

# Universidad de las Ciencias Informáticas

## Facultad 2



**Título:** Implementación del módulo de inventario del subsistema Activo Fijo Tangible del Sistema Integral de Gestión CEDRUX.

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

**Autor:**

Cadete Carlos Fábrega Avila.

**Tutor(es):** Tte. Ing. Meylin Martínez Chong.

**Cotutor(es):** Ing. Osnier Ramírez Alea.

Ciudad de La Habana, 2010.

“Año 52 de la Revolución”

## *Pensamiento*



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

*Ernesto Che Guevara*

## DECLARACIÓN DE AUTORÍA

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

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

**Cadete: Carlos Fábrega Avila**

**Tte.Ing. Meylin Martínez Chong**

\_\_\_\_\_  
Firma del Autor

\_\_\_\_\_  
Firma del Tutor

## *Datos de contacto*

**Tutor:** Tte. Ing Meylin Martínez Chong.

**Clasificación:** Profesional.

**Correo electrónico:** [meylin@uci.cu](mailto:meylin@uci.cu)

**Síntesis del Tutor:** Ingeniero informático graduado en la CUJAE, año 2005.

**Co-Tutor:** Ing. Osnier Ramírez Alea.

**Clasificación:** Profesional.

**Correo electrónico:** [oramirez@uci.cu](mailto:oramirez@uci.cu)

**Síntesis del Co-Tutor:** Instructor recién graduado de ingeniero en Ciencias Informáticas en la UCI (2009).

## *Agradecimientos*

A mis padres por confiar en mí, a mis abuelos, mis hermanos, a mis tíos, a mi padrastro, en general a toda mi familia por ayudarme a cumplir mi sueño.

A mis tutores y a todo el equipo de desarrollo de la línea de logística por su ayuda y dedicación.

A mis amigos por los consejos y por la seguridad que siempre han tenido en mí.

Al comandante en jefe Fidel Castro Ruz por darme la oportunidad de convertirme en profesional.

A todos, muchas gracias

## *Dedicatoria*

A mi abuelo Mario que por cuestiones de la vida no se encuentra físicamente entre nosotros.

## *Resumen*

En el presente trabajo de diploma se expone la implementación del módulo de Inventario del Subsistema Activo Fijo Tangible del Sistema Integral de Gestión Cedrux. Actualmente la situación general de los procesos de gestión de las entidades presupuestadas y empresariales a escala nacional está afectada por la existencia de sistemas informáticos que no cumplen con las expectativas de las nuevas tecnologías y la misión de nuestro país en el desarrollo de sus empresas. Proteger los recursos de las empresas así como llegar a la exactitud y confiabilidad de la información económica y contable, ha llevado a la necesidad de mejorar los procesos de gestión de las áreas que las conforman, utilizando plataformas confiables y eficientes. Con el desarrollo de éste módulo se deberá lograr optimizar el proceso de gestión de inventario de Activos Fijos Tangibles, disminuir los costos totales de operación, compartir información entre todos los componentes de la empresa y disminuir el margen de contaminación y repetición de la información. Una vez finalizado el módulo se prueba la aplicación utilizando pruebas unitarias.

## Tabla de contenido

<i>INTRODUCCIÓN</i> .....	1
<i>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</i> .....	5
Introducción .....	5
1.1. Conceptos generales.....	6
1.2. Sistemas automatizados existentes .....	7
1.3. Modelo de desarrollo de software basado en componentes.....	14
1.4. Herramientas y tecnologías utilizadas .....	16
Conclusiones parciales. ....	33
<i>CAPÍTULO 2: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA</i> .....	34
Introducción.....	34
2.1. Objeto de estudio .....	34
2.2. Problema .....	34
2.3. Situación problemática .....	34
2.4. Objeto de automatización .....	35
2.5. Información que se maneja. ....	35
2.6. Propuesta de sistema.....	35
2.7. Especificación de los requisitos de software .....	36
2.8. Principales Interfaces .....	38
Conclusiones Parciales .....	40
<i>CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA</i> .....	41
Introducción.....	41
3.1. Implementación.....	42
3.1.1. Matriz de Integración de Componentes Interna. ....	43
3.1.2. Matriz de Integración de Componentes Externa.....	43
3.2. Prueba .....	44
3.2.1. Resultado de las Pruebas.....	70
Conclusiones Parciales .....	71
<i>CONCLUSIONES GENERALES</i> .....	72
<i>RECOMENDACIONES</i> .....	73
<i>REFERENCIAS BIBLIOGRÁFICAS</i> .....	74
<i>BIBLIOGRAFÍA</i> .....	76
<i>GLOSARIO DE TÉRMINOS</i> .....	78



# INTRODUCCIÓN

En la actualidad las Tecnologías de la Información y las Comunicaciones (TIC) son casi imprescindibles en las actividades de la vida cotidiana. En el país se llevan a cabo un grupo de proyectos con vista a lograr la informatización de la sociedad cubana.

Uno de los proyectos más grandes y emprendedores que se llevan a cabo en Cuba, es la Universidad de las Ciencias Informáticas (UCI), que por sus características especiales tiene una misión muy importante: desarrollar la industria del software cubano, con ese fin, la Revolución ha puesto en sus manos los medios necesarios para construir el futuro del país.

En la UCI se trabaja actualmente en un software llamado CEDRUX<sup>1</sup>, que controlará la actividad económica, financiera y contable del país, está diseñado para modelar y automatizar la mayoría de los procesos en las entidades. Lo más destacable del producto es que unifica y ordena toda la información de la entidad en un solo lugar, de este modo cualquier suceso queda a la vista de forma inmediata, posibilitando la toma de decisiones de forma más rápida y segura, acortando los ciclos productivos. Con él se tendrá la entidad bajo control y se incrementará la calidad de los servicios y productos. La implantación del CEDRUX conlleva la eliminación de barreras inter departamentales, la información fluye de forma que elimina la improvisación por falta de información. Este software cuenta con varias áreas dentro de las cuales se encuentra Logística la cual tiene un subsistema llamado Activo Fijo Tangible (AFT) que necesita un módulo de Inventario, el cual se encargaría de inventariar todos los activos fijos de la entidad.

Hay que destacar que la gestión de inventarios es un aspecto crítico de la administración de recursos de las empresas. Los inventarios surgen porque permiten reducir los costes en la actividad empresarial, y pueden servir tanto para regular los procesos de producción como para determinar el nivel de mercancías almacenadas. Es importante señalar que el control de los mismos posibilita la optimización de los tiempos, el mantenimiento del nivel competitivo y la protección contra aumentos de precios y escasez de materia prima. (1)

---

<sup>1</sup> **Cedrux:** Vocablo formado por la unión de las palabras “Cedro” (fortaleza, resistencia) y “Linux” (tecnología libre).

En el país hay varios software que cuentan con el módulo de inventarios pero no resultan soluciones factibles para las entidades cubanas debido a que: fueron desarrollados sobre plataformas de software propietario, lo que implica incrementos de gastos en licencias de uso y mantenimiento del software. Además, las soluciones nacionales constituyen aplicaciones de escritorio lo que trae como desventaja que el usuario deba instalar la aplicación en cada estación de trabajo. Son productos que se caracterizan por abordar solamente partes del problema de la gestión de la empresa o la unidad presupuestada, no soportan mecanismos estándares de integración con otras aplicaciones donde casi ninguno bajo conceptos de informática multicapa y distribuida en la red. Por otra parte los sistemas internacionales desde el punto de vista de la solución serían capaces de resolver el problema del módulo Inventario Físico debido a su alto nivel de configuración y servicios que proveen, pero tienen la desventaja de que algunos de ellos utilizan tecnologías que no son accesibles a Cuba debido a las restricciones impuestas por Estados Unidos. Algunos sistemas como OpenBravo están basados en la plataforma J2EE<sup>2</sup> cuya máquina virtual es propiedad de SUN, empresa norteamericana, aunque ha comenzado a liberar el código sigue estando bajo las leyes de su gobierno; además J2EE requiere un consumo de memoria elevado en comparación con PHP/Apache. Como otra desventaja aparece que el diseño de estos ERP ha sido para empresas capitalistas que tienen un modelo de gestión y de procesos muy diferente a las empresas o unidades presupuestadas cubanas donde la economía es centralizada y operan otros mecanismos. Por último, el resto de los softwares propietarios no constituyen una opción viable pues representan gastos muy elevados al país por conceptos de licencias y mantenimiento.

Analizado lo planteado anteriormente surge como **Problema a resolver**: La no existencia de una herramienta que gestione la información referente a los procesos de Inventario del subsistema del Subsistema Activo Fijo Tangible. Definiendo como **Objeto de estudio** los procesos para el Control de Inventario. Como **Campo de acción** los procesos de Inventario Físico para los activos fijos tangibles.

Para la realización de esta investigación se ha planteado la siguiente **idea a defender**:

---

<sup>2</sup> J2EE: Java 2 Platform, Enterprise Edition.

Si se realiza la implementación del componente Inventario Físico, perteneciente al subsistema Activo Fijo Tangible del Sistema de Gestión Integral Cedrux, se obtendrá un producto funcional para la gestión de dichos procesos.

El **Objetivo General** es Implementar los procesos de inventario del Subsistema Activo Fijo Tangible para el entorno empresarial cubano, en correspondencia con las nuevas concepciones de informatización.

De este se derivan los siguientes **Objetivos específicos**:

- Realizar el diseño del marco teórico de la investigación.
- Analizar los procesos de inventario en los AFT así como las herramientas que se utilizarán para el desarrollo de la solución.
- Implementar las interfaces a partir del prototipo entregado por los analistas.
  
- Implementar la capa de negocio que dé respuesta a los requisitos propuestos por los analistas.
  
- Implementar la capa de acceso a datos.
  
- Implementar las validaciones y excepciones.
  
- Implementar los servicios incluidos dentro de sus responsabilidades que se necesiten para la implementación de otros módulos y componentes.
  
- Validar el resultado obtenido con pruebas unitarias.

Los siguientes **métodos teóricos** sustentan la investigación:

Histórico-Lógico: Su empleo permitió el desarrollo evolutivo y coherente en el estudio de la metodología orientada a objetos, patrones de diseño, herramientas de desarrollo de software y sistemas ERP para el desarrollo de los artefactos que proponen los flujos estudiados.

Analítico-Sintético: Permitió integrar y descomponer el conocimiento, descubriendo las relaciones para la utilización de artefactos propuestos por constituir este método una unidad dialéctica, determinando los aspectos esenciales y el arribo a conclusiones prácticas y teóricas.

Modelación: Su utilización permitió crear abstracciones que explican la realidad, por ejemplo, todos los modelos y diagramas presentados

### **Estructuración del contenido**

Capítulo 1. Fundamentación Teórica: Incluye un estado del arte del tema tratado, a nivel internacional, nacional y en la Universidad, de las tendencias, técnicas, tecnologías, metodologías y software usados en la actualidad, los que se utilizarán como guía para darle solución al problema que nos enfrentamos.

Capítulo 2. Características del sistema: En este capítulo se hará un análisis del problema y situación problemática. Se identificarán los procesos que requieren automatización para su descripción. Así como también describiríamos los sistemas automatizados que existen en la empresa. Plantearemos nuestra propuesta de sistema y listaremos los requisitos funcionales y no funcionales.

Capítulo 3. Implementación y Prueba: Refleja los aspectos relacionados con la construcción de la solución propuesta. Se representa el modelo de implementación, o sea, el diagrama de componentes. Además de reflejar el modelo de prueba con la descripción de los casos de prueba.

