de datos, concretamente los estándares ayudan a identificar grupos o tipos de entidades según el prefijo, por ejemplo: las entidades de datos tienen prefijo “dat_” y los nomencladores o clasificadores “nom_”. Cuando se elabora el modelo de acceso a datos, se debe realizar un estudio de todas las entidades con las que se relacionan. Esto le permitirá ir identificando patrones que posteriormente serían presuntas clases abstractas, ejemplo de esto es meSimple.

En una base de datos X se afirma que del 100% de las entidades el 70% son nomencladores o clasificadores, el 25% son entidades de datos o principales, un 2% son auxiliares, un 2% puede ser de seguridad y el restante 1% otros tipos. Dentro de los dos grupos más grandes, específicamente en el de los nomencladores, existe un patrón básico al que se le ha llamado “entidades simples”, son fáciles de identificar ya que estas entidades siempre tienen la estructura siguiente: idALGO, ALGO, actual. Por tanto las operaciones básicas tendrán un comportamiento exacto en el 99% de los casos y de ahí es que surge meSimple.

Los nomencladores simples heredan de meSimple y de meBase, en el diagrama no se refleja la relación de herencia entre ellos, porque como meSimple hereda de meBase, se infiere que el nomenclador simple también hereda de meBase.

3.6 Diagramas de secuencias

El diagrama de secuencia, es aquel que se usa para modelar la interacción entre objetos de un sistema a través del tiempo y se realiza uno para cada caso de uso o cada proceso. Este diagrama incluye tanto la implementación del escenario como los objetos y clases que se utilizan para implementar en el escenario. El escenario que se muestra a continuación es el de Configuración de la Reserva perteneciente a Configuración.

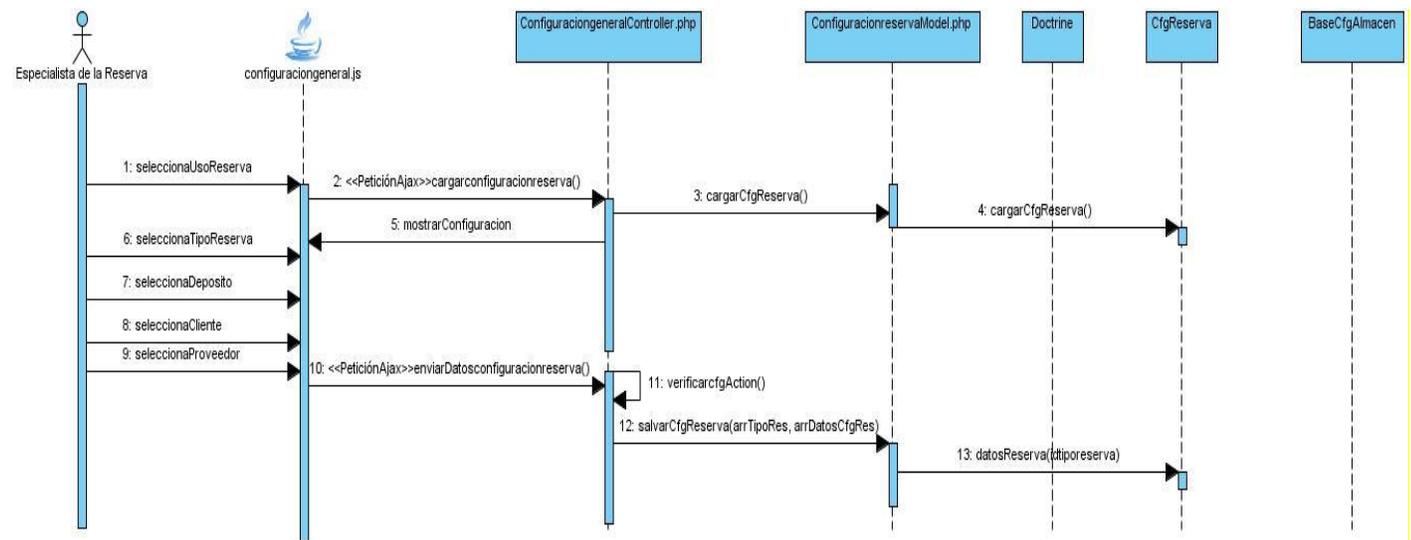


Figura 14 Diagrama de secuencia. Configuración de la Reserva

3.6 Diagrama de clases del diseño Web

El diagrama de clases del diseño Web es el que muestra la relación entre las clases de una aplicación. Se ha realizado un diagrama para cada uno de los procesos de Reservas Materiales.

Este sistema tiene concebido por cada proceso lo siguiente:

- Una clase de lógica de negocio (In_ **Proceso**) por cada proceso.
- Un fichero **GLOBAL** para la configuración de las direcciones de las clases a través de la función `__autoload` y algún elemento de configuración.
- La página servidora tendrá las instancias de los objetos In_ **Proceso**.
- Una página servidora **principal** que redireccionará a servidora de cada Caso de Uso por módulo.
- Una interfaz **cControlacceso** que permite la autenticación de los usuarios.
- Una clase **InBase**: Es la clase base de lógica de negocio, de la cual heredarán todas las clases "In_Proceso". En ella se define el objeto de la clase Factoría Típica.
- Una clase **FactoríaTípica** que a través de la cual se crean y se manipulan los objetos de acceso a datos como las tablas y consultas. Es una interfaz entre la Lógica de Negocio y la capa de Acceso a Datos. Basada en el patrón Factoría.
- Los ficheros .js que son los ficheros ext controls (menú, toolbar, formularios).
- Una hoja de estilo en cascada (CSS), para definir la presentación del documento.
- Se utiliza un marco para cargar las interfaces en la misma área de trabajo con el objetivo de usar el mismo banner y menú en todas las interfaces.

A continuación se muestran el diagrama del diseño Web para Configuración, los demás diagramas se encontrarán en el Anexo 7.

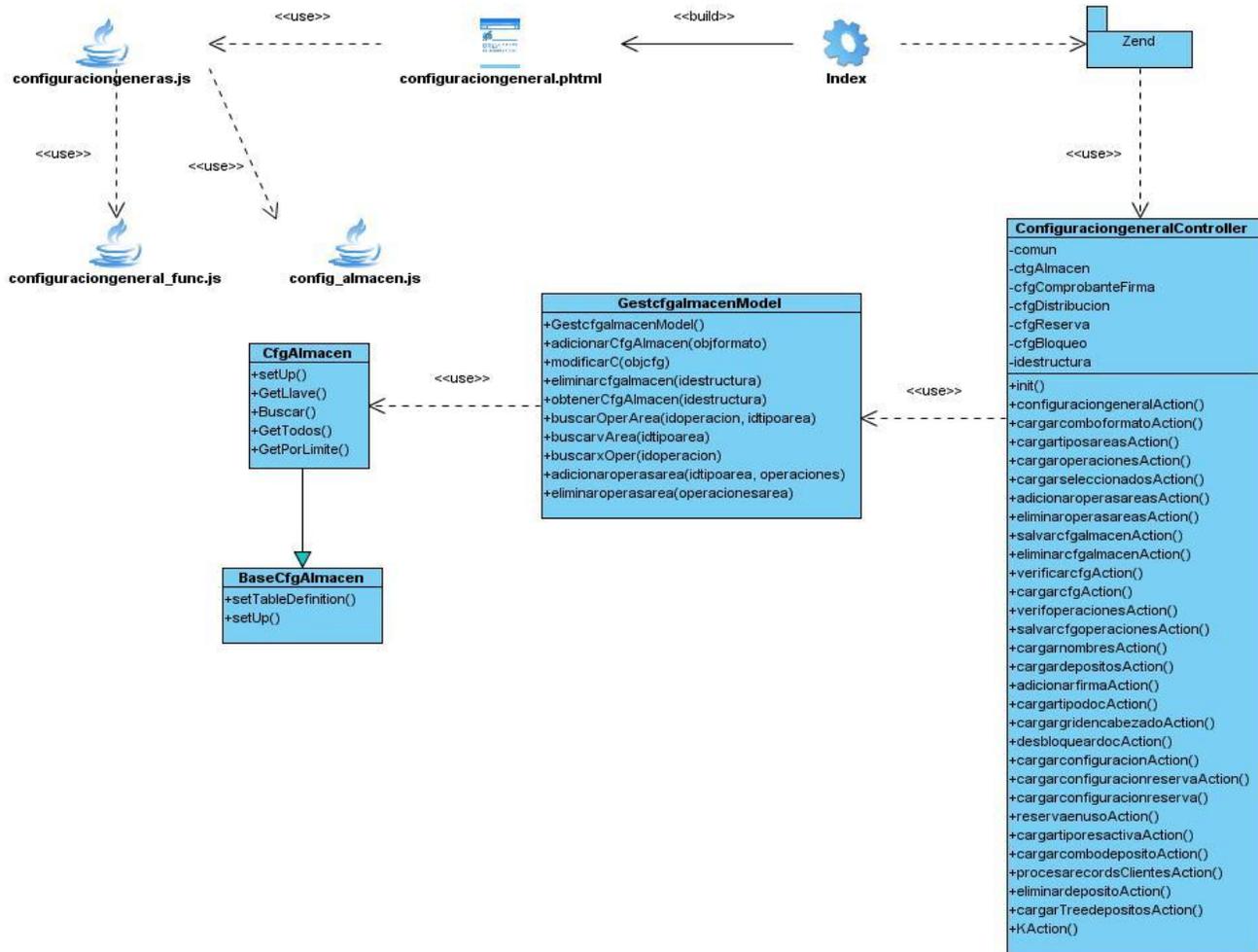


Figura 15 Diagrama de clases del diseño. Configuración

3.7 Diseño de la Base de Datos

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Puesto que un diseño correcto es esencial para lograr los objetivos fijados para la base de datos, parece lógico emplear el tiempo que sea necesario en aprender los principios de un buen diseño ya que, en ese caso, es mucho más probable que la base de datos termine adaptándose a sus necesidades y pueda modificarse fácilmente. Se propone el siguiente diseño de datos.