# *Capítulo 1: Fundamentación Teórica*

## **Introducción**

El ambiente competitivo que se vive en el ámbito empresarial actualmente requiere promover los procesos y actividades de negocio que generan las ventajas competitivas de las compañías ante sus más fuertes competidores.

Por esto, desde hace ya varios años, se ha dado mayor importancia a las Tecnologías de Información y su alineación con las estrategias del negocio para mejorar sus procesos clave. Prueba de ello, es el incremento tan sustancial de adquisiciones de paquetes de software empresariales tales como el ERP (Enterprise Resource Planning), con el cual los directivos de las compañías esperan tener integradas todas las áreas o departamentos de la compañía que apoyan para la generación de sus productos y servicios.

Hoy más que nunca las empresas requieren de herramientas que les proporcionen control y centralización de su información, esto con el fin tomar las mejores decisiones para sus procesos y estrategias de negocios. Los ERP son una solución robusta para aquellas empresas que buscan una solución universal a la centralización de su información, estos sistemas automatizan los procesos para el Control de Inventario de AFT. Además de los ERP existen otros sistemas tanto nacionales como internacionales que poseen módulos de Inventarios de AFT los cuales han sido analizados para obtener sus ventajas, desventajas y las herramientas utilizadas para su desarrollo.

## 1.1. Conceptos generales

### ¿Qué es un ERP?

Los sistemas de Planificación de Recursos Empresariales (Enterprise Resource Planning, ERP) son sistemas de gestión de información gerenciales que integran, automatizan y manejan muchas de las acciones asociadas con las operaciones de producción y distribución de una compañía dedicada a la producción de bienes o servicios.

Los ERP pueden intervenir en el control de muchas actividades de negocios, tales como ventas, compras, entregas, pagos, producción, contabilidad, administración de inventarios y de recursos humanos.

Funcionan en todo tipo de empresas. Para ello, integran todos los departamentos funcionales de la empresa, herramientas de mercadotecnia y administración estratégica en un solo sistema. (2)

### Características de los sistemas ERP.

Existen tres características principales que distinguen a un sistema de Gestión empresarial, de una simple aplicación de gestión.

- **Integración:** El objetivo de un sistema ERP es integrar todos los procesos de la empresa, entendiéndola como una serie de áreas que se relacionan entre sí. Este enfoque permite una mayor eficiencia, reducción de tiempo y costes.
- **Modularidad:** Cada área funcional de la empresa se corresponde con un módulo del sistema de gestión. Estos módulos aunque independientes comparten información entre sí mediante una base de datos centralizada, lo que facilita la personalización y adaptabilidad por una lado, y por otro la facilidad de integración.
- **Adaptabilidad:** Gracias a la modularidad y capacidad de integración de las funcionalidades un sistema ERP es fácilmente adaptable a las necesidades de cada empresa, permitiendo una total configuración. (3)

### ¿Qué es un Inventario?

El inventario de AFT consiste en verificar físicamente los medios con que cuenta la organización. Su finalidad es llevar a cabo un registro de la existencia, cantidad, características, condiciones de uso, valor de los medios y las personas responsables de su manejo. (4)

## **¿Qué son los Activos fijos tangibles?**

Se denomina activos fijos a los bienes de larga vida adquiridos para usarlos en la operación de la empresa y que su destino no es la venta. Son ejemplos de éstos, los edificios, los muebles y equipos de oficina, los medios de computación, los medios de comunicaciones, y los medios de transporte de todo tipo. (5)

## **Situación actual de los procesos**

En la actualidad no existe en Cuba un sistema informático integral de gestión que cumpla con la totalidad de los requerimientos de funcionalidad, interoperabilidad y seguridad que espera el gobierno cubano de una solución de este tipo, de manera que pueda ser utilizada como herramienta para potenciar el cumplimiento de las funciones de las entidades a todos los niveles con un máximo de racionalidad y control de los recursos financieros, materiales y humanos. (6)

### **1.2. Sistemas automatizados existentes**

#### **Sistemas nacionales**

##### **Inventario en la UCI. Assets**

El *Sistema de Gestión Integral (ASSETS)*, no es más, que un sistema multiusuario que se monta en una plataforma de servidores SQL, dividido en módulos económicos que trabajan en conjunto para el control de las actividades económica, financiera y contable sobre los medios materiales y financieros.

El Módulo Inventario del Assets divide los medios en dos tipos de cuentas: Activos Fijos y Útiles Herramientas. Estas últimas son las que se utilizan para realizar las actividades de mantenimiento, talleres, almacenes, así como los equipos de protección física. Comprende entre otros, herramientas manuales, artículos de protección personal, utensilios de laboratorios, mini calculadoras, utensilios menores de cocina. Los medios que pertenecen a esta cuenta tienen poco tiempo de duración.

##### **SISCONT5**

El sistema se aviene a las definiciones y conceptos del Ministerio de la Industria Básica aunque por las acciones contables financieras que permite puede ser utilizado en otras entidades nacionales.

Está formado por varios Módulos:

- Efectivo en Caja y Bancos.
- **Inventarios.**
- Cobros y Pagos.
- Facturación.
- **Activos Fijos Tangibles.**
- Nóminas.
- Contabilidad.

Puede ser explotado en régimen mono usuario y multiusuario. Se define para mono entidad y multientidad, en esta última existe el control de su acceso para las entidades en un mismo equipo de cómputo como servidor.

El módulo de Inventario maneja toda la información referida al Submayor de Inventarios de la entidad, garantizando el cuadro permanente con las respectivas cuentas de la Contabilidad General. El sistema esta preparado para controlar el saldo de cada material en dos monedas, a partir de los procedimientos vigentes en el país en cuanto a política monetaria.

Permite el tratamiento de las contabilizaciones de forma transaccional, por resúmenes diarios o de forma personalizada según se defina por parámetros. Incorpora tratamiento de lotes a los productos y dos tipos de valoración de las cuentas FIFO<sup>3</sup> y Promedio Ponderado. (7)

### **Versat-Sarasola**

Es un producto cubano. Es el primer Sistema Integral de gestión de contabilidad certificado, desarrollado para la gestión económica eficaz y fiable. Actualmente es utilizado en Cuba en alrededor de 200 entidades de varias provincias y en lo adelante será introducido en más de dos mil 500 unidades presupuestadas. Este sistema integrado cuenta con un conjunto de 12 módulos entre los que se encuentran:

- Configuración y seguridad.
- Contabilidad general y de gastos.
- Costos y procesos.
- Análisis económico empresarial.
- Control de activos fijos.

---

<sup>3</sup> FIFO: First in first out



- Finanza y caja.
- Planificación y presupuestos.
- **Control de inventarios.**
- Pago de salario.
- Paquete de gestión.
- Contratación.
- Facturación.

En el módulo de **Control de Inventarios** se definen formatos del clasificador<sup>4</sup> de productos para lograr una uniformidad en el registro y la agregación de información en los reportes de salida, se conceptualizan los movimientos para lograr una información amplia sobre los orígenes y destinos de los recursos. Permite el control de las existencias y movimientos en diferentes monedas. Muestra el cuadro diario de cada uno de los almacenes por las diferentes cuentas. Ofrece la posibilidad de duplicar documentos para agilizar los pases de los mismos y lograr que un mismo documento se convierta en otro con solo adicionar un mínimo de información, realiza un control de las existencias y movimientos por custodios y se emiten diferentes reportes e información de utilidad para la correcta administración de los recursos materiales. (8)

**Control de Activos Fijos:** Recoge las operaciones normales que en esta actividad se realizan (altas, bajas, modificaciones de los medios). También se le ha incorporado otras opciones novedosas, dentro de las cuales se encuentran la posibilidad de amortizar los activos por los 4 métodos más conocidos internacionalmente. Posee un asistente (wizard) para la configuración del subsistema, la tarea más compleja de este módulo.

### **Características:**

- Es una aplicación de escritorio.
- Implementado en Delphi.
- Trabaja sobre el sistema operativo Windows.
- Soporte para base de datos SQL Server 2000.

---

<sup>4</sup> Clasificador: nomenclador.

## **Rodas XXI**

Sistema multiempresa y multiusuario creado por CITMATEL para la automatización de la gestión empresarial. Contiene diferentes módulos que pueden usarse integrados o independientes:

- Contabilidad.
- Efectivo caja y banco.
- Nóminas.
- **Activos Fijos Tangibles.**
- **Inventarios.**
- Cobros y pagos.
- Facturación.
- Finanzas.
- Tele-cobranza.

Además, cuenta con el módulo Administrador, que brinda mayor integralidad al sistema y garantiza facilidades adicionales durante su instalación y explotación.

El módulo de **Activos Fijos Tangibles** de RODAS XXI le permite tener un control detallado de los activos fijos de su entidad, realizando en el mismo momento que se registra un movimiento, su contabilización. Se pueden realizar todo tipo de operaciones de activos fijos con facilidad en el momento que se desee, generando el documento asociado al movimiento de que se trate de forma automática previa configuración del sistema para ello. Permite el control por separado de los activos fijos que se encuentran en almacén de los que se encuentran en explotación. Es posible además realizar ajustes a los activos fijos con facilidad mediante una opción que brinda el sistema. **(11)**

### **Valoración crítica.**

Una vez analizados los sistemas implantados en Cuba, se concluye que no resultan soluciones factibles para las entidades cubanas debido a que fueron desarrollados sobre plataformas de software propietario, lo que implica incrementos de gastos en licencias de uso y mantenimiento del software. Además, las soluciones nacionales constituyen aplicaciones de escritorio lo que trae como desventaja que el usuario deba instalar la aplicación en cada estación de trabajo. Son productos que se caracterizan por abordar solamente partes del problema de la gestión de la empresa o la unidad presupuestada, no soportan

mecanismos estándares de integración con otras aplicaciones donde casi ninguno bajo conceptos de informática multicapa y distribuida en la red.

## **Sistemas internacionales**

### **Condor**

Sistema automatizado de alta complejidad y seguridad que abarca todos los aspectos del proceso contable de una entidad, tales como la dualidad de moneda y el pago por resultados. Está formado por varios módulos:

- **Activos fijos.**
- Contabilidad general.
- Nóminas.
- **Control de inventarios.**
- Condexce.
- Recursos humanos.

Este brinda mayor autonomía al cliente para efectuar cambios de estructura sin necesidad de la intervención de especialistas, quedando registrados de forma que puedan ser auditables. Incluye la contabilidad multimonedada.

Entre sus módulos se encuentra el de Gestión de Inventarios, el cual permite entre otras funcionalidades:

- Control de los esquemas de depósitos y almacenes.
- Realizar inventarios rotativos.
- Serializar los productos.
- Generar transacciones de costeo.

### **Openbravo**

Openbravo ERP ha sido específicamente diseñado para ayudar a las empresas a mejorar su rendimiento. La cobertura funcional del producto incluye todas las áreas típicas de un sistema de gestión integrado, destacando la solución de contabilidad.

Además, esta aplicación se integra de manera natural con otras áreas como la gestión de relaciones con clientes o CRM<sup>5</sup>, inteligencia de negocio o BI (Business Intelligence) y terminales punto de venta o POS (Point of Sale).