3.7.1. Modelo lógico de datos

Partiendo del diseño conceptual, se llega a un diseño lógico. Transformando las entidades y relaciones obtenidas del modelo anterior en tablas. Para ello se usa la normalización. El modelo lógico permite una

definición clara y específica de los esquemas conceptuales y en los esquemas conceptuales se definen las estructuras lógicas de almacenamiento y las relaciones que se darán entre ellas. (Ver Anexo 17 Modelo lógico de datos)

3.7.2. Modelo físico de datos

Este modelo describe los datos en el nivel más bajo y permite identificar algunos detalles de implantación para el manejo del hardware de almacenamiento.

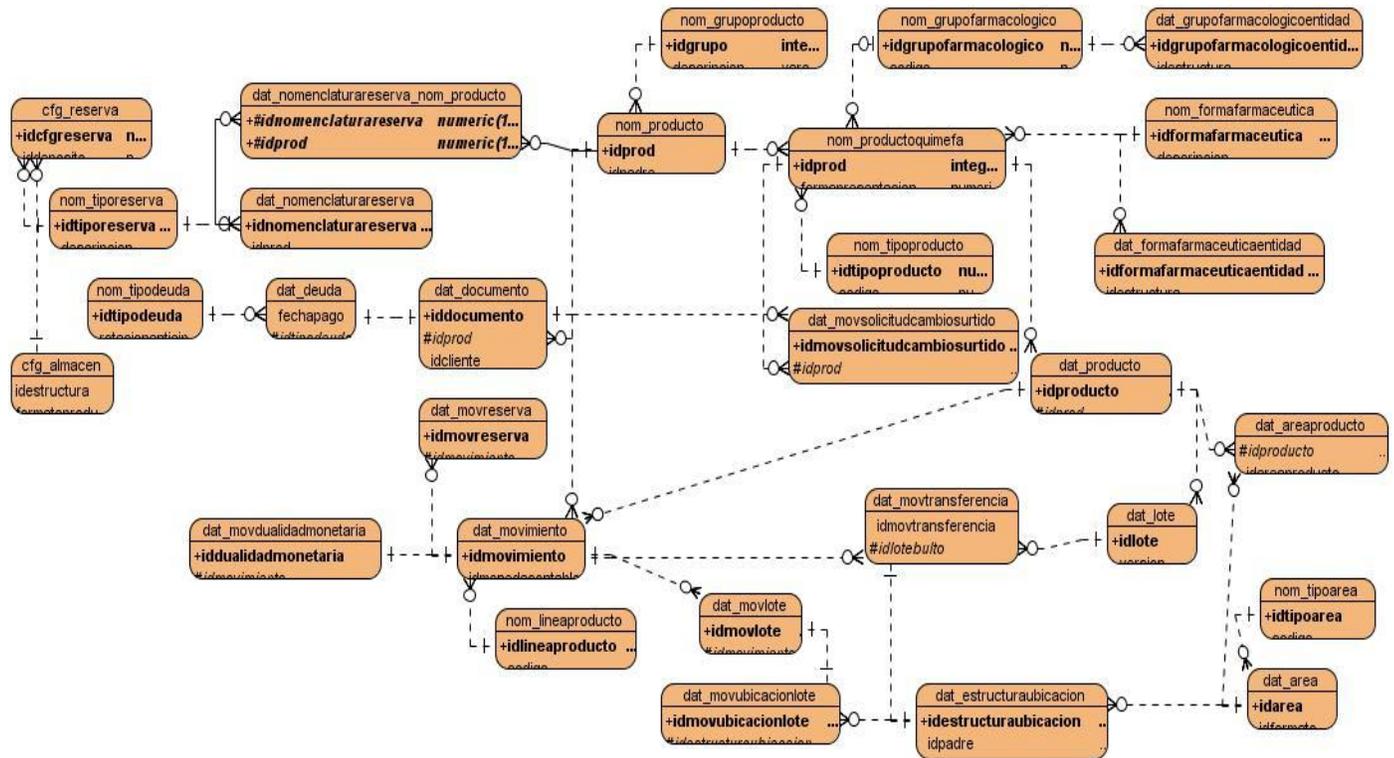


Figura 22 Modelo físico de datos

3.7.3. Descripción de las tablas

Este modelo describe los datos en el nivel más bajo y permite identificar algunos detalles de implantación para el manejo del hardware de almacenamiento.

A continuación se muestran las descripciones de las tablas `cfg_almacen` y `cfg_reserva`, las cuales se encuentran en el modelo físico de datos presente en el subepígrafe anterior. Las demás descripciones se podrán consultar en el Anexo 8 Descripción de las tablas del modelo físico de datos.

| Nombre: <code>cfg_almacen</code> | | |
|---|------|--|
| Descripción: Tabla que guarda la previa configuración de la entidad. | | |
| Atributo | Tipo | Descripción |
| <code>idestructura</code> | int | Identificador que proviene de la tabla de las entidades (<code>nom_filaestruc</code>) del esquema de |

| | | |
|--------------------|--------|---|
| | | estructura y composición que permite poner en práctica el concepto de la multientidad. Es el atributo llave que dice según id a que entidad pertenece. |
| formatoproducto | String | Formato de configuración que provee el formato que tiene el producto por entidades. Formato que proviene del esquema de datos maestros. |
| realordenado | String | Comprueba si se puede variar la cantidad de producto de los documentos de salida verificando si se genera o no un comprobante de operaciones que por defecto esta en 1 porque siempre se va a generar un comprobante de salida. |
| comprobante | String | Comprobante. |
| recargodescuento | String | Campo para saber si el almacén tiene recargo descuento comercial. |
| categoriaproducto | String | Atributo para identificar si se va a trabajar con la categoría del producto. |
| idejercicio | int | Identificador del ejercicio. |
| preciodiferenciado | Double | Campo para identificar si la entidad vende a precios diferenciado los productos (0) no vende (1) si vende. |
| certificadocalidad | String | 1 si la entidad va a crear certificado de calidad. |
| guiaexpedicion | String | 1 si la entidad va crear guía expedición. |
| reservasmateriales | String | si la entidad va a trabajar con reservas materiales y 0 no. |

Tabla 9 Descripción de la tabla “cfg_almacen” de la base de datos

| | | |
|---|-------------|---|
| Nombre: cfg_reserva | | |
| Descripción: Tabla que guarda la previa configuración de la reserva. | | |
| Atributo | Tipo | Descripción |
| idcfgreserva | int | Identificador que se genera automáticamente con la secuencia sec_cfgreserva_seq. |
| idtiporeserva | int | Identificador del tipo reserva el cual hace referencia a mod_invenatrio.nom_tiporeserva |

| | | |
|-------------|-----|------------------------------|
| iddeposito | int | Identificador del depósito. |
| idcliente | int | Identificador del cliente. |
| idproveedor | int | Identificador del proveedor. |

Tabla 10 Descripción de la tabla “cfg_reserva” de la base de datos

3.8 Conclusiones parciales

Al concluir el presente capítulo se definió las especificaciones de cada uno de los requisitos identificados, además de la arquitectura y la modelación de la solución propuesta mediante el uso de técnicas para la captura de requisitos. Se generaron los artefactos necesarios para la realización de la implementación del futuro software.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

4.1 Introducción

En el presente capítulo se muestra el diagrama de despliegue y el diagrama de componentes general respectivamente, dándole cumplimiento a la etapa de Construcción que propone el modelo de desarrollo utilizado. La elaboración de los diagramas anteriormente mencionados, implica la implementación del componente de Reservas Materiales, así como la realización de las pruebas necesarias, garantizando de esta forma el correcto funcionamiento del sistema y su calidad.

4.2 Implementación

El flujo de trabajo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros. En este flujo de trabajo se implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. Además se deben hacer los tests de unidad: cada desarrollador es responsable de testear las unidades que produzca. El resultado final de este flujo de trabajo es un sistema ejecutable. (EcuRed, 2011)

4.2.1 Diagrama de despliegue

Los Diagramas de Despliegue o de Distribución muestran la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre los mismos. Los nodos se utilizan para modelar la topología del hardware sobre el que se ejecuta el sistema. Representa típicamente un procesador o un dispositivo sobre el que se pueden desplegar los componentes y se representa como un cubo. (Software, 2009)

La siguiente figura muestra el diagrama de despliegue propuesto por el equipo de arquitectura.

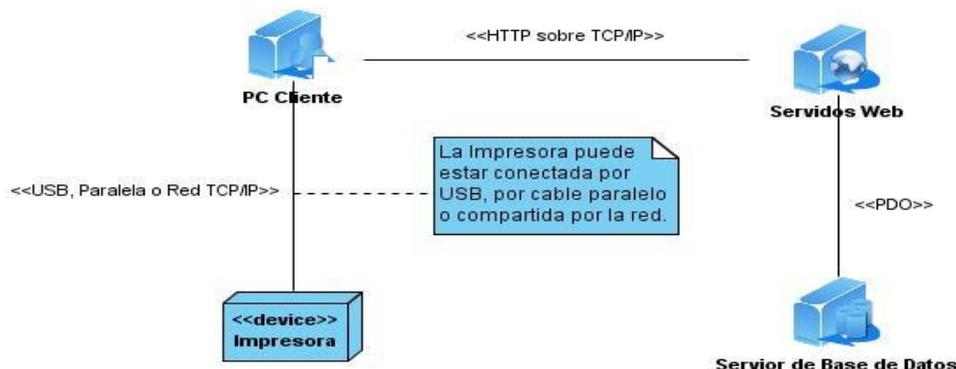


Figura 21 Diagrama de despliegue. Reservas Materiales

4.2.2 Diagrama de componentes

Un diagrama de componentes Muestra como el sistema está dividido en componentes y las dependencias entre ellos. Proveen una vista arquitectónica de alto nivel del sistema. Ayuda a los desarrolladores a visualizar el camino de la implementación. Permite tomar decisiones respecto a las tareas de implementación y los skills ¹⁴requeridos. (Eduardo Rivera Alva, 2011)

El diagrama de Componente General.del proceso de Reservas Materiales se encuentra en el Anexo 17.

4.3 Prueba

El único instrumento adecuado para determinar el status de la calidad de un producto software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos. En las pruebas se usan casos de prueba, especificados de forma estructurada mediante Técnicas de prueba entre las que se encuentran: las pruebas de caja blanca (pruebas unitarias) y de caja negra. (software, 2005)

4.3.1 Pruebas de unidad

Las pruebas unitarias son una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. Luego, con las pruebas de integración, se podrá asegurar el correcto funcionamiento del sistema o subsistema en cuestión.

El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas. Proporcionan un contrato escrito que el trozo de código debe satisfacer. Las pruebas unitarias son capaces de:

- Asegurar la calidad del código entregado. Es la mejor forma de detectar errores tempranamente en el desarrollo. No obstante, esto no asegura detectar todos los errores, por tanto prueba de integración y aceptación siguen siendo necesarias.
- Ayudar a definir los requerimientos y responsabilidades de cada método en cada clase probada.
- Constituyen una buena forma de ejecutar pruebas de concepto. Cuando es necesario hacer pruebas de conceptos sin integrar usar pruebas unitarias se convierte en un método efectivo.

¹⁴ Skills (Habilidades): Es la capacidad adquirida para llevar a cabo resultados predeterminados con la mínima inversión de tiempo, energía, o ambos. Las habilidades a menudo se pueden dividir en habilidades de dominio general y específicos. Por ejemplo, en el ámbito de trabajo, algunas habilidades generales incluyen la gestión del tiempo el trabajo en equipo y liderazgo, la auto-motivación y otros, mientras que las habilidades de dominio específico, sería útil sólo para un determinado trabajo. Por lo general requiere de ciertos estímulos ambientales y situaciones para evaluar el nivel de habilidad que se muestra y se utiliza.

- Permiten hacer refactoring¹⁵ tempranamente en el código. No es necesario todo un ciclo de integración para hacer refactoring en la aplicación, basta con ver como se comporta un caso de prueba para hacer refactoring unitario sobre la clase que estamos probando en cuestión.
- Permiten incluso hacer pruebas de Stress tempranamente en el código. Por ejemplo un método que realice una consulta SQL que exceda los tiempos de aceptación es posible optimizarla antes de integrar con la aplicación.
- Permiten encontrar errores o bugs tempranamente en el desarrollo. Y está demostrado que mientras más temprano se corrijan los errores, menos costará corregirlos. (Rodríguez, 2006)

4.3.1.1. Pruebas de caja blanca

Al desarrollar un nuevo software o sistema de información, la primera etapa de pruebas a considerar es la etapa de pruebas unitarias o también llamada pruebas de caja blanca (White Box), estas pruebas también son llamadas pruebas modulares ya que permiten determinar si un módulo del programa esta listo y correctamente terminado, estas pruebas no se deben confundir con las pruebas informales que realiza el programador mientras esta desarrollando el módulo. (B., 2009)

Estas pruebas se realizan sobre las funciones internas de un módulo en concreto, están dirigidas a las funciones internas. Entre las técnicas usadas se encuentran; la cobertura de caminos, pruebas sobre las expresiones lógico-aritméticas, pruebas de camino de datos, comprobación de bucles.

En la siguiente figura se representa las pruebas de Caja Blanca.

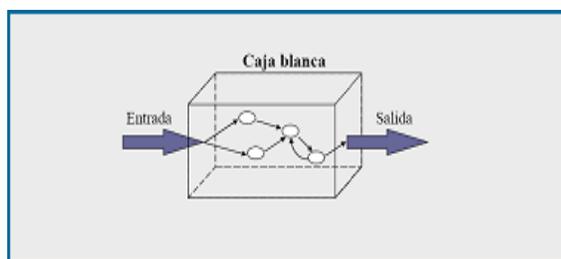


Figura 23 Representación de pruebas de caja blanca

Existen varios tipos de prueba de caja blanca como se muestra a continuación:

Prueba de Condición: Es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa.

Prueba de Flujo de Datos: Se seleccionan caminos de prueba de un programa de acuerdo con la ubicación de las definiciones y los usos de las variables del programa.

¹⁵ Refactoring o **refactorización** es una técnica de la ingeniería de software para reestructurar un código fuente, alterando su estructura interna sin cambiar su comportamiento externo.

Prueba de Bucles: Es una técnica de prueba de caja blanca que se centra exclusivamente en la validez de las construcciones de bucles.

Prueba del Camino Básico: Esta técnica permite obtener una medida de la complejidad lógica de un diseño y usar esta medida como guía para la definición de un conjunto básico. La idea es derivar casos de prueba a partir de un conjunto dado de caminos independientes por los cuales puede circular el flujo de control. Para obtener dicho conjunto de caminos independientes se construye el grafo de flujo asociado y se calcula su complejidad ciclomática¹⁶.

La técnica del camino básico es la que se va a utilizar para realizar las pruebas pertinentes al software en cuestión, esta técnica permite obtener una medida de la complejidad lógica del código de cada método, programa o módulo dado. La idea es derivar casos de prueba a partir de un conjunto dado de caminos independientes que existen en la codificación por los cuales puede circular el flujo de control. Es además una de las más eficientes en cuanto a cobertura de código, pues logra que se ejecuten todos los bucles en sus límites operacionales. (Fuentes, 2009)

Los pasos que se sigue para aplicar esta técnica son:

- A partir del diseño o del código fuente, se dibuja el grafo de flujo asociado.
- Se calcula la complejidad ciclomática del grafo.
- Se determina un conjunto básico de caminos independientes.
- Se preparan los casos de prueba que obliguen a la ejecución de cada camino del conjunto básico.
- Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.

Para aplicar la técnica del camino básico se debe introducir la notación para la representación del flujo de control, este puede representarse por un grafo de flujo en el cual:

- Cada nodo del grafo corresponde a una o más sentencias de código fuente.
- Todo segmento de código de cualquier programa se puede traducir a un grafo de flujo.
- Se calcula la complejidad ciclomática del grafo.
- El grafo de flujo se utiliza tres elementos fundamentales:
- Nodos: representan cero, una o varias sentencias en secuencia. Cada nodo comprende como máximo una sentencia de decisión (bifurcación).
- Aristas: líneas que unen dos nodos.
- Regiones: áreas delimitadas por aristas y nodos. Cuando se contabilizan las regiones de un programa debe incluirse el área externa como una región más.

¹⁶ Complejidad ciclomática: Este indicador fue desarrollado en 1976 por Thomas J. McCabe y refleja directamente el número de caminos independientes que un programa puede tomar durante su ejecución. (Gomez, 2008)

- Nodos predicado: cuando en una condición aparecen uno o más operadores lógicos (AND, OR, XOR...) se crea un nodo distinto por cada una de las condiciones simples. Cada nodo generado de esta forma se denomina nodo predicado.

Para la construcción del mismo se debe tener en cuenta la notación de las siguientes instrucciones.