Openbravo ERP utiliza tecnologías modernas, pero sólidas y suficientemente probadas, para cumplir los requerimientos estrictos de rendimiento y escalabilidad de cualquier entorno empresarial:

- Java y Javascript
- SQL y PL/SQL
- XML
- HTML

Los procesos de gestión de almacenes que incorpora Openbravo ERP permiten que las existencias en su organización estén siempre al día y correctamente valoradas. La posibilidad de definir la estructura de almacenes de su organización hasta el mínimo nivel (ubicación) facilita que los stocks estén siempre perfectamente localizados. Adicionalmente, las capacidades para gestionar los lotes de mercancías y la posibilidad de utilizar números de serie aseguran el cumplimiento de los requisitos de trazabilidad impuestos en la mayoría de industrias. (12)

La Gestión de almacenes incluye:

- Almacenes y ubicaciones (multi-almacén).
- Stock por producto en doble unidad (por ejemplo, en kilogramos y cajas).
- Atributos del producto en almacén personalizables (color, talla, descripción de calidad, etc.).
- Lote y número de serie.
- Impresión de etiquetas. Códigos de barras (EAN, UPC, UCC, Code, otras.).
- Gestión de bultos en almacén.
- Control de reposición.
- Trazabilidad configurable por producto.
- Movimiento entre almacenes.
- Gestión automática de salidas de stock (vaciado según existencias, con reglas de prioridad por caducidad, ubicación, etc.).
- **Inventario físico. Planificación de inventarios. Inventario continuado.**

---

<sup>5</sup> CRM: Customer Relationship Management

- Informes de movimientos, seguimiento, stocks, entradas/salidas, caducidades, inventario, ubicaciones, etc. Informes personalizables.
- Integrado con Openbravo POS.

Sincronización y control del stock en la misma tienda

## **SAP**

Software desarrollado en la Ciudad de Mannheim, Alemania, por antiguos empleados de IBM. La corporación se ha desarrollado hasta convertirse en la quinta más grande compañía mundial de software. Cuenta con el módulo Controlling (CO), que permite el control de los gastos generales, costes de producto, cuenta de resultados y centros de beneficio.

El subsistema Inventario de SAP permite reducir los costes de almacenamiento, transporte, cumplimiento de pedidos y manipulación de materiales y, a la vez, mejorar el servicio al cliente. Puede mejorar significativamente la rotación de inventario, optimizar el flujo de mercancías y acortar las rutas en su almacén o centro de distribución. Entre los beneficios adicionales de la gestión de inventarios se encuentran la mejora del flujo de caja, la visibilidad y la toma de decisiones.

Para la gestión de almacenes, puede realizar un seguimiento de la cantidad y el valor de todos sus materiales, realizar un inventario físico y optimizar los recursos del almacén. Los empleados pueden planificar, introducir y documentar movimientos de almacén interno gestionando las entradas y las salidas de mercancías, el almacenamiento, la recogida y el embalaje, los traslados físicos, etc.

Entre sus características principales se encuentran:

- Implementado en .NET y WebSphere.
- SAP también ofrece una nueva plataforma tecnológica denominada SAP NetWeaver, esta plataforma tecnológica convierte a SAP en un programa Web-enabled, lo que significa que estaría totalmente preparado para trabajar con él mediante la web.
- Trabaja sobre el sistema operativo Windows.
- Soporte para bases de datos Oracle .(13)

## Valoración crítica

Desde el punto de vista de la solución estos sistemas serían capaces de resolver el problema del módulo Inventario Físico debido a su alto nivel de configuración y servicios que proveen, pero tienen la desventaja de que algunos de ellos utilizan tecnologías que no son accesibles a Cuba debido a las restricciones impuestas por Estados Unidos. Algunos sistemas como Openbravo están basados en la plataforma J2EE cuya máquina virtual es propiedad de SUN, empresa norteamericana, aunque ha comenzado a liberar el código sigue estando bajo las leyes de su gobierno; además J2EE requiere un consumo de memoria elevado en comparación con PHP/Apache. Como otra desventaja aparece que el diseño de estos ERP ha sido para empresas capitalistas que tienen un modelo de gestión y de procesos muy diferente a las empresas o unidades presupuestadas cubanas donde la economía es centralizada y operan otros mecanismos. Por último, el resto de los softwares propietarios no constituyen una opción viable pues representan gastos muy elevados al país por conceptos de licencias y mantenimiento.

### 1.3. Modelo de desarrollo de software basado en componentes

Por la magnitud del proyecto ERP-Cuba es necesario que cada uno de los equipos de desarrollo posean un modelo estandarizado, así como una definición clara y precisa de las responsabilidades de cada uno de los roles que se ven involucrados en el desarrollo de la solución. (14)

Es por ello que la dirección del proyecto definió su propia metodología basada en un modelo de desarrollo. Esta es orientada a componentes, define claramente las responsabilidades de cada uno de los roles involucrados en el desarrollo de la solución, el flujo de actividades y los artefactos que deben ser generados por cada uno de los roles involucrados en el proceso de desarrollo de software.

A continuación se muestra la tabla donde aparecen las responsabilidades definidas para cada uno de los roles en el proyecto.

Roles	Responsabilidades
Jefe de Línea de Desarrollo	Responsable de garantizar los cronogramas y compromisos de la línea Supervisar el proceso de desarrollo Organiza y controla el trabajo de los miembros de su línea

		Controla los indicadores de eficiencia
Planificador		Mantener actualizado el cronograma Mantener actualizada la plantilla de Capital Humano. Planificar y controlar las tareas de los miembros del equipo, según las prioridades Controlar los horarios de trabajo y distribución de máquinas. Llevar las actas de las reuniones y talleres. Controlar los planes de trabajo Individuales
Arquitecto de Sistema	de	Que se cumplan las políticas y estándares definidos en la Arquitectura. Las decisiones de integración en el proyecto y la Arquitectura del Sistema. Modera el Taller de Diseño.
Arquitecto de Datos	de	Construye y actualiza el Modelo de Datos, además responde por el manejo y recuperación de la información del mismo
Analista Principal		Dirigir y organizar el trabajo del grupo de analistas de la Línea. Elaborar el Mapa de Procesos de la Línea según los estándares. Participar en la definición y construcción de la Arquitectura de Negocio del ERP.
Especialista de Calidad	de	Revisar, controlar las normas y estándares que establece el grupo de aseguramiento de la calidad incluyendo el proceso de desarrollo. Guiar al grupo de auditoría y revisiones Coordinar el proceso de diseño de casos de prueba. Coordinar las pruebas de aceptación o liberación.
Especialista Funcional		Participar en las sesiones de trabajo para identificar, describir y validar los procesos de negocio y los requisitos de software Validar, desde el punto de vista funcional, los procesos de negocio y requisitos de software Elaborar Casos de Prueba según los estándares establecidos para ello
Analista		Participar en las sesiones de trabajo para identificar, describir y validar los procesos de negocio y los requisitos de software Elaborar la Descripción de Procesos de Negocio, Especificación de Requisitos y Casos de Prueba según los estándares establecidos para ello

	Participar en el Taller de Diseño
Desarrollador	Diseña y Construye los componentes de software de la línea

Tabla 1 Responsabilidades por roles definidas en el modelo de desarrollo del proyecto ERP Cuba.

## 1.4. Herramientas y tecnologías utilizadas

### Herramientas de desarrollo de software

Las herramientas para el desarrollo de software (HDS) desempeñan un importante papel en el desarrollo de aplicaciones. Como parte de la ingeniería de software (IS), las HDS han experimentado cambios en los últimos años. Aún ante la existencia de numerosas herramientas, es necesario abordar temas genéricos sobre las HDS, identificar tópicos relevantes y sus relaciones para facilitar su comprensión y análisis.

El soporte que brindan las HDS al proceso de desarrollo proporciona importantes ventajas para el equipo de trabajo de IS. Estas mejoras se sintetizan en:

- Apoyan a las metodologías y métodos, integrando actividades y propiciando visión de continuidad entre fases metodológicas.
- Mejoran la comunicación entre los actores involucrados, facilitándoles compartir su trabajo y desempeñarlo de forma dinámica e iterativa.
- Establecen métodos efectivos para almacenar y utilizar los datos, lo que permite organizar y correlacionar componentes, para accederlos a través de un repositorio.
- Agregan eficiencia al mantenimiento, ya que los programas son construidos sobre las mismas estructuras y estándares, facilitando la adherencia a la disciplina de diseño y facilitan también la conversión automática de programas a versiones más recientes de lenguajes de programación.
- Automatizan porciones del análisis y diseño, tediosas y propensas a error, con influencia sobre la generación de código, las pruebas y el control. Resalta la consideración de que los beneficios potenciales sólo pueden ser alcanzados si las HDS son utilizadas de forma correcta.



## Herramientas CASE

### Visual Paradigm for UML 6.1

Visual Paradigm para UML<sup>6</sup> es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Visual Paradigm es una herramienta CASE concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas entre otras opciones. Constituye una herramienta de probada utilidad para el analista. Dentro de sus características se aprecia que soporta BPMN y UML versión 2.1. Muestra también:

- Diagramas de Procesos de Negocio.
- Modelado colaborativo con CVS y Subversión.
- Generador de informes para generación de documentación.
- Distribución automática de diagramas - Reorganización de las figuras y conectores de los diagramas UML.
- Dibujo de diagramas UML con plantillas (stencils) de MS Visio.
- Editor de figuras.

(15)

### Lenguajes para el modelado

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos o parte de este.

Algunas organizaciones los usan extensivamente en combinación con una metodología de desarrollo de software para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo un equipo de desarrolladores. El uso de un lenguaje de modelado es más sencillo que la auténtica programación, pues existen menos medios para verificar efectivamente el funcionamiento adecuado del modelo.

---

<sup>6</sup> **UML:** (Unified Modeling Language) Lenguaje Unificado de Modelado

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

UML (Unified Modeling Language) o Lenguaje Unificado de Modelado es un lenguaje basado en una notación gráfica la cual permite: especificar, construir, visualizar y documentar los objetos de un sistema programado. UML se quiere convertir en un lenguaje estándar con el que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones, ha sido ampliamente aceptado debido al prestigio de sus creadores. Hay que tener en cuenta que el estándar UML no es un proceso, no es una metodología de desarrollo, sino una notación, un lenguaje (16).

De forma general las principales características son:

- Lenguaje unificado para la modelación de sistemas
- Tecnología orientada a objetos
- El cliente participa en todas las etapas del proyecto
- Corrección de errores viables en todas las etapas
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor
- Facilita a los integrantes de un equipo multidisciplinario participar e intercomunicarse fácilmente, estos integrantes son: los analistas, diseñadores, especialistas de área y desde luego los programadores.

## **Base de Datos**

Un Sistema Gestor de Bases de Datos (SGBD) o DBMS (DataBase Management System) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos.

Algunos ejemplos de SGBD son Oracle, DB2, PostgreSQL, MySQL, MS SQL Server, etc.

Las características de un Sistema Gestor de Base de Datos SGBD son: Abstracción de la información, Independencia, Redundancia mínima, Consistencia, Seguridad, Integridad, Respaldo y recuperación, Control de la concurrencia.

Un SGBD debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.

- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD
- Manipular la base de datos: realizar consultas, actualizarla, generar informes (17).

### **PostgreSQL 8.3**

PostgreSQL es un sistema de gestión de Bases de Datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES de la universidad de Berkeley. Está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo, publicado bajo la licencia BSD.

A continuación se enumeran las principales características de este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos array.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta queries complejos, incluyendo subselects, integridad referencial (Foreign Keys), triggers, vistas (Views), integridad transaccional (ACID), control de versionado concurrente (MVCC). (18)

### **Herramientas de Base de Datos**

#### **pgAdmin III**

pgAdmin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Es capaz de gestionar versiones a partir de PostgreSQL 7.3 ejecutándose en cualquier plataforma, así como versiones comerciales de PostgreSQL entre ellas: Pervasive Postgres, EnterpriseDB, Mammoth Replicator y SRA PowerGres. Incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados, soporte para el motor de replicación Slony-I y mucho más. La conexión al servidor puede hacerse mediante conexión TCP/IP o Unix Domain Sockets (en plataformas \*nix), y puede encriptarse mediante SSL para mayor seguridad (19).