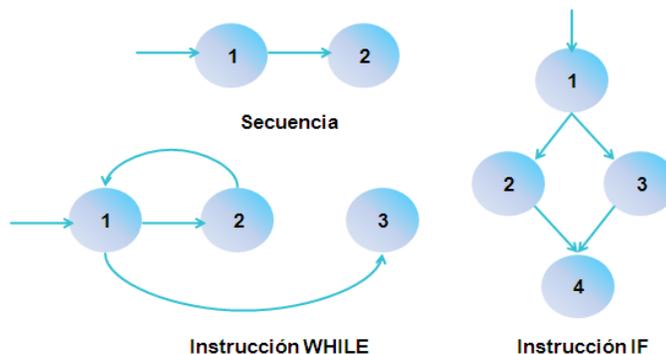


Figura 24 Notación de grafos de flujos para las instrucciones: Secuenciales, If y While

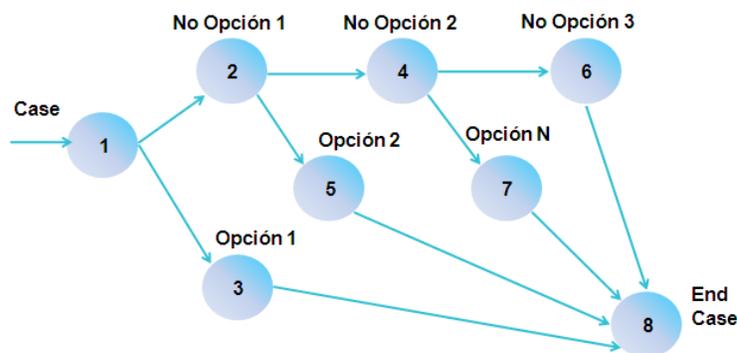


Figura 25 Notación de grafos de flujos para la instrucción: Case

Para la realización de la prueba de caja blanca, se necesita calcular en primer lugar la complejidad ciclomática del algoritmo o fragmento de código a analizar. En el contexto del método de prueba del camino básico, el valor de esta complejidad define el número de caminos independientes de dicho programa, y por lo tanto, el número de casos de prueba a realizar.

Se muestra a continuación las sentencias de código enumeradas del procedimiento realizado sobre el método modificarProducto (\$arrProducto), el cual pertenece al proceso de Gestionar nomenclatura de la reserva. Dicho método se encarga de modificar los valores de los productos existentes en la nomenclatura de la Reserva que se seleccione para realizarle algún cambio pertinente. (dinamica, 2009)

```

public function modificarProducto($arrProducto) {
    if ($arrProducto) { (1)
        $objprod = Doctrine::getTable('DatNomenclaturareserva')->find($arrProducto->idnomenclaturareserva); (2)
        foreach ($arrProducto as $atrib => $value) { (3)
            if ($atrib != idnomenclaturareserva) (4)
                $objprod->$atrib = $value; (5)
        } (6)
        $objprod->save(); (7)
    }
    return true; (8)
}

```

Figura 26. Representación del algoritmo modificarProducto (\$arrProducto)

A continuación se muestra en la Figura 27 el grafo asociado al algoritmo anterior.

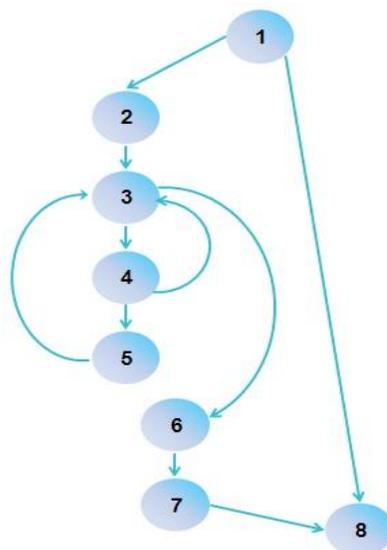


Figura 27 Grafo de flujo asociado al algoritmo modificarProducto (\$arrProducto)

Luego de la construcción del grafo ya se tienen la cantidad total tanto de nodos (posibles resultados) como de aristas (son los caminos que puede o no seguir según los resultados) y nodos predicado (son los nodos de los cuales salen 2 o más caminos) que conforman los datos necesarios para realizar los cálculos de la complejidad ciclomática.

Existen varias formas de calcular la complejidad ciclomática de un programa a partir de un grafo de flujo:

- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como

$$V(G) = \text{Aristas} - \text{Nodos} + 2$$

- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como

$$V(G) = \text{Nodos Predicado} + 1$$

- El número de regiones del grafo coincide con la complejidad ciclomática,

$$V(G) = R$$

Aplicando estas fórmulas al grafo de la Figura 27 se tiene que:

$$V(G) = \text{Aristas} - \text{Nodos} + 2$$

$$V(G) = 10 - 8 + 2$$

$$V(G) = 4$$

$$V(G) = \text{Nodos Predicado} + 1$$

$$V(G) = 3 + 1$$

$$V(G) = 4$$

$$V(G) = R$$

$$V(G) = 4$$

Mediante los cálculos realizados anteriormente basados en las formas en que se puede obtener el valor de la complejidad ciclomática, se puede observar que todos arrojaron como resultado el valor de 4, lo que significa que existen cuatro posibles caminos por donde puede circular el flujo. Este valor representa el límite mínimo del número total de casos de pruebas para el procedimiento tratado.

A continuación se muestran los caminos básicos por los que puede recorrer el flujo. En estas representaciones se subrayan los elementos de cada camino que los hacen independientes a los demás.

| Número | Camino básico |
|--------|---------------------------|
| 1 | 1, 8 |
| 2 | 1, 2, 3, 6, 7, 8 |
| 3 | 1, 2, 3, 4, 3, 6, 7, 8 |
| 4 | 1, 2, 3, 4, 5, 3, 6, 7, 8 |

Tabla 38 Caminos básicos del flujo

Una vez extraídos los caminos del flujo se procede a la realización de los diseños de casos de prueba para este procedimiento. Es necesario realizar al menos un caso de prueba por cada camino básico.

Para realizar estos casos de prueba es necesario cumplir con las siguientes exigencias:

- Descripción: Se hace la entrada de datos necesaria, validando que ningún parámetro obligatorio pase nulo al procedimiento o no se entre algún dato erróneo.
- Condición de ejecución: Se especifica cada parámetro para que cumpla una condición deseada para ver el funcionamiento del procedimiento.
- Entrada: Se muestran los parámetros que entran al procedimiento.
- Resultados Esperados: Se expone resultado que se espera que devuelva el procedimiento.

Caso de prueba para el camino básico 1

Camino 1: [1- 8]

Descripción:

- Los datos de entrada cumplirán con los siguientes requisitos:

El parámetro arrProducto será un arreglo con los atributos de un producto.

Condición de ejecución:

- Debe existir al menos un producto insertado en la base de datos para que pueda ser modificado.

Entrada:

- \$arrProductos = array ().
- Clave surtido = nulo.
- % Acumulación = nulo.
- Plan = nulo.
- Norma = nulo.
- Margen Vence Rotación = nulo.

Resultados esperados:

Se espera que el producto sea modificado satisfactoriamente.

El arreglo de productos pasado por parámetros llega vacío, significando que no existe ningún producto insertado en la base de datos. No se lograron los resultados esperados.

Caso de prueba para el camino básico 2

Camino 2: [1 - 2 - 3 - 6 - 7- 8]

Descripción:

- Los datos de entrada cumplirán con los siguientes requisitos:

El parámetro arrProducto será un arreglo con los atributos de un producto.

Condición de ejecución:

- Debe existir al menos un producto insertado en la base de datos para que pueda ser modificado.

Entrada:

- \$arrProductos = array (1 producto).
- Clave surtido = 2222.
- % Acumulación = 2.
- Plan = 500.
- Norma = 60.
- Margen Vence Rotación = 8.
- Nunca se cumplió la condición (\$atrib != idnomenclaturareserva) de la línea 4 del código.

Resultados esperados:

Se espera que el producto sea modificado satisfactoriamente.

El producto se modifico satisfactoriamente.

Caso de prueba para el camino básico 3

Camino 2: [1 - 2 - 3 - 4 - 3 - 6 - 7 - 8]

Descripción:

- Los datos de entrada cumplirán con los siguientes requisitos:

El parámetro arrProducto será un arreglo con los atributos de un producto.

Condición de ejecución:

- Debe existir al menos un producto insertado en la base de datos para que pueda ser modificado.

Entrada:

- \$arrProductos = array (1 producto).
- Clave surtido = nulo.
- % Acumulación = 2.
- Plan = nulo.
- Norma = 60.
- Margen Vence Rotación = 8.

Resultados esperados:

Se espera que el producto sea modificado satisfactoriamente.

Se espera que se muestre un mensaje con el siguiente texto: “Faltan campos obligatorios o existen valores incorrectos”. El sistema subraya el campo en rojo mostrando el mensaje: “Este campo es obligatorio.”.

Caso de prueba para el camino básico 4

Camino 2: [1 - 2 - 3 - 4 - 5 - 3 - 6 - 7- 8]

Descripción:

- Los datos de entrada cumplirán con los siguientes requisitos:

El parámetro arrProducto será un arreglo con los atributos de un producto.

Condición de ejecución:

- Debe existir al menos un producto insertado en la base de datos para que pueda ser modificado.

Entrada:

- \$arrProductos = array (1 producto).
- Clave surtido = 2222.
- % Acumulación = 2.
- Plan = 500.
- Norma = 60.

- Margen Vence Rotación = 8.

Resultados esperados:

Se espera que el producto se modifique satisfactoriamente.

El producto fue modificado satisfactoriamente.

Luego de aplicar los distintos casos de pruebas, se pudo comprobar que el flujo de trabajo de la función está correcto ya que cumple con las condiciones necesarias que se habían planteado.

4.4 Validación del modelo de diseño utilizando métricas

Es importante tener en cuenta en la evaluación de la calidad del diseño la creación de métricas básicas inspiradas en el estudio de la calidad del diseño orientado a objeto referenciadas por Pressman, teniendo en cuenta que este estudio brinda un esquema sencillo de implementar y que a la vez cubre los principales atributos de calidad de software.

Los atributos de calidad a tener en cuenta son los siguientes:

Responsabilidad: Consiste en la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta.

Complejidad del diseño: Consiste en la complejidad que posee una estructura de diseño de clases.

Complejidad de implementación: Consiste en el grado de dificultad que tiene implementar un diseño de clases determinado.

Reutilización: Consiste en el grado de reutilización de presente en una clase o estructura de clase, dentro de un diseño de software.

Acoplamiento: Consiste en el grado de dependencia o interconexión de una clase o estructura de clase, con otras, esta muy ligada a la característica de Reutilización.

Complejidad del mantenimiento: Consiste en el grado de esfuerzo necesario a realizar para desarrollar un arreglo, una mejora o una rectificación de algún error de un diseño de software. Puede influir indirecta, pero fuertemente en los costes y la planificación del proyecto.

Cantidad de pruebas: Consiste en el número o el grado de esfuerzo para realizar las pruebas de calidad (Unidad) del producto (Componente, modulo, clase, conjunto de clases, etc.) diseñado.

Nivel de Cohesión: Consiste en el grado de especialización de las clases concebidas para modelar un dominio o concepto específico.

Abstracción del diseño: Consiste en la capacidad de modelar lo más cercano posible a la realidad un concepto o dominio determinado.

Para el desarrollo del sistema integral Cedrux aun no está definida la etapa de diseño de una aplicación por lo que la dirección del equipo de trabajo del proyecto ERP-Cuba definió que para la validación del

diseño se iban a utilizar las métricas: Tamaño Operacional de Clase (TOC), Relaciones entre Clases (RC), Profundidad de Herencia (PH) y Número de Operaciones Redefinidas para una clase hija (NOR).

4.4.1. Tamaño operacional de clase (TOC)

El Tamaño operacional de clase está dado por el número de métodos asignados a una clase. La cual afecta un grupo de atributos como se muestra a continuación en la tabla 39.

| Atributo que afecta | Modo que lo afecta |
|-------------------------------|--|
| Responsabilidad | Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase. |
| Complejidad de implementación | Un aumento del TOC implica un aumento de la complejidad de implementación de la clase. |
| Reutilización | Un aumento del TOC implica una disminución en el grado de reutilización de la clase. |

Tabla 39 Tamaño operacional de clase (TOC)

4.4.2. Relaciones entre clases (RC)

Las relaciones entre clases están dadas por el número de relaciones de uso de una clase con otras. La cual afecta un grupo de atributos como se muestra en la siguiente tabla.

| Atributo que afecta | Modo que lo afecta |
|--------------------------------------|--|
| Acoplamiento | Un aumento del RC implica un aumento del acoplamiento de la clase. |
| Complejidad del mantenimiento | Un aumento del RC implica un aumento de la complejidad del mantenimiento de la clase. |
| Reutilización | Un aumento del RC implica una disminución en el grado de reutilización de la clase. |
| Cantidad de pruebas | Un aumento del RC implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase. |

Tabla 40 Relaciones entre clase (RC)

4.4.3. Profundidad de herencia (PH)

La profundidad de herencia está dada por la profundidad en la herencia de las clases heredadas de un nodo padre. La cual afecta un grupo de atributos como se muestra a continuación en la siguiente tabla.

| Atributo que afecta | Modo que lo afecta |
|--------------------------------------|---|
| Complejidad del mantenimiento | Un aumento del PH implica un aumento de la complejidad del mantenimiento de la clase. |
| Reutilización | Un aumento del PH implica un aumento en el grado de reutilización de la clase. |
| Complejidad del diseño | Un aumento del PH implica un aumento de la complejidad del diseño de la clase. |

Tabla 41 Profundidad de herencia (PH)

4.4.4. Número de Operaciones Redefinidas para una clase hija (NOR)

El número de Operaciones Redefinidas para una clase hija está dado por la cantidad de operaciones redefinidas en cada clase hija. La cual afecta un grupo de atributos como se muestra a continuación.

| Atributo que afecta | Modo que lo afecta |
|--------------------------------------|---|
| Abstracción del diseño | Un aumento del NOR implica que se ha logrado una buena definición de la abstracción del diseño de clases. |
| Cantidad de pruebas | Un aumento del NOR implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase. |
| Complejidad del mantenimiento | Un aumento del NOR implica un aumento de la complejidad del mantenimiento de la clase. |

Tabla 42 Número de Operaciones Redefinidas para una clase hija (NOR)

4.4.5. Resultados de los instrumentos de evaluación de las métricas

4.4.5.1. Resultados del instrumento de evaluación de la métrica Tamaño operacional de la Clase (TOC)

En las figuras 28 y 29 se muestran los resultados obtenidos por el instrumento de medición agrupados en intervalos definidos. Por otra parte en las figuras 31, 32 y 33 presentes en el Anexo 10 Representación de

los atributos de la métrica TOC, se pueden observar todos los demás resultados obtenidos en el instrumento de medición TOC para los atributos de calidad, responsabilidad, complejidad y reutilización estarán. Para obtener más información, se pueden ver: el instrumento y la tabla de resultado ubicada en el Anexo 9 Instrumento de medición de la métrica Tamaño operacional de clase (TOC).

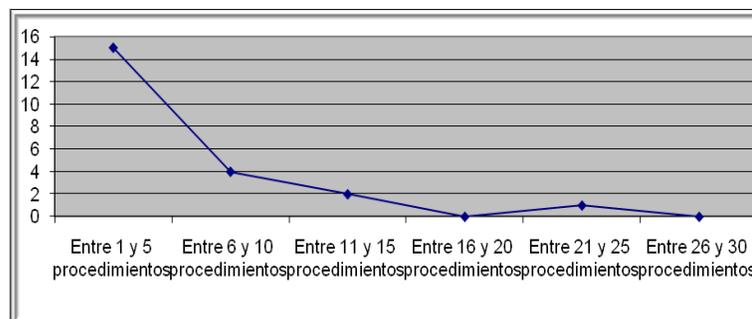


Figura 28 Representación de los resultados obtenidos en el instrumento agrupados en los intervalos definidos

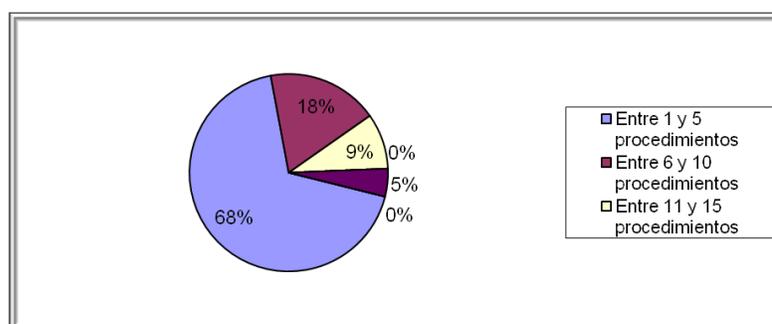


Figura 29 Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica TOC, se obtiene como resultado que la implementación de las clases necesarias para dar solución a la problemática, tiene una calidad aceptable, teniendo en cuenta que el 68 % de las clases utilizadas, incluidas en la solución posee menos cantidad de operaciones que la mitad del valor máximo registrado en las mediciones. Además el 100% de las clases poseen evaluaciones positivas en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).

4.4.5.2. Resultados del instrumento de evaluación de la métrica Relaciones entre clases (RC)

En las figuras 34 y 35 se muestran los resultados obtenidos por el instrumento de medición agrupados por la tendencia de los valores. Por otra parte en las figuras 37, 38, 39 y 40, se pueden observar todos los demás resultados obtenidos en el instrumento de medición RC para los atributos de calidad, acoplamiento, complejidad de pruebas y reutilización en el Anexo 12 Representación de los atributos de la

métrica RC. Para obtener más información, se pueden ver: el instrumento y la tabla de resultado ubicada en el Anexo 11 Instrumento de medición de la métrica Relaciones entre clases (RC).