## Herramienta IDE

Un IDE<sup>7</sup> es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. El lenguaje Visual Basic, por ejemplo, puede ser usado dentro de las aplicaciones de Microsoft Office, lo que hace posible escribir sentencias Visual Basic en forma de macros para Microsoft Word.

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto, como es el caso de Smalltalk u Objective-C.

Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación. Este es el caso de Eclipse, al que mediante pluggins se le puede añadir soporte de lenguajes adicionales.

## Zend Studio for Eclipse 6.0

Zend Studio para Eclipse posee las capacidades para crear poderosas herramientas web. Permite a los desarrolladores crear estrategias de integración más eficientes. Al proporcionar potentes capacidades de acción con el lenguaje PHP, mejoras de soporte con JavaScript y profunda integración a Zend Framework, el desarrollo de aplicaciones se realiza en un tiempo récord. Incluye un poderoso editor de código con soporte para todos los formatos web. Posee un fuerte manejo de la arquitectura Cliente/Servidor, excelente depuración, elaboración de perfiles y un código de cobertura. Zend Studio tiene todas las herramientas que un desarrollador necesita para asegurarse de que el código es correcto y empezar a diagnosticar problemas con antelación. Tiene características de depuración avanzadas, incluyendo: condiciones límites, visualización de errores, variables y buffer de salida. Asegura la protección máxima de ubicaciones de proyectos o en Internet con depuradores remotos (20).

---

<sup>7</sup> **IDE:** (Integrated Development Environment) Entorno de desarrollo integrado.

## **Tecnologías Web**

### **Lenguajes de programación del lado del cliente**

#### **HTML**

HTML<sup>8</sup> es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Es un lenguaje que se basa en las marcas para crear los hipertextos. Permite que se creen enlaces entre distintas partes del mismo documento o entre distintas fuentes de información a través de hiperenlaces o hipervínculos, e incluso insertar otros elementos como imágenes y sonidos.

No es mas que un conjunto de etiquetas o comandos, complementados en la mayoría de los casos por extensiones que permiten dar formato a un archivo, con el objetivo básico de crear un documento que pueda ser visualizado en ambiente Internet en forma de Página Web y que esta, además, pueda, por medio de dichas etiquetas, tener la estructura o forma deseada por quien la diseñó. (21).

#### **JavaScript**

JavaScript es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web, puede ser utilizado por profesionales y para quienes se inician en el desarrollo y diseño de sitios web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos.

Entre los diferentes servicios que se encuentran realizados con JavaScript en Internet se encuentran: Correo, Chat, Buscadores de Información.

También podemos encontrar o crear códigos para insertarlos en las páginas como: Reloj, Contadores de visitas, Fechas, Calculadoras, Validadores de formularios, Detectores de navegadores e idiomas (22).

#### **Ajax**

AJAX, acrónimo de Asynchronous JavaScript And XML<sup>9</sup> (JavaScript y XML asíncronos), es una técnica de desarrollo web para crear aplicaciones interactivas. Éstas se ejecutan en el cliente, es decir, en el navegador del usuario, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma, es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

---

<sup>8</sup> **HTML:** (Hyper Text Markup Language) Lenguaje de Marcas de Hipertexto

<sup>9</sup> **XML:** (Extensible Markup Language) Lenguaje de Etiquetado Extensible

AJAX es una combinación de tres tecnologías ya existentes:

XHTML (o HTML) y hojas de estilos en cascada (CSS<sup>10</sup>) para el diseño que acompaña a la información.

DOM<sup>11</sup> accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y Script, para mostrar e interactuar dinámicamente con la información presentada.

El objeto XMLHttpRequest<sup>12</sup> para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios.

XML es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano y JSON.

AJAX no constituye una tecnología en sí, sino que es un término que engloba a un grupo de éstas que trabajan conjuntamente (23).

## Json

JSON<sup>13</sup> es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

JSON está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes se conoce como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

---

<sup>10</sup> **CSS:** (Cascading Style Sheets) Hojas de Estilo en Cascada

<sup>11</sup> **DOM:** (Document Object Model) Modelo en Objetos para la representación de Documentos

<sup>12</sup> **XMLHttpRequest:** Interfaz empleada para realizar peticiones HTTP y HTTPS a servidores WEB.

<sup>13</sup> **JSON:** (JavaScript Object Notation) Notación de Objetos de JavaScript

El formato JSON es el más adecuado para la respuesta del servidor cuando la acción Ajax debe devolver una estructura de datos a la página que realizó la llamada de forma que se pueda procesar con JavaScript. Este mecanismo es útil por ejemplo cuando una sola petición Ajax debe actualizar varios elementos en la página.

JSON se ha convertido en un estándar en el desarrollo de aplicaciones web. Los servicios web proponen la utilización de JSON en vez de XML para permitir la integración de servicios en el navegador del usuario en vez de en el servidor. El formato JSON es seguramente la mejor opción para el intercambio de información entre el servidor y las funciones JavaScript (24).

## **CSS**

CSS es la tecnología desarrollada por el World Wide Web Consortium (W3C) con el fin de separar la estructura de la presentación, permite crear páginas web de una manera más exacta, gracias a las CSS el desarrollador es mucho más dueño de los resultados finales de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como fuentes, colores, márgenes, líneas, altura, anchura, imágenes de fondo, posicionamiento avanzado y muchos otros temas.

Entre los beneficios concretos de CSS encontramos:

- Control de la presentación de muchos documentos desde una única hoja de estilo.
- Control más preciso de la presentación.
- Aplicación de diferentes presentaciones a diferentes tipos de medios (pantalla, impresión, etc.).
- Numerosas técnicas avanzadas y sofisticadas.

(25)

## **Lenguaje de programación del lado del servidor**

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una computadora. Consiste en un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Un lenguaje de programación permite a uno o más programadores especificar de manera precisa: sobre qué datos una computadora debe operar, cómo deben ser estos almacenados, transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural.

Se les clasifica como lenguajes del lado del servidor a los lenguajes de programación en la tecnología cliente servidor que se ejecutan del lado del servidor y de los que los cuales los usuarios solo obtienen el beneficio del procesamiento de la información.

## **PHP**

PHP es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje y significa una ventaja importante.

Algunas de las más importantes capacidades de PHP son: compatibilidad con las bases de datos más comunes, como MySQL, mSQL, Oracle, Informix, y ODBC<sup>14</sup>, por ejemplo. Incluye funciones para el envío de correo electrónico, upload de archivos, crear dinámicamente en el servidor imágenes en formato GIF, incluso animadas y una lista interminable de utilidades adicionales.

No es un lenguaje de marcas y la meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. PHP también soporta el uso de otros servicios que usen protocolos como IMAP<sup>15</sup>, SNMP<sup>16</sup>, POP3<sup>17</sup>, HTTP y derivados. También se pueden abrir sockets de red directos (raw sockets) e interactuar con otros protocolos.

Al ser un lenguaje libre dispone de una gran cantidad de características que lo convierten en la herramienta ideal para la creación de páginas Web dinámicas: Integración con varias bibliotecas externas, permite generar documentos en PDF (documentos de Acrobat Reader), analizar código XML. Perceptiblemente más fácil de mantener y poner al día que el código desarrollado en otros lenguajes (26).

---

<sup>14</sup> **ODBC:** (Open Database Connectivity) Estándar de acceso a Bases de datos.

<sup>15</sup> **IMAP:** (Internet Message Access Protocol) Protocolo de acceso a mensajes almacenados en un servidor.

<sup>16</sup> **SNMP:** (Simple Network Management Protocol) Protocolo de la capa de aplicación que facilita el intercambio de información de administración.

<sup>17</sup> **POP3:** (Post Office Protocol) Protocolo 3 de Correo, está diseñado para recibir correos, no para enviarlo.



## Arquitectura

Una Arquitectura de Software, también denominada Arquitectura lógica, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información. Esta se selecciona y diseña con base en objetivos y restricciones. Los objetivos son aquellos prefijados para el sistema de información, pero no solamente los de tipo funcional, también otros objetivos como la mantenibilidad, auditabilidad, flexibilidad e interacción con otros sistemas de información. Las restricciones son aquellas limitaciones derivadas de las tecnologías disponibles para implementar sistemas de información. Unas arquitecturas son más recomendables de implementar con ciertas tecnologías mientras que otras tecnologías no son aptas para determinadas arquitecturas.

## Modelo-Vista-Controlador (MVC)

La arquitectura MVC (Model/View/Controller) fue introducida como parte de la versión Smalltalk-80 del lenguaje de programación Smalltalk. Fue diseñada para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales son que el Modelo, las Vistas y los Controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el Modelo se refleje automáticamente en cada una de las Vistas. El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos.

La Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario.

El Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo.

Este modelo de arquitectura presenta varias **ventajas**, las principales son:

- Hay una clara separación entre los componentes de un programa; lo cual nos permite implementarlos por separado.

- La conexión entre el Modelo y sus Vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

### **Desventaja:**

Costo de actualizaciones frecuentes: Si el modelo experimenta cambios frecuentes, por ejemplo, podrían desbordar las vistas con una lluvia de requerimientos de actualización.

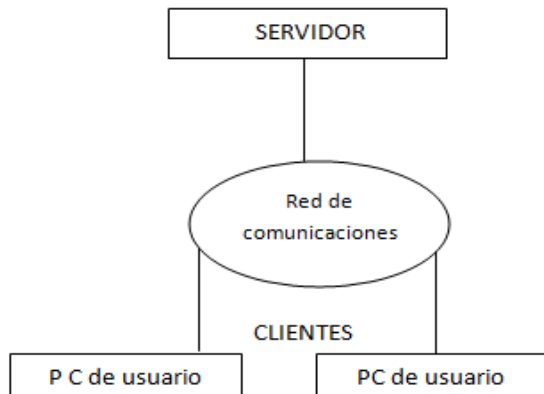
### **Arquitectura Cliente-Servidor**

Esta arquitectura se divide en dos partes claramente diferenciadas, la primera es la parte del servidor y la segunda la de un conjunto de clientes.

Normalmente el servidor es una máquina bastante potente que actúa de depósito de datos y funciona como un sistema gestor de base de datos (SGBD).

Por otro lado los clientes suelen ser estaciones de trabajo que solicitan varios servicios al servidor. Ambas partes deben estar conectadas entre sí mediante una red.

Una representación gráfica de este tipo de arquitectura sería la siguiente.



Este tipo de arquitectura es la más utilizada en la actualidad, debido a que es la más avanzada y la que mejor ha evolucionado en estos últimos años.

Podemos decir que esta arquitectura necesita tres tipos de software para su correcto funcionamiento:

- Software de gestión de datos: Este software se encarga de la manipulación y gestión de los datos almacenados y requeridos por las diferentes aplicaciones. Normalmente este software se aloja en el servidor.
- Software de desarrollo: este tipo de software se aloja en los clientes y solo en aquellos que se dedique al desarrollo de aplicaciones.

- Software de interacción con los usuarios: También reside en los clientes y es la aplicación gráfica de usuario para la manipulación de datos, siempre claro a nivel usuario (consultas principalmente).

## Ventajas

- **recursos centralizados:** debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada se utilizaría para evitar problemas provocados por datos contradictorios y redundantes.
- **seguridad mejorada:** ya que la cantidad de puntos de entrada que permite el acceso a los datos no es importante.
- **administración al nivel del servidor:** ya que los clientes no juegan un papel importante en este modelo, requieren menos administración.
- **red escalable:** gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones.