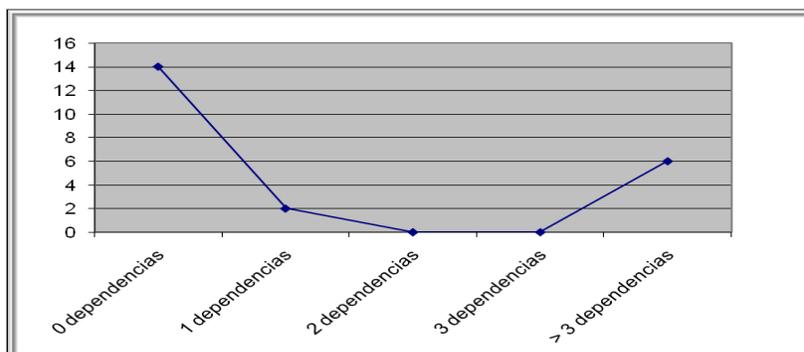


Figura 34 Representación de los resultados de la evaluación de la métrica RC agrupados por la tendencia de los valores

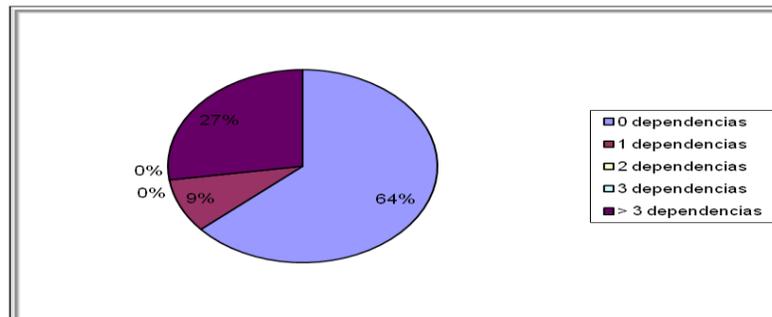


Figura 35 Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos

Al realizar un análisis a los resultados obtenidos al evaluar la métrica Relaciones entre clases (RC) con el instrumento de evaluación, se puede concluir que la implementación de las clases necesarias para dar solución a la problemática tiene una calidad aceptable teniendo en cuenta que de las clases implementadas el 57 % de las clases desarrolladas posee 3 o menos dependencias de otras clases. Además de que el 57 % de las clases no posee acoplamiento alguno y el 57% posee índices de calidad aceptables respecto al mismo atributo de calidad. A la vez que los atributos de calidad Complejidad de Mantenimiento, Cantidad de Pruebas y Reutilización se comportan satisfactoriamente en un 100 % de las clases.

4.4.5.3. Resultados del instrumento de evaluación de la métrica Profundidad de herencia (PH)

En las figuras 47 se representan en por ciento los resultados obtenidos por el instrumento de medición agrupados por nivel. Por otra parte en las figuras 49, 50 y 51, se pueden observar todos los demás

resultados obtenidos en el instrumento de medición PH para los atributos de calidad, complejidad de mantenimiento, de implementación y de reutilización en el Anexo 14 Representación de los atributos de la métrica PH. Para obtener más información, se pueden ver: el instrumento y la tabla de resultado ubicada en el Anexo 13 Instrumento de medición de la métrica Profundidad de herencia (PH).

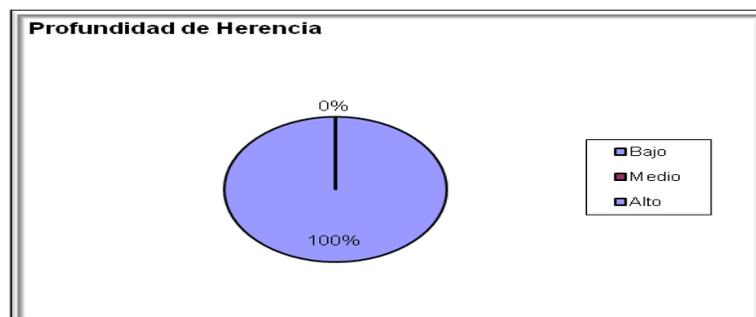


Figura 47 Representación en % de los resultados obtenidos en el instrumento agrupados por nivel

Al hacer un estudio de los resultados obtenidos al evaluar la métrica Profundidad de Herencia (PH) con el instrumento de evaluación, se llega a la conclusión de que la implementación de las clases necesarias para dar solución a la problemática tiene una calidad aceptable teniendo en cuenta que la profundidad de herencia presentes en los subsistemas desarrollados es siempre de 1, un nivel menos que el valor de nivel mínimo posible.

Una vez analizando los atributos de calidad Complejidad de Mantenimiento, se puede observar que los mismos poseen buenos valores producto de que en la implementación se cuenta el nivel dos, como máximo nivel de profundidad en la herencia. No obstante el atributo reutilización no posee indicadores alentadores debido al nivel tan bajo de profundidad que posee la herencia.

4.4.5.4. Resultados del instrumento de evaluación de la métrica Número de Operaciones

Redefinidas para una clase hija (NOR)

En las figuras 52 se representan en por ciento los resultados obtenidos por el instrumento de medición para la redefinición de métodos agrupados en los valores existentes. Por otra parte en las figuras 54, 55 y 56, se pueden observar todos los demás resultados obtenidos en el instrumento de medición NOR para los atributos de calidad, complejidad de mantenimiento, de implementación y de reutilización en el Anexo 16 Representación de los atributos de la métrica NOR. Para obtener más información, se pueden ver: el instrumento y la tabla de resultado ubicada en el Anexo 15 Instrumento de medición de la métrica Número de Operaciones Redefinidas para una clase hija (NOR).

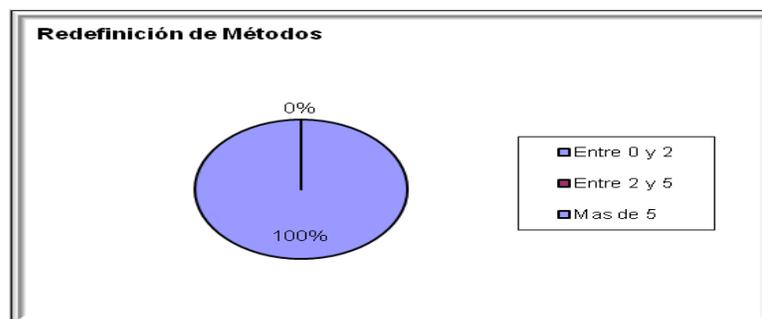


Figura 52 Representación en % de los resultados obtenidos en el instrumento agrupados en los valores existentes

AL realizar un profundo análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica NOR, se puede concluir que la implementación realizada tiene una calidad aceptable teniendo en cuenta que no se redefinen ningún método. Además los indicadores se comportan de forma adecuada para los atributos de calidad Complejidad del Mantenimiento, Cantidad de Pruebas, Violación de la astrictión representada por la superclase.

4.4.6. Matriz de cubrimiento de los parámetros de calidad evaluados con las métricas propuestas

La matriz de cubrimiento o matriz de inferencia de los indicadores de calidad, es el resumen de los resultados obtenidos al aplicar las métricas tratadas en el epígrafe anterior (TOC, RC, PH, NOR). Esta matriz es una representación estructurada de los atributos de calidad y métricas utilizadas para evaluar la calidad de la solución obtenida como se muestra en la tabla 16. La misma permite conocer si el resultado obtenido de la relación atributo/métricas es positivo o negativo. Llevando estos resultados a una escala numérica se le asignará el valor 1 si los resultados son positivos, en caso de sean negativos el valor 0 y si no existe relación alguna tomará valor -1, tal y como se muestra en la tabla 43.

| Calificativo | Valor |
|--------------|-------|
| Positivo | 1 |
| Negativo | 0 |
| Nulo | -1 |

Tabla 43 Clasificación de los parámetros de calidad según impacto en el diseño propuesto

Una vez definidos los datos de dicha relación se determina el promedio de los valores obtenidos de la relación atributo/métrica (solo se toman en consideración las que arrojan un resultado distinto de -1). Este valor representa el impacto que tiene cada atributo en la implementación de la solución determinando si su

impacto fue bueno, regular o malo atendiendo al rango en que se encuentre dicho valor, tal y como se muestra en la tabla 44. Al desarrollar este procedimiento con los resultados que se obtuvieron una vez aplicadas las métricas en la implementación realizada se obtuvo la matriz de cubrimiento que se expone a continuación en la tabla 45.

| Calificativo | Valor |
|--------------|----------------------|
| Malo | $\leq 0,4$ |
| Regular | $> 0,4$ y $\leq 0,7$ |
| Bueno | $> 0,7$ |

Tabla 44 Rangos para evaluar el impacto de los parámetros de calidad en la solución

| Atributos de calidad evaluados | Métricas aplicadas al diseño de la solución propuesta. | | | | |
|--------------------------------|--|----|----|-----|----------|
| | TOC | RC | PH | NOR | PROMEDIO |
| Complejidad implementación | 1 | -1 | 1 | -1 | 1 |
| Reutilización | 1 | 1 | 0 | -1 | 0,67 |
| Acoplamiento | -1 | 1 | -1 | -1 | 1 |
| Complejidad Mantenimiento | 1 | 1 | 1 | 1 | 1 |
| Cantidad Pruebas | -1 | 1 | -1 | 1 | 1 |
| Abstracción | -1 | -1 | -1 | 1 | 1 |
| Calidad Diseño | 1 | -1 | -1 | 1 | 1 |

Tabla 45 Matriz de cubrimiento para los parámetros de calidad evaluados con las métricas aplicadas a la solución

Los datos observados en la tabla anterior, arrojan resultados positivos en casi todos los atributos de la calidad evaluados, exceptuando a la Reutilización, que al aplicar la métrica PH se ve afectado por las características propias del sistema, donde las clases hijas poseen un nivel bajo de herencia, provocando de esta manera una escasa reutilización. Sin embargo, se muestran excelentes resultados en cuanto a la

Complejidad de Mantenimiento y de Implementación. Por todo lo anteriormente expuesto se permitió demostrar las características funcionales más importantes definidas fueron cubiertas en la solución obtenida.

4.5 Conclusiones parciales

En el desarrollo del presente capítulo se realizó un análisis de las pruebas de unidad, en específico las pruebas de camino básico. Se diseñaron casos de pruebas específicos para los principales algoritmos, comprobándose que el flujo de trabajo de los mismos estuvo correcto ya que cumplieron con las condiciones necesarias que se habían planteado.

Finalmente se elaboraron instrumentos de medición para calidad del diseño como tamaño operacional de clase (TOC), relaciones entre clases (RC), profundidad de herencia (PH) y número de operaciones redefinidas para una clase hija (NOR), mediante su empleo permitió afirmar que el diseño realizado se puede valorar de aceptable debido a que el valor de los atributos de calidad, responsabilidad, complejidad de implementación, reutilización, acoplamiento, complejidad del mantenimiento y cantidad de pruebas, fue positivo.

CONCLUSIONES GENERALES

El presente trabajo recoge todo el proceso de implementación del Módulo de Recepción del Subsistema de Inventario del Sistema Cedrux quedando como colofón del mismo:

El análisis realizado de los procesos de la Reserva y los sistemas que existen actualmente para su control, así como la especificación de las tecnologías y herramientas que se utilizaron para el desarrollo de la solución.

La descripción de los aspectos más significativos de la solución como son, la reutilización de componentes, el cálculo de la complejidad ciclomática de algoritmos no triviales así como las principales clases y estándares de codificación utilizados en la implementación de los requerimientos especificados para los procesos de Recepción.

La validación de la solución propuesta mediante el diseño y aplicación de las pruebas de caja blanca, permitiendo validar el correcto funcionamiento de las funciones implementadas.

La evaluación de la implementación a partir de la aplicación de métricas que permitieron analizar el comportamiento de los atributos de calidad de reutilización, facilidad de mantenimiento, complejidad del diseño, complejidad de implementación, cohesión, acoplamiento y cantidad de pruebas, lo cual arrojó resultados positivos que permiten catalogar de aceptable la implementación realizada.

Por todo lo anteriormente expresado se puede concluir que se cumplió con el objetivo general propuesto: Desarrollar la solución para el control del tratamiento de las Reservas Materiales como parte de los procesos de control de inventario.

RECOMENDACIONES

Se recomienda para versiones posteriores:

- Definir lo relacionado con los Anexos de la Reserva que pertenece a las Recuperaciones.
- Definir el proceso de Cambio de Ubicación del mismo tipo de Reserva a nivel nacional.

BIBLIOGRAFÍA

- Shang, S. (2000).** A Comprehensive Framework for Classifying the Benefits of ERP Systems.
- DeepInder, B. M. (2004).** An integrative framework for the assimilation of Enterprise Resource Planning . In B. M. DeepInder, *An integrative framework for the assimilation of Enterprise Resource Planning* (Vols. Vol. 44, Iss. 3; pp. 81, 10.). Spring .
- Edgarin. 2006.** Edgarin.com. [En línea] 2006. <http://edgarin.com/2006/07/21/modelo-vista-controlador/>.
- B., Ing. Alexander Oré. 2009.** CalidadSoftware.com. [En línea] 2009. [Citado el: 19 de mayo de 2011.] http://www.calidadsoftware.com/testing/pruebas_unitarias1.php.
- 2009.** Desarrollo Web. *Desarrollo Web*. [En línea] 2009. <http://desarrolloparaweb.blogspot.com/2010/04/tecnicas-para-conocer-nuestra-audiencia.html>.
- Desarrolloweb.com. 2009.** Desarrolloweb.com. [En línea] 2009. <http://www.desarrolloweb.com/articulos/1622.php>.
- dinamica, Técnicas de evaluación. 2009.** Técnicas de evaluación dinamica. [En línea] 2009. [Citado el: 21 de mayo de 2011.] <http://www.lsi.us.es/docencia/get.php?id=361>.
- EcuRed. 2011.** EcuRed. Conocimientos con todos y para todos. [En línea] 2011. http://www.ecured.cu/index.php/Flujo_de_Trabajo_de_Implementaci%C3%B3n.
- Eduardo Rivera Alva . 2011.** Scribd. [En línea] 2011. [Citado el: 15 de mayo de 2011.] <http://es.scribd.com/doc/7884665/Arquitectura-de-Software-II-Diagrama-de-Componentes-y-Despliegue>.
- Fernández González, Mairelys y Zorrilla Rivera, Osley. 2010.** *Diseño e implementación del componente Ajuste al Costo del Subsistema Costos y Procesos del Sistema Integral de Gestión de Entidades Cedrux*. Ciudad de la Habana : s.n., 2010.
- Fuentes, Alien Fernández. 2009.** *Implementación del Módulo Inventario Físico del Subsistema Inventario del Sistema Integral Cedrux*. Ciudad de la Habana : s.n., 2009.
- Galán Ramírez, Yuliet y Bernal Vidal, Néstor. 2008.** *Proceso de Planificación por Objetivos en las entidades de las FAR*. Ciudad de la Habana : s.n., 2008.
- Gestión, Centro de Soluciones de. 2009.** *Modelo Conceptual del sibsistema de Dirección por Objetivo*. La Habana : s.n., 2009.
- Gomez, Diego. 2008.** DosIdeas. [En línea] 15 de noviembre de 2008. [Citado el: 19 de mayo de 2011.] <http://www.dosideas.com/noticias/metodologias/320-como-entender-la-complejidad-ciclotomica.html>.
- Isis, Sistema. 2010.** Sistema Isis. [En línea] 2010. http://www.sistemaisis.com/erp_caracteristicas.html.
- 2009.** Minería Web. *Minería Web*. [En línea] 2009. http://bvs.sld.cu/revistas/aci/vol16_4_07/aci111007.html.
- Rodríguez, Jorge. 2006.** Pruebas unitarias. [En línea] Marzo de 2006. [Citado el: 18 de mayo de 2011.] <http://blog.continuum.cl/wp-content/uploads/2008/08/pruebas-unitarias.pdf>.

- 2010.** Sistema Isis. [En línea] 2010. <http://www.sistemaisis.com/erp-produccion.htm>.
- Slideshare. 2011.** slideshare. [En línea] 2011. <http://www.slideshare.net/guest0a6e49/sem-8-modelo-de-analisis-1362411>.
- Soft, Golden. 2010.** Golden Soft. [En línea] 2010. http://www.goldensoft.com/index.php?option=com_content&view=article&id=65&Itemid=133.
- Softland. 2009.** Softland. [En línea] 2009. http://www.softland.cl/productos/erp/back_fact.asp.
- software, Gestion y pruebas de calidad del. 2005.** pruebasdesoftware. [En línea] 2005. [Citado el: 18 de mayo de 2011.] <http://www.pruebasdesoftware.com/laspruebasdesoftware.htm>.
- Software, Ingeniería del. 2009.** Módulo 2. Tema 12: Modelo de Implementación. [En línea] 2009. [Citado el: 15 de mayo de 2011.] <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>.
- Visconti, Marcello y Astudillo, Hernán. 2009.** Fundamentos de ingeniería de software. [En línea] 2009. http://www.google.com.cu/url?sa=t&source=web&cd=2&ved=0CCMQFjAB&url=http%3A%2F%2Ffeisc.univ-alle.edu.co%2Fmaterias%2FMaterial_Desarrollo_Software%2FDISCLASES_A12.pdf&rct=j&q=modelo%20de%20clases%20del%20dise%C3%B1o&ei=bo21TYsbyLW2B5jZxecO&usq=AFQjCNFDOAXOJ1.
- Bizagi, Corporate. 2006.** *BPMN Business Process Modeling Notation*. Bogotá, Colombia : s.n., 2006.
- datos, Sistema Gestor de base de. 2004.** Sistema Gestor de base de datos. [En línea] 2004. [Citado el: 15 de junio de 2011.] http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php.
- Desarrolloweb.com. 2010.** Programación PHP. [En línea] 2010. [Citado el: 15 de junio de 2011.] <http://www.desarrolloweb.com/manuales/12/>.
- Diccionario. 2008.** PANHISPANICO DE DUDAS (2ª ED). [En línea] 2008. [Citado el: 15 de junio de 2011.]
- Díez González, Oscar. 2006.** *Safety y Requisitos No Funcionales*. 2006.
- Esser, Stefan. 2009.** *Secure Progaming with the Zend-Framwork*. 2009.
- Ferré Grau, Xavier y Sánchez Segura, María Isabel. 2009.** Desarrollo Orientado a Objetos con UML. [En línea] 2009. [Citado el: 14 de junio de 2011.] <http://msanchez.usach.cl/comprimido/OBJETOS.pdf>.
- Firefox, Mozilla. 2009.** Mozilla Firefox. [En línea] 2009. [Citado el: 15 de junio de 2011.] <http://www.mozilla.com/es-ES/firefox/>.
- Frederick, Shea, Ramsay, Colin y Cutter Blades, Steve. 2009.** *Learning Ext JS*. 2009.
- Gómez Baryolo, Oiner, Morejón Borbón, Yoandry y García Trejo, Darien. 2010.** *Arquitectura tecnológica para el desarrollo de software*. Ciudad Habana : s.n., 2010.
- HTML, Lenguaje. 2002.** Desarrolloweb.com. [En línea] 2002. [Citado el: 15 de Junio de 2011.] <http://www.desarrolloweb.com/articulos/711.php>.
- Hurtado Bustamante, Diana Paola, y otros. 2009.** *Patrones GRASP*. 2009.