## Desventajas

- **costo elevado:** debido a la complejidad técnica del servidor.
- **un eslabón débil:** el servidor es el único eslabón débil en la red de cliente/servidor, debido a que toda la red está construida en torno a él. Afortunadamente, el servidor es altamente tolerante a los fallos.

## Servidor Web Apache

Apache es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

Entre sus principales características se encuentran:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita de código fuente abierto. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si queremos ver que es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto, sin ninguna puerta trasera).
- Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor. Actualmente existen muchos módulos para Apache que son

adaptables a este, y están ahí para que los instalemos cuando los necesitemos. Otra cosa importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada.

- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de logs<sup>18</sup>. Este permite la creación de ficheros de log a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en tu servidor.

(27)

## Frameworks

Un framework es una estructura de software compuesta por componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta, puede incluir soporte de programas, bibliotecas y un lenguaje de scripting entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Los frameworks son diseñados con el intento de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional.

Un framework permite separar en capas la aplicación:

- La lógica de presentación que administra las interacciones entre el usuario y el software.
- La lógica de dominio o de negocio, que manipula los modelos de datos de acuerdo a los comandos recibidos desde la presentación.
- La lógica de datos que permite el acceso a un agente de almacenamiento persistente u otros.

---

<sup>18</sup> **Logs:** Registro de errores.

## Zend Frameworks

Es un framework para desarrollo de aplicaciones Web y servicios Web con PHP. Brinda soluciones para construir sitios web modernos, robustos y seguros. Además es código abierto y trabaja con PHP 5.

Presenta entre otras las siguientes características:

- Simplifica la gestión de archivos de configuración.
- Proporciona los componentes que forma la infraestructura del patrón Modelo-Vista-Controlador.
- Proporciona una capa de acceso a base de datos, construida sobre PDO<sup>19</sup> pero ampliándola con diferentes características.
- Cuenta con módulos para manejar archivos PDF, canales RSS, Web Services (Amazon, Flickr, Yahoo)
- Proporciona mecanismos de filtrado y validación de entradas de datos.
- Clientes para servicios web, incluidos Google Data APIs y Strikelron

(28)

## Zend\_Ext Framework.

Es un framework open Source, que esta diseñado para php 5 y buenas capacidades de ampliación. Es elaborado a partir de Zend Framework cumpliendo con todas sus características. Este trae un motor de reglas para las validaciones en el servidor, se incluyó el IoC para la comunicación entre los módulos o componentes. Se le incorporó la integración con el ORM (Object Relational Mapper) Doctrine Framework para trabajo en la capa de abstracción a base de datos y el ExtJs Framework para el desarrollo de las vistas.

## Doctrine Framework

Es un potente y completo sistema ORM para PHP 5.2+ que incorpora una DBL (capa de abstracción a base de datos). Uno de sus rasgos importantes es la habilidad de escribir opcionalmente las preguntas de la base de datos orientado a objeto. Esto les proporciona una alternativa poderosa a diseñadores de SQL que mantiene un máximo de flexibilidad sin requerir la duplicación del código innecesario. También permite exportar una base de datos existente a sus clases correspondientes y también convierte clases (convenientemente creadas siguiendo las pautas del ORM) a tablas de una base de datos (29).

---

<sup>19</sup> PDO: (PHP Data Objects) Capa de abstracción de acceso a datos para PHP

## ExtJs Framework

Es una librería construida con JavaScript que proporciona una interfaz a las famosas librerías de Yahoo, jQuery, y Prototype + Scriptaculous, su potencia radica en la rica colección de componentes para el diseño de GUI's del lado del cliente haciendo uso extensivo de Ajax.

ExtJS es neutral al lenguaje que se use en el servidor. Siempre que el resultado se envíe a la página en el formato adecuado, ExtJS no se preocupará de lo que pase en el servidor. Hay docenas de widgets a escoger en ExtJS, incluyendo composiciones automáticas de páginas, pestañas, menús, barras de herramientas, diálogos, vistas en árbol. Proporciona un selector de nodos DOM extremadamente poderoso llamado DomQuery (puede usarse como una librería independiente, pero en el contexto de ExtJS se usará para seleccionar elementos para poder interactuar con ellos a través de la interfaz Element, contiene mucho de los métodos y propiedades de DOM que se necesitará proporcionando una interfaz conveniente, unificada y multinavegador).

Entre los componentes que esta librería ofrece encontramos cuadros de diálogo, menús, tablas editables, layouts, paneles, pestañas y todo lo necesario para construir atractivos desarrollos al estilo de Web 2.0.

### Ventajas:

- La orientación a objetos intensa hará modular todos los scripts.
- El diseño está completamente separado de la funcionalidad.
- Funciones comunes como validación, comboboxes editables, ventanas arrastrables (con minimizar y maximizar), son muy fáciles de implementar.
- Buena y amplia documentación, así como también su comunidad.

### Desventajas:

- Crear un sistema serio con esta herramienta requiere un previo uso prolongado, ya que se dificulta el manejo de los nuevos objetos en su extensa y bien documentada API. El tiempo de aprendizaje puede llegar a compararse con aprender a programar en un lenguaje nuevo.
- Al estar todo el sitio en JS, no podrá ser accesible para los buscadores, limitando su uso a sistemas y no sitios web.

(30)

## UCID Framework

Es el Framework encargado del trabajo con la vistas. Abarca la integración de ExtJs Framework con el sistema incluyendo el integrador de interfaz, el generador de interfaz dinámica y la impresión de documentos. Integra la iconografía, los diferentes temas de escritorio de la aplicación, el multilinguaje.

## Navegador

Un navegador web o de Internet, en inglés un "browser", es un programa que permite visualizar la información que contiene una página web (que es un página de los sitios en la red, ya sea esta la Internet o en una red local). Además son usados para visualizar archivos que utilicen el mismo formato de los documentos en la Internet.

## Mozilla Firefox 3.0

Es el nuevo e innovador navegador open source acerca del cual todo el mundo habla. Firefox ha sido creado por el proyecto Mozilla, un esfuerzo open source sin ánimo de lucro que incluye a miles de voluntarios alrededor del mundo. La misión del proyecto Mozilla es preservar la elección y la innovación en Internet. El apoyo organizativo del proyecto Mozilla es proporcionado por Mozilla Foundation (en los Estados Unidos de América), Mozilla Europe y Mozilla Japón.

Por nombrar algunas de las posibilidades que ofrece Firefox y que no ofrece IE<sup>20</sup>, están:

Es Software libre.

- En Firefox no existen la cantidad de bugs que posee el catastrófico IE, inmediatamente se encuentra un bug en el producto es notificado al Proyecto Mozilla para que sea reparado el problema.
- Navegación por tabs: Esta es una de las principales características que tiene Firefox.
- También existen excelentes extensiones de fácil instalación, estos mejoran la usabilidad y el aspecto del navegador, cosa que no se logra en IE el cual siempre permanece con los mismos colores.

(31)

---

<sup>20</sup> IE: Internet Explorer

## Control de Versiones

Un sistema de control de versiones (o sistema de control de revisiones) es una combinación de tecnologías y practicas para seguir y controlar los cambios realizados en los ficheros del proyecto, en particular en el código fuente, en la documentación y en las páginas web.

Un sistema de control de versiones debe proporcionar:

- Mecanismo de almacenaje de los elementos que deba gestionar (ej. archivos de texto, imágenes, documentación...).
- Posibilidad de realizar cambios sobre los elementos almacenados (ej. modificaciones parciales, añadir, borrar, renombrar o mover elementos).
- Registro histórico de las acciones realizadas con cada elemento o conjunto de elementos (normalmente pudiendo volver o extraer un estado anterior del producto).

### Subversion 1.4.5

Subversión es un controlador de versiones empleado en la administración de archivos utilizados en el desarrollo de software o contenido. CVS<sup>21</sup> considerado su antecesor es uno de los controladores de versiones más utilizados en proyectos de software libre, sin embargo, a pesar de su amplio uso, el mismo diseño de CVS resultó ineficiente para diversos grupos de usuarios, y ante estas inconformidades se dio inicio al proyecto que hoy es conocido como Subversión, el mismo que ha empezado a socavar el dominio de CVS.

### Ventajas

- Se sigue la historia de los archivos y directorios a través de copias y renombrados.
- Las modificaciones (incluyendo cambios a varios archivos) son atómicas.
- La creación de ramas y etiquetas es una operación más eficiente; Tiene costo de complejidad constante ( $O(1)$ ) y no lineal ( $O(n)$ ) como en CVS.
- Se envían sólo las diferencias en ambas direcciones (en CVS siempre se envían al servidor archivos completos).
- Maneja eficientemente archivos binarios (a diferencia de CVS que los trata internamente como si fueran de texto).

---

<sup>21</sup> **CVS:** Concurrent Versions System



- Cuando se usa integrado a Apache permite utilizar todas las opciones que este servidor provee a la hora de autenticar archivos (**SQL**, **LDAP**<sup>22</sup>, **PAM**<sup>23</sup>).
- Existen varias interfaces a Subversión, ya sea programas individuales como interfaces que lo integran en entornos de desarrollo.
- TortoiseSVN. Provee integración con el explorador de Windows. Es la interfaz más popular en este sistema operativo.
- Subclipse. "Plugin" que integra Subversión al entorno Eclipse.
- Subversive. "Plugin" alternativo para Eclipse.
- ViewVC. Interfaz web, que también trabaja delante de CVS.

Para Mac, pueden emplearse los interfaces SvnX, RapidSVN y Zigversion (32).

### **Conclusiones parciales.**

En el presente capítulo se realizó el estudio de las herramientas, tecnologías y la metodología propuesta por la dirección de arquitectura del proyecto, permitiendo sentar las bases para el desarrollo. Además se han tratado de forma general y resumida los sistemas que abarcan la gestión de inventarios, determinando las ventajas, desventajas y factibilidad de utilización en las entidades cubanas.

---

<sup>22</sup> **LDAP:** (Lightweight Directory Access Protocol) Protocolo Ligero de Acceso a Directorios.

<sup>23</sup> **PAM:** Pluggable Authentication Modules.

## *Capítulo 2: Descripción de la Solución Propuesta*

### **Introducción**

En este capítulo se abordará fundamentalmente los procesos que serán objeto de automatización. Se hará mención de los documentos procesados para la manipulación de la información. Se describirá la propuesta de la solución para dar una visión general de cómo funcionará el software permitiendo un modelado de los procesos de negocio. Además de analizar los requisitos funcionales del software.

### **2.1. Objeto de estudio**

El objeto de estudio girará entorno a: Los procesos para el Control de Inventario.

### **2.2. Problema**

La no existencia de una herramienta que gestione la información referente a los procesos de Inventario del subsistema del Subsistema Activo Fijo Tangible.