- Isis. 2010.** Sistema Isis. [En línea] 2010. [Citado el: 13 de junio de 2011.] http://www.sistemaisis.com/erp_caracteristicas.html.
- MAEZTU, IÑIGO. 2011.** TIC euskadi. [En línea] 12 de enero de 2011. [Citado el: 14 de junio de 2011.] http://www.tic-euskadi.com/2011/01/penetracion-de-los-erp-en-las-empresas_20.html.
- Ministro, Consejo de. 2006.** *Reglamento del Sistema de Reservas Materiales*. La Habana : s.n., 2006.
- Olivares Rojas, Juan Carlos. 2009.** *Patrones de diseño*. 2009.
- paniagua, Angel Barbero. 1999.** Tutorial de XML. [En línea] 1999. [Citado el: 15 de junio de 2011.] <http://www.dat.etsit.upm.es/~abarbero/curso/xml/xmltutorial.html>.
- Paradigm, Visual. 2005.** Visual Paradigm. [En línea] 2005. [Citado el: 15 de junio de 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
- Patrones. 2009.** Software Guisho. [En línea] 2009. [Citado el: 15 de junio de 2011.] <http://software.guisho.com/patrones-de-diseno>.
- Pérez, Javier Eguíluz. 2009.** *Introducción a JavaScript*. 2009.
- . 2008. *Introducción a XHTML*. 2008.
- PostgreSQL. 2011.** The World's most advanced open source database. [En línea] 2011. [Citado el: 15 de junio de 2011.] <http://www.postgresql.org/>.
- Repo. 2011.** Repositorio. [En línea] 2011. [Citado el: 15 de junio de 2011.] https://repositorio.ceige.prod.uci.cu/svn/repo/CSGPROD/Expediente_Proyectos/Quimefa/version%201.0/1%20ingenieria/ingenieria%201.0/1.1%20requisitos/descripci%C3%B3n%20de%20requisitos/Inventario/.
- Repo1. 2011.** Repositorio. [En línea] 2011. [Citado el: 15 de junio de 2011.] <http://websvn.ucid.uci.cu/listing.php?repname=5.+Centro+MD&path=%2FLinea+Contabilidad+material%2FProyectos%2FInventario%2F&#a63052b55affd5837ffa195a7fdb02ba>.
- Repositorio. 2011.** Repositorio. [En línea] 2011. [Citado el: 15 de junio de 2011.] https://repositorio.ceige.prod.uci.cu/svn/repo/CSGPROD/Expediente_Proyectos.
- S.A., Mistral Caribe Holding. 2011.** Mistral Caribe. [En línea] 2011. [Citado el: 6 de junio de 2011.] http://www.mistralcaribe.com/index.php?option=com_content&view=article&id=126&Itemid=132.
- Smith, Lukas. 2009.** *Introduction to the Doctrine Object Relational Mapper*. 2009.
- Softland. 2009.** Softland. [En línea] 2009. [Citado el: 14 de junio de 2001.] <http://www.softland.cl/productos/default.asp?ap=productos&pd=erp>.
- software, Gestion y pruebas de calidad del. 2005.** pruebasdesoftware. [En línea] 2005. [Citado el: 18 de mayo de 2011.] <http://www.pruebasdesoftware.com/laspruebasdesoftware.htm>.
- Software, Ingeniería de. 2010.** *Capítulo 1. Herramienta Case*. 2010.

Subversion. 2004. Subversion. [En línea] 2004. [Citado el: 15 de junio de 2011.]
<http://polaris.dit.upm.es/~rubentb/docs/subversion/TutorialSubversion/index.html>.

Unidad de Compatibilización, Integración y Desarrollo de Software para la defensa (UCID). 2009.
Proceso de Desarrollo y Gestión de Proyectos de Software. Ciudad Habana : s.n., 2009.

GLOSARIO DE TÉRMINOS GENERAL

Procesos: Un proceso es un conjunto de actividades o eventos (coordinados u organizados) que se realizan o suceden (alternativa o simultáneamente) bajo ciertas circunstancias con un fin determinado. Este término tiene significados diferentes según la rama de la ciencia o la técnica en que se utilice. En la informática, un proceso es un programa en ejecución. Los procesos son gestionados por el sistema operativo

Reserva: Conjunto de cosas disponibles para ser usadas en el momento oportuno o para obrar de una manera determinada.

Software: Se conoce como software al equipamiento lógico o soporte lógico de una computadora digital; comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos, que son llamados hardware.

Área de MPV: Es el área dentro de una entidad que se dedica a la venta de mercancías.

Transferencia entre áreas: Es el cambio de materiales de un área a otra sin necesidad de una factura.

Inventario físico: Se da el nombre de inventario de mercancía a la verificación o confirmación de la existencia de los materiales o bienes patrimoniales de la empresa. En realidad, el inventario es una estadística física o conteo de los materiales existentes, para confrontarla con la existencia anotadas en los ficheros de existencias o en el banco de datos sobre materiales.

Facturación: La factura es un documento que refleja toda la información de una operación de compraventa. La información fundamental que aparece en una factura debe reflejar la entrega de un producto o la provisión de un servicio, junto a la fecha de devengo, además de indicar la cantidad a pagar. El proceso de la facturación, es donde todas las facturas se guardan en los registros y se lleva un control de este mediante la aplicación.

Cambio definitivo de surtido por riesgo de vencimiento: Se activa cuando existen en la Reserva Estatal lotes de productos que caen en riesgo de vencimiento y no se pueden rotar por alguna razón. Se solicita al INRE cambiar las cantidades de esos productos de manera definitiva por otro surtido similar. El INRE factura los productos con riesgo de vence que se le dan entrada a MPV a través de un IR. El surtido sustituto se factura en MPV y se le da entrada en la RESE por un IR.

Cambio temporal de surtido por calidad: Cuando llega un plan de aviso emitido por el Centro para el Control Estatal de la Calidad de los Medicamentos (CECMED) o para la Empresa Comercializadora de Medicamentos (EMCOMED), se verifica si existe algún producto de la Reserva Estatal (RESE) en el plan de aviso. Si hay, se facturan los productos que tienen problemas de calidad de RESE para el Área de MPV. Se dan entrada en el inventario corriente por un IR y se realiza a continuación una transferencia entre áreas de MPV para Retenidos. Si tengo otros lotes sin problemas de calidad en MPV, se factura ese lote y se le da entrada por un Informe de Recepción (IR) en la RESE. En caso contrario, se realiza la

solicitud de cambio temporal de surtido por calidad. Cuando llega la aprobación se ejecuta la facturación de MPV y se da entrada a RESE por IR.

Pago de préstamo: Las deudas se generan por dos conceptos: préstamos y cambios de surtidos. Se hace una autorización de entrega que se convierte en factura desde reserva hacia el área de mpv. Para el caso de RESE: Se factura de MPV con un gravamen impuesto por el INRE (1 al 10%) del préstamo que se está pagando. Se da entrada en la RESE por un IR. Para el caso de la Reserva Movilizativa (RESEM): Se hace transferencia de salida de MPV y transferencia de entrada a RESEM. La carta de aprobación de préstamo llega con la copia de autorización del INRE. Con esos documentos, es que se procede a realizar la facturación para el caso de la RESE. En la carta viene ya impresa la fecha en que debe pagarse el préstamo y el gravamen que se debe aplicar.

Rotación de lotes: La rotación de lotes se realiza de manera automática (durante las recepciones) y dirigida (cuando el especialista de la reserva material la orienta realizar por un análisis previo realizado).

Rotación anticipada: Productos de la RESEF con riesgos de vence y no hay manera de rotarlos. Se factura de la RESEF y se da entrada en MPV con un IR. Para pagar esta rotación anticipada se hace una factura y en las Observaciones se explica que es por el concepto de Pago de una rotación anticipada.

Cambio de surtido: Son movimientos entre los mismos tipos de reserva, y no generan nada contable. Dirigidos por el INRE.

Factura: Su objetivo es formalizar las ventas de productos y prestaciones de servicios que se efectúen, así como las entregas de productos en consignación o en depósito, que no son sujetas de la Factura Comercial. Se utiliza además, para formalizar las ventas de activos fijos tangibles y de productos para efectuar devoluciones.

Inventario: Controla los medios que posee una organización. Se encarga de inspeccionar las existencias de productos, dándole entrada y salida del almacén a los mismos, regulando el flujo de mercancía en el almacén, con el fin de hacer más rentable su posesión y garantizar en cierto grado el éxito de la organización.

Stock: Término utilizado en los almacenes. En otras palabras, es todo lo referente a los bienes que la empresa tiene y que sirven como materia prima para la realización de sus productos, lo referente a Inventario y a Existencias, como concepto para contabilidad.

Informe de recepción: Documento que se obtiene a partir de una Recepción Ciega guardada, en este también se refleja todos los datos relacionados con el o los productos recepcionados.

Tarjeta de estiba: Documento que refleja los datos generales de un producto, código CUP, descripción, unidad de medida, etc., así como todos los movimientos que ha tenido el producto dentro del almacén, tales como fecha del movimiento, tipo de movimiento, existencias antes y después de los movimientos realizados, entre otros.