### **2.3. Situación problemática**

En el país hay varios software que cuentan con el módulo de inventarios pero no resultan soluciones factibles para las entidades cubanas debido a que:

Fueron desarrollados sobre plataformas de software propietario, lo que implica incrementos de gastos en licencias de uso y mantenimiento del software. Además, las soluciones nacionales constituyen aplicaciones de escritorio lo que trae como desventaja que el usuario deba instalar la aplicación en cada estación de trabajo. Son productos que se caracterizan por abordar solamente partes del problema de la gestión de la empresa o la unidad presupuestada, no soportan mecanismos estándares de integración con otras aplicaciones donde casi ninguno bajo conceptos de informática multicapa y distribuida en la red. Por otra parte los sistemas internacionales desde el punto de vista de la solución serían capaces de resolver el problema del módulo Inventario Físico debido a su alto nivel de configuración y servicios que proveen, pero tienen la desventaja de que algunos de ellos utilizan tecnologías que no son accesibles a Cuba debido a las restricciones impuestas por Estados Unidos. Algunos sistemas como OpenBravo están

basado en la plataforma J2EE<sup>24</sup> cuya máquina virtual es propiedad de SUN, empresa norteamericana, aunque ha comenzado a liberar el código sigue estando bajo las leyes de su gobierno; además J2EE requiere un consumo de memoria elevado en comparación con PHP/Apache. Como otra desventaja aparece que el diseño de estos ERP ha sido para empresas capitalistas que tienen un modelo de gestión y de procesos muy diferente a las empresas o unidades presupuestadas cubanas donde la economía es centralizada y operan otros mecanismos. Por último, el resto de los softwares propietarios no constituyen una opción viable pues representan gastos muy elevados al país por conceptos de licencias y mantenimiento.

#### **2.4. Objeto de automatización**

Para realizar el proceso inventario de AFT se selecciona la hoja de inventario donde desea registrar el inventario, se verifica que se pueda realizar el inventario, se registra el resultado de la verificación realizada, si el activo está en el lugar que le corresponde, el usuario debe marcar el activo como verificado y se registra el inventario.

#### **2.5. Información que se maneja.**

Inventarios de Activos Fijos Tangibles.

#### **2.6. Propuesta de sistema**

Al iniciar la sesión el sistema verificará los privilegios que posee el usuario logueado, permitiendo acceder solo a las entidades y Subsistemas que tenga acceso, así como a las funcionalidades y las distintas acciones definidas según el rol que desempeña. Para poder llevar a cabo los procesos de Gestión de Inventarios de AFT presentes en el módulo de Inventario de AFT se muestran dos funcionalidades:

- Gestionar documento de Inventario
- Gestionar AFT a documentos.

Si se selecciona la primera funcionalidad el sistema permitirá realizar las siguientes acciones:

- Permite adicionar un documento de Inventario
- Modificar el documento de Inventario

---

<sup>24</sup> J2EE: Java 2 Platform, Enterprise Edition.

Donde se pueden realizar varias acciones como:

- Modificar el campo Aprobado por, el cual permitirá introducir el nuevo nombre.
- Modificar el campo fecha de aprobación.
- Modificar el campo Observaciones.

- Eliminar el documento de Inventario.
- Confirmar el documento de Inventario.
- Imprimir modelo de inventario general de Activo Fijo Tangible.

Si se selecciona la segunda opción permite:

- Adicionar AFT nuevos a documentos.
- Adicionar AFT existentes a documentos.
- Eliminar AFT del documento.
- Buscar AFT.
- Realizar búsqueda avanzada de AFT.

## **2.7. Especificación de los requisitos de software**

### **Requerimientos Funcionales**

Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir. Estos describen los servicios que se espera que el sistema cumpla para satisfacer las necesidades del usuario y define el comportamiento del sistema.

Se definen los siguientes requisitos después de haber hecho un análisis profundo del negocio.

- RF 1: Gestionar documento de Inventario.
  - RF 1.1: Adicionar documento de Inventario
  - RF 1.2: Modificar documento de Inventario
  - RF 1.3: Eliminar documento de Inventario
  - RF 1.4: Confirmar documento de Inventario
  - RF 1.5: Imprimir modelo de inventario general de Activo Fijo Tangible

- RF 2: Gestionar AFT a documentos.
  - RF 2.1: Adicionar AFT nuevos a documentos.
  - RF 2.2: Adicionar AFT existentes a documentos.
  - RF 2.3: Eliminar AFT del documento.
  - RF 2.4: Buscar AFT.
  - RF 2.5: Realizar búsqueda avanzada de AFT.

### **Requerimientos no funcionales**

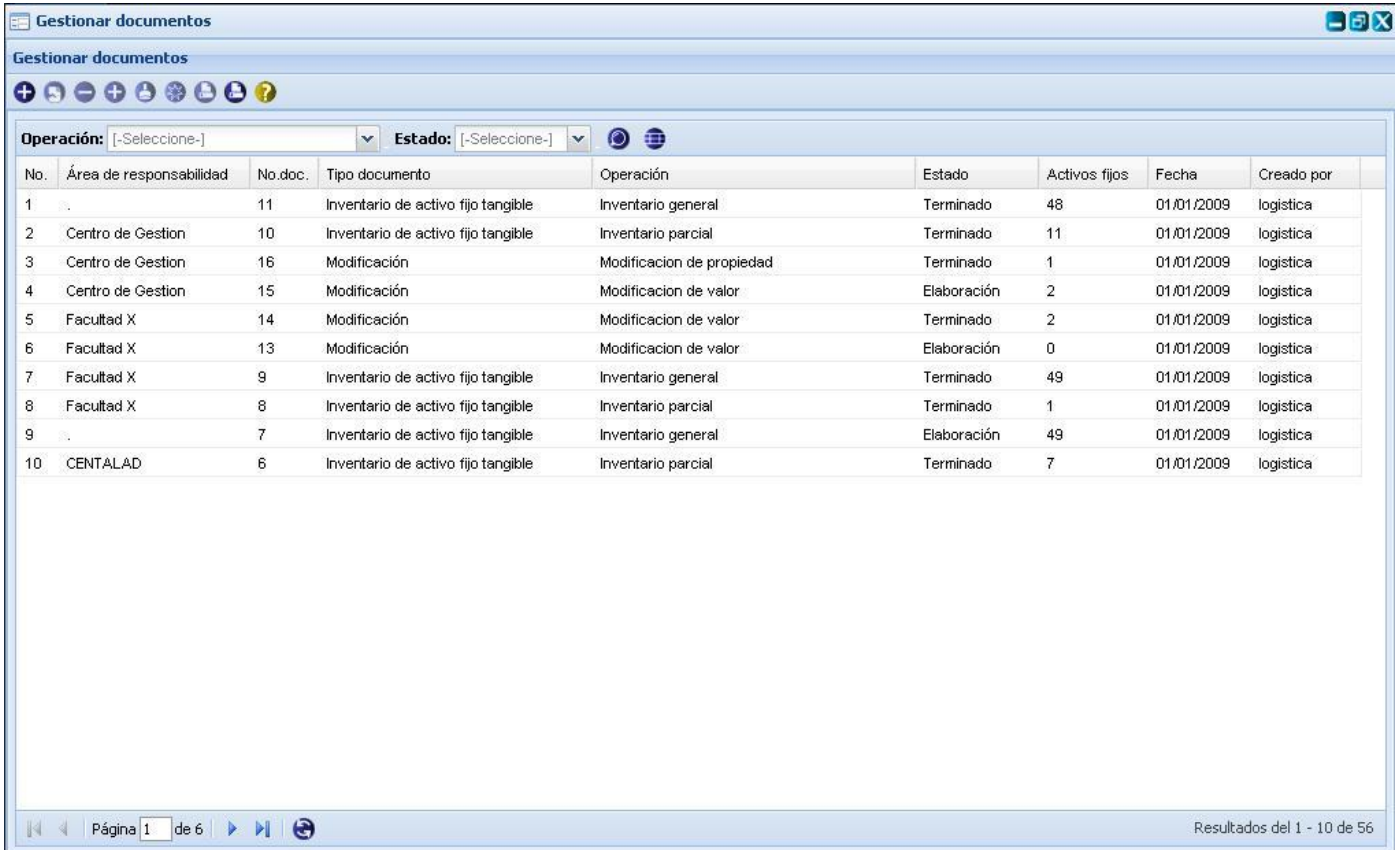
Son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

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

- Requerimientos de Software
- Requerimientos de Hardware
- Restricciones en el diseño y la implementación
- Requerimientos de apariencia o interfaz externa
- Requerimientos de Usabilidad
- Requerimientos de Soporte
- Requerimientos Legales
- Requerimientos de confiabilidad
- Requerimientos de interfaz Interna
- Requerimientos de Seguridad

## 2.8. Principales Interfaces

### ➤ Gestionar documentos.



The screenshot displays a web application window titled "Gestionar documentos". At the top, there are navigation icons and a search bar. Below the search bar, there are two dropdown menus: "Operación:" and "Estado:", both currently set to "[-Seleccione-]". The main content area contains a table with the following data:

No.	Área de responsabilidad	No.doc.	Tipo documento	Operación	Estado	Activos fijos	Fecha	Creado por
1	.	11	Inventario de activo fijo tangible	Inventario general	Terminado	48	01/01/2009	logistica
2	Centro de Gestion	10	Inventario de activo fijo tangible	Inventario parcial	Terminado	11	01/01/2009	logistica
3	Centro de Gestion	16	Modificación	Modificacion de propiedad	Terminado	1	01/01/2009	logistica
4	Centro de Gestion	15	Modificación	Modificacion de valor	Elaboración	2	01/01/2009	logistica
5	Facultad X	14	Modificación	Modificacion de valor	Terminado	2	01/01/2009	logistica
6	Facultad X	13	Modificación	Modificacion de valor	Elaboración	0	01/01/2009	logistica
7	Facultad X	9	Inventario de activo fijo tangible	Inventario general	Terminado	49	01/01/2009	logistica
8	Facultad X	8	Inventario de activo fijo tangible	Inventario parcial	Terminado	1	01/01/2009	logistica
9	.	7	Inventario de activo fijo tangible	Inventario general	Elaboración	49	01/01/2009	logistica
10	CENTALAD	6	Inventario de activo fijo tangible	Inventario parcial	Terminado	7	01/01/2009	logistica

At the bottom of the interface, there is a pagination control showing "Página 1 de 6" and a status bar indicating "Resultados del 1 - 10 de 56".

➤ Adicionar un nuevo documento.

**Nuevo documento**

**Datos del documento:**

**Entidad:** UCI AFT  
**Número:** 0  
**Fecha:** 01/01/2009  
**Estado:** Elaboración

**Datos generales del documento**    Movimiento de activos fijos

**Tipo de documento:** [-Seleccione-]  
**Operaciones:** [-Seleccione-]  
**Aprobado por:**

**Área de responsabilidad:** [-Seleccione-]  
**Centro de Costo:** [-Seleccione-]  
**Fecha de aprobación:**

**Documento de origen:**  
**Nro documento:**     **Fecha:**   
**Tipo de documento:** [-Seleccione-]

**Observaciones:**

## ➤ Gestionar AFT al Documento.

Gestion de inventario general

Área de responsabilidad:

Datos del documento:

Entidad: UCI AFT  
Número: 11  
Fecha: 01/01/2009  
Estado: Terminado

Activos Fijos por Área de Responsabilidad

Descripción:		Número Inventario:								
Número	Código	Descripción	Número Inventario	Área de responsabilidad	Valor contable	Vida útil	Tasa	Depreciación Acumulada	Estado	
1	006101	Samsung ML 1640	2222222	Centro de Gestion	9.00	7	14.29	0.00		
2	4252010002	BMP	10001	Centro de Gestion	0.00	10	0	0.00		
3	4252010002	BMP	10003	Centro de Gestion	0.00	10	0	0.00		
4	005202	HP Laserjet P1005	55222	Centro de Gestion	0.00	7	0	0.00		
5	001102	M3800S-BN	65456	Centro de Gestion	0.00	10	0	0.00		
6	001102	M3800S-BN	34564	Centro de Gestion	0.00	10	0	0.00		
7	005202	HP Laserjet P1005	55224	Centro de Gestion	0.00	7	0	0.00		
8	005202	HP Laserjet P1005	55221	Centro de Gestion	0.00	7	0	0.00		
9	005101	HP Color Laserjet CP1215	55556	Centro de Gestion	5.00	7	14.29	0.00		
10	005101	HP Color Laserjet CP1215	55557	Centro de Gestion	5.00	7	14.29	0.00		

Página 1 de 5

Resultados del 1 - 10 de 48

Página 1 de 6

Cancelar

## Conclusiones Parciales

Con la realización de este capítulo se han tratado los procesos fundamentales que serán objeto de automatización con el objetivo de que se comprenda mejor el negocio. Se ha hecho mención de los documentos procesados para la manipulación de la información para tener una idea de cuales son los documentos con los que se trataran. Se han listado los requisitos funcionales del software para que sepa cuales son las funcionalidades con las que contará la aplicación y se ha descrito la propuesta de la solución para dar una visión general de cómo funcionará el software.



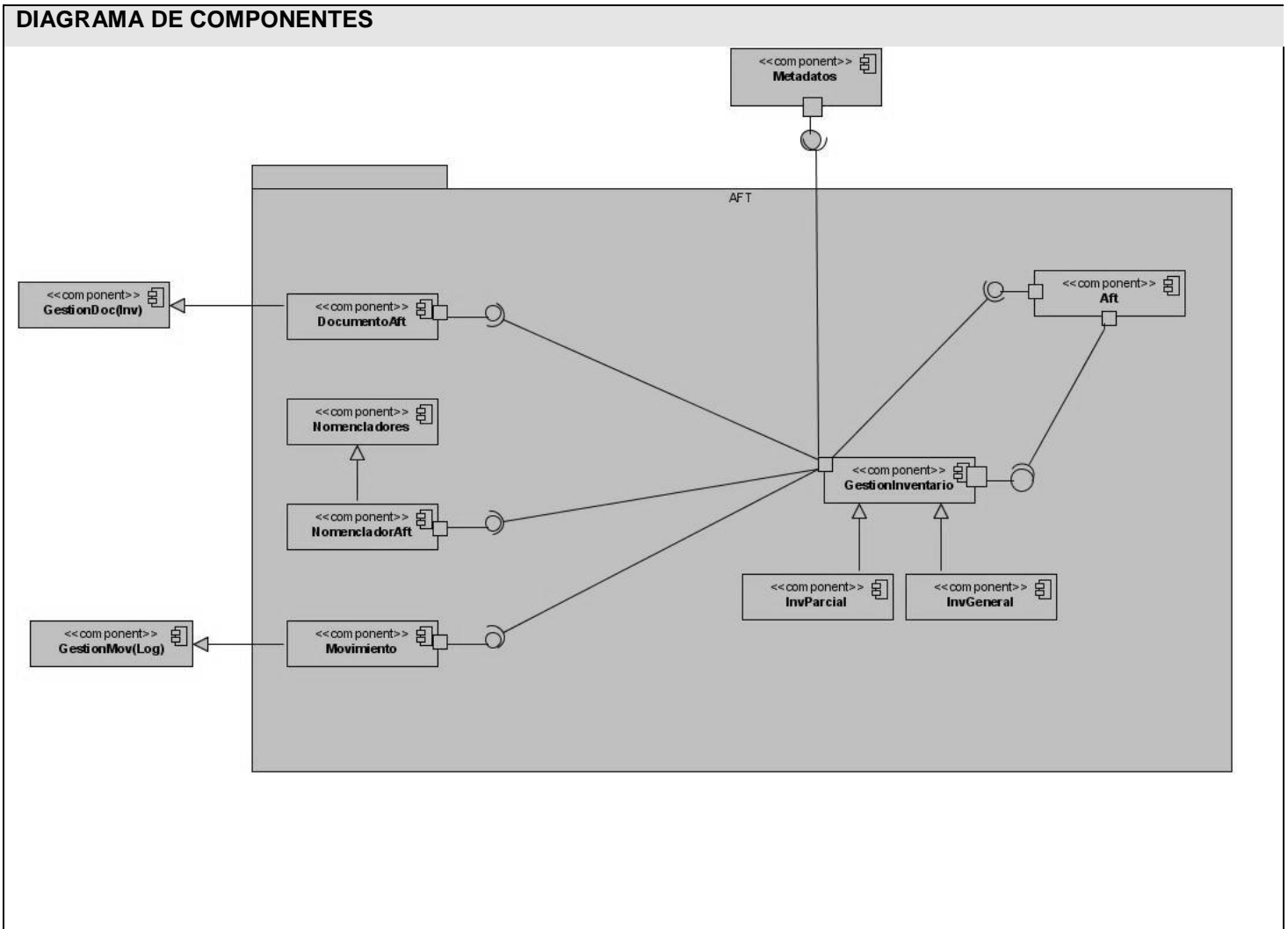
## *Capítulo 3: Implementación y prueba.*

### **Introducción**

En este capítulo se presenta el diagrama de componentes, se realizan las matrices de integración de componentes donde se muestran los servicios que se consumen de otros componentes. Además como el desarrollo de un software es algo complejo y son innumerables las posibilidades de errores, debe ir acompañado de alguna actividad que garantice que el producto es aceptable; las pruebas son un elemento crítico para la garantía de la calidad del software. Es por eso que se utilizaron técnicas como las pruebas unitarias que no son más que una forma de verificar el correcto funcionamiento del código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

### 3.1. Implementación

Este diagrama representa la forma en que un sistema de software se divide en componentes mostrando así las dependencias entre dichos componentes. Los componentes físicos incluyen archivos, cabeceras, librerías compartidas, módulos, ejecutables y paquetes.



### 3.1.1. Matriz de Integración de Componentes Interna.

Las siguientes matrices resumen brevemente los servicios que consume el módulo de Inventario de otros componentes tanto externos al subsistema AFT como internos.

	Componentes Internos			
Componentes Internos	AFT	Movimiento	Nomencladores	Documentos
<b>Componente Inventario de AFT.</b>	mostrarCmpAFT(), mostrarCmpNomAFT(), obtenerTodosAFTDeEntidad(), obtenerCodigoAFTDadold().	crearMovimientoAft(), adicionarSobranteMovimientoAft(), existeMovimientoAft(), adicionarMovimientoAft (), buscarMovimientosPorDocumentoAft (), buscarMovimientoAft(), modificarMovimientoAft(), cantidadMovimientosPorDocumento (),eliminarMovimientoAft	ObtenerGrupo()	DatosEncabezadoInventario (),

### 3.1.2. Matriz de Integración de Componentes Externa.

	Componentes externos
<b>Componentes Internos</b>	Metadatos
<b>Componente Inventario de AFT.</b>	DameEstructurasInternas(), DameAreasPorId(), DameHijosInterna()

### 3.2. Prueba

En el siguiente epígrafe se presentarán Los DCP (Diseños de Casos de pruebas) que fueron utilizados para aplicar las pruebas unitarias que no son más que una forma de verificar el correcto funcionamiento del código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

#### Condiciones de ejecución

- Se debe identificar y autenticar ante el sistema y además debe tener los permisos para ejecutar esta acción.
- Se debe seleccionar el subsistema de Logística.
- Se debe seleccionar la opción **Activo Fijo Tangible/Gestión de documentos/Gestionar documentos**

#### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Adicionar documento.	Se adiciona un documento al sistema.	EP 1.1: Adicionar un documento introduciendo los datos válidos.	<ul style="list-style-type: none"><li>– Se selecciona adicionar nuevo documento.</li><li>– El sistema debe cargar automáticamente los atributos del nuevo documento. (Entidad, número , Fecha, Estado)</li><li>– El usuario selecciona el tipo de documento.</li><li>– El usuario introduce los datos.</li><li>– El sistema muestra un mensaje: “La información fue introducida satisfactoriamente.”.</li><li>– Se presiona el botón</li></ul>

			<b>Aceptar.</b>
		EP 1.2: Adicionar un documento introduciendo los datos con información errónea.	<ul style="list-style-type: none"> <li>- Se selecciona adicionar nuevo documento.</li> <li>- El sistema debe cargar automáticamente los atributos del nuevo documento. (Entidad, número , Fecha, Estado)</li> <li>- El sistema muestra un mensaje de error: “La información es errónea.”.</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		EP 1.3: Adicionar un documento cuando la información es incompleta.	<ul style="list-style-type: none"> <li>- Se selecciona adicionar nuevo documento.</li> <li>- El sistema debe cargar automáticamente los atributos del nuevo documento. (Entidad, número , Fecha, Estado)</li> <li>- El usuario selecciona el tipo de documento.</li> <li>- El usuario introduce los datos.</li> <li>- El sistema muestra un mensaje de error: “La información es incompleta.”.</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>

		<p>EP 1.4: Adicionar un documento cuando se insertan AFT en el momento de crear el documento.</p>	<ul style="list-style-type: none"> <li>- Se selecciona adicionar nuevo documento.</li> <li>- El sistema debe cargar automáticamente los atributos del nuevo documento. (Entidad, número , Fecha, Estado)</li> <li>- Se insertan los AFT en el documento.</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		<p>EP 1.5: Cancelar.</p>	<ul style="list-style-type: none"> <li>- Se selecciona o no Adicionar documento.</li> <li>- Se presiona el botón <b>Cancelar.</b></li> </ul>

### Juegos de datos a probar

Id del escenario	Escenario	Observación de la operación	Datos del documento de origen	Nombre del que lo aprueba	Cargo del que lo aprueba	Nombre del que lo crea	Respuesta del sistema	Resultado de la prueba

EP 1.1	Adicionar un documento o introduciendo los datos válidos.	V(faltan datos del documento)	V(250)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema registra la adición del documento y muestra una ventana de confirmación: "Los datos han sido guardado satisfactoriamente."	
EP 1.2	Adicionar un documento o introduciendo los datos con información errónea.	V(erf6?)	V(250)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
		V(faltan datos del documento)	V(kek4\$)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	

		V(faltan datos del documento)	V(256)	I(265?)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
		V(faltan datos del documento)	V(256)	V(José Carlos)	I(258\$)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
		V(faltan datos del documento)	V(letras)	V(José Carlos)	V(Jefe de Logística)	I(258\$)	El sistema muestra un mensaje de error: "La información es incompleta."	
EP 1.3	Adicionar un documento cuando la información es incompleta	I(Vacio)	V(250)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	



a.	V(faltan datos del documento)	I(Vacio)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
	V(faltan datos del documento)	V(250)	I(Vacio)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
	V(faltan datos del documento)	V(250)	V(José Carlos)	I(Vacio)	V(Arismaida)	El sistema muestra un mensaje de error: "La información es incompleta."	
	V(faltan datos del documento)	V(250)	V(José Carlos)	V(Jefe de Logística)	I(Vacio)	El sistema muestra un mensaje de error: "La información es incompleta."	

EP 1.4	Adicionar un documento o cuando se insertan AFT en el momento de crear el documento.	V(faltan datos del documento)	V(250)	V(José Carlos)	V(Jefe de Logística)	V(Arismaida)	El sistema muestra un mensaje: “Los datos han sido guardado satisfactoriamente.”.	
EP 1.5	Cancelar.	N/A	N/A	N/A	N/A	N/A	Se cancela la operación y se cierra la ventana.	

### Condiciones de ejecución

- Se debe identificar y autenticar ante el sistema y además debe tener los permisos para ejecutar esta acción.
- Se debe seleccionar el subsistema de Logística.
- Se debe seleccionar la opción **Activo Fijo Tangible/Gestión de documentos/Gestionar documentos**
- Se debe haber creado algún documento.

### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Modificar	Se modifica el	EP 1.1: Modificar un	– Se selecciona modificar

documento.	documento con los datos:	documento introduciendo los datos válidos.	<p>documento.</p> <ul style="list-style-type: none"> <li>- El sistema muestra y permite editar los datos del documento: número documento, fecha operación, estado, entidad, tipo, nombre aprueba, cargo aprueba, observaciones y creado por.</li> <li>- El sistema registrar los datos</li> <li>- El sistema muestra un mensaje: "La información fue introducida satisfactoriamente."</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		EP 1.2: Modificar un documento introduciendo los datos con información errónea.	<ul style="list-style-type: none"> <li>- Se selecciona modificar documento.</li> <li>- El sistema muestra y permite editar los datos del documento: número documento, fecha operación, estado, entidad, tipo, nombre aprueba, cargo aprueba, observaciones y creado por.</li> <li>- El sistema registrar los</li> </ul>

			<p>datos</p> <ul style="list-style-type: none"> <li>- El sistema muestra un mensaje: "La información introducida es incorrecta."</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		<p>EP 1.3: Adicionar un documento cuando la información es incompleta.</p>	<ul style="list-style-type: none"> <li>- Se selecciona modificar documento.</li> <li>- El sistema muestra y permite editar los datos del documento: número documento, fecha operación, estado, entidad, tipo, nombre aprueba, cargo aprueba, observaciones y creado por.</li> <li>- El sistema registrar los datos</li> <li>- El sistema muestra un mensaje: "La información introducida es incompleta."</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		<p>EP 1.4: Cancelar.</p>	<ul style="list-style-type: none"> <li>- Se selecciona o no Adicionar documento.</li> <li>- Se presiona el botón</li> </ul>

			<b>Cancelar.</b>
--	--	--	------------------

### Juegos de datos a probar

Id del escenario	Escenario	Número documento	Fecha operación	Estado	Entidad	Tipo	Nombre prueba	Cargo prueba	Observaciones	Creado por	Respuesta del sistema	Resultado de la prueba
EP 1.1	Modificar un documento introduciendo los datos	V(15)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información fue introducida"	
EP 1.2	Modificar un documento introduciendo los datos con	I(erf6?)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información introducida es incorrecta"	

información errónea	V(16)	I(8/12)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información
	V(16)	V(8/12/2009)	I(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información
	V(16)	V(8/12/2009)	V(Elaboración)	I(584\$)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información
	V(17)	V(8/12/2009)	V(Elaboración)	V(Logística)	I(584\$)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información

		V(17)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	I(584\$)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(17)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	I(584\$)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(17)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	I(584\$)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(17)	V(8/12/2009)	V(Elaboración)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	I(584\$)	El sistema muestra un mensaje: "La información"

EP 1.3	Adicionar un documento cuando la información es incompleta.	I(Vacío )	V(8/12/2009)	V(Elaboración )	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información introducida es
		V(17)	I(Vacío )	V(Elaboración )	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información
		V(17)	V(8/12/2009)	I(Vacío )	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información
		V(17)	V(8/12/2009)	V(Terminado )	I(Vacío )	V(documento)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información



		V(17)	V(8/12/2009)	V(Terminado)	V(Logística)	I(Vacío)	V(Lilian)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(19)	V(8/12/2009)	V(Terminado)	V(Logística)	V(documento)	I(Vacío)	V(Analista de AFT)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(19)	V(8/12/2009)	V(Terminado)	V(Logística)	V(documento)	V(Lilian)	I(Vacío)	V(Errores en DCP)	V(Yosleidy)	El sistema muestra un mensaje: "La información"
		V(20)	V(8/12/2009)	V(Terminado)	V(Logística)	V(documento)	V(Lilian)	V(Analista de AFT)	I(Vacío)	V(Yosleidy)	El sistema muestra un mensaje: "La información"

		V(20)	V(8/12/2009)	V(Terminado)	V(Logística)	V(documento)	V(Lilian)	V(Análisis de AFT)	V(Errores en DCP)	I(Vacío)	El sistema muestra un mensaje: "La información"
EP 1.4	Cancelar	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se cancela la operación y se cierra

#### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Eliminar documento.	Se elimina el documento.	EP 1.1: Eliminar el documento correctamente.	<ul style="list-style-type: none"> <li>- Se selecciona el documento a eliminar.</li> <li>- Se presiona el botón <b>Eliminar</b>.</li> <li>- El sistema muestra un mensaje: "Está seguro que desea eliminar el documento."</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Eliminar el documento cuando ya está confirmado.	<ul style="list-style-type: none"> <li>- Se selecciona el documento a eliminar.</li> <li>- Se presiona el botón <b>Eliminar</b>.</li> </ul>

			<ul style="list-style-type: none"> <li>- El sistema muestra un mensaje: "El documento no se puede eliminar pues ha sido confirmado."</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		EP 1.3: Cancelar.	<ul style="list-style-type: none"> <li>- Se selecciona o no Eliminar documento.</li> <li>- Se presiona el botón <b>Cancelar.</b></li> </ul>

### Juegos de datos a probar

Id del escenario	Escenario	Documento	Respuesta del sistema	Resultado de la prueba
EP 1.1	Eliminar el documento correctamente.	N/A	El sistema muestra un mensaje: "Está seguro que desea eliminar el documento."	
EP 1.2	Eliminar el documento cuando ya está confirmado.	N/A	El sistema muestra un mensaje: "El documento no se puede eliminar pues ha sido confirmado."	

EP 1.3	Cancelar.	N/A	Se cancela la operación y se cierra la ventana.	
--------	-----------	-----	---	--

### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Confirmar documento.	El sistema debe permitir confirmar documentos.	EP 1.1: Confirmar documentos satisfactoriamente.	<ul style="list-style-type: none"> <li>– Se selecciona el documento.</li> <li>– Se presiona el botón <b>Confirmar</b>.</li> <li>– El sistema muestra un mensaje: “Se a confirmado el documento satisfactoriamente.”.</li> <li>– Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Confirmar un documento cuando el AFT proviene del almacén y no se han llenado todos sus datos.	<ul style="list-style-type: none"> <li>– Se selecciona el documento.</li> <li>– Se presiona el botón <b>Confirmar</b>.</li> <li>– El sistema muestra un mensaje de error: “No se puede confirmar el documento hasta que no llene todos sus datos.”.</li> <li>– Se presiona el botón <b>Aceptar</b>.</li> </ul>

### Juegos de datos a probar

Id del escenario	Escenario	Documento	Respuesta del sistema	Resultado de la prueba
EP 1.1	Confirmar documentos satisfactoriamente.	N/A	El sistema muestra un mensaje: "Se a confirmado el documento satisfactoriamente."	
EP 1.2	Confirmar un documento cuando el AFT proviene del almacén y no se han llenado todos sus datos.	N/A	El sistema muestra un mensaje de error: "No se puede confirmar el documento hasta que no llene todos sus datos."	

### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Imprimir documento.	El sistema permite imprimir un documento.	EP 1.1: Imprimir un documento satisfactoriamente.	<ul style="list-style-type: none"> <li>- El sistema permite seleccionar los elementos que se desean imprimir.</li> <li>- Se presiona el botón <b>Imprimir.</b></li> </ul>

		<p>EP 1.2: Imprimir un documento cuando el sistema permite realizar una vista previa.</p>	<ul style="list-style-type: none"> <li>- El sistema permite seleccionar los elementos que se desean imprimir.</li> <li>- Se presiona el botón <b>Vista previa.</b></li> <li>- El sistema permite mostrar una vista previa de lo que se desea imprimir.</li> <li>- Se presiona el botón <b>Imprimir.</b></li> </ul>
		<p>EP 1.3: Imprimir un documento cuando el sistema presenta errores con la red o la impresora.</p>	<ul style="list-style-type: none"> <li>- El sistema permite seleccionar los elementos que se desean imprimir.</li> <li>- Se presiona el botón <b>Imprimir.</b></li> <li>- El sistema muestra un mensaje: "Los datos no pueden ser impresos."</li> <li>- Se presiona el botón <b>Aceptar.</b></li> </ul>
		<p>EP 1.4: Cancelar</p>	<ul style="list-style-type: none"> <li>- Se selecciona o no la opción Imprimir documento.</li> <li>- Se presiona el botón <b>Cancelar.</b></li> </ul>

### Juegos de datos a probar

Id del escenario	Escenario	Documento	Respuesta del sistema	Resultado de la prueba
EP 1.1	Imprimir un documento satisfactoriamente.	N/A	Se imprimen los datos.	
EP 1.2	Imprimir un documento cuando el sistema permite realizar una	N/A	Se imprimen los datos.	
EP 1.3	Imprimir un documento cuando el sistema presenta errores con la	N/A	El sistema muestra un mensaje: "Los datos no pueden ser impresos.".	
EP 1.4	Cancelar.	N/A	Se cancela la operación y se cierra la ventana.	

### Condiciones de ejecución

- Se debe seleccionar el subsistema de Logística.

- Se debe seleccionar la opción **Activo Fijo Tangible/Gestión de documentos/Gestionar documentos**

#### Requisitos a probar:

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Adicionar AFT existentes a documentos.	Se adicionan los AFT existentes al documento seleccionado.	EP 1.1: Adicionar AFT existentes a documentos.	<ul style="list-style-type: none"> <li>– Seleccionar el documento al que se le desean adicionar AFT.</li> <li>– Se verifica que se puede adicionar.</li> <li>– Se registra el movimiento de AFT al documento.</li> <li>– Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Cancelar.	<ul style="list-style-type: none"> <li>– Se selecciona o no Adicionar AFT existentes a documentos.</li> <li>– Se presiona el botón <b>Cancelar</b>.</li> </ul>

#### Juegos de datos a probar

Id del escenario	Escenario	Documento	AFT	Respuesta del sistema	Resultado de la prueba
EP 1.1	Adicionar AFT existentes a documentos.	N/A	N/A	El sistema registra los datos. El sistema confirma el registro de los datos.	
EP 1.2	Cancelar.	N/A	N/A	Se cancela la operación y se cierra la ventana.	



**Requisitos a probar:**

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Adicionar AFT nuevos a documentos.	Se adicionan AFT nuevos a documentos con los datos: descripción, valor, vida útil, número de inventario, fecha de explotación, fecha de adquisición, depreciación.	EP 1.1: Adicionar AFT nuevos a documentos introduciendo los datos válidos.	<ul style="list-style-type: none"> <li>- Se selecciona el documento.</li> <li>- Se introducen los datos de activos fijos si es alta, un ajuste por sobrante, una apertura o una operación nueva definida con concepto entrada.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Adicionar AFT nuevos a documentos introduciendo los datos incorrectos.	<ul style="list-style-type: none"> <li>- Se selecciona el documento.</li> <li>- Se introducen los datos incorrectos.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.3: Adicionar AFT nuevos a documentos dejando campos requeridos en blanco.	<ul style="list-style-type: none"> <li>- Se selecciona el documento.</li> <li>- Se introducen los datos dejando campos requeridos en blanco.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.3: Cancelar.	<ul style="list-style-type: none"> <li>- Se selecciona o no Adicionar AFT nuevos a</li> </ul>

			documentos. – Se presiona el botón <b>Cancelar.</b>
--	--	--	--

### Juegos de datos a probar

Id del escenario	Escenario	Descripción	Valor	Vida útil	Número de inventario	Fecha de explotación	Fecha de adquisición	Depreciación	Respuesta del sistema	Resultado de la prueba
EP 1.1	Adicionar AFT nuevos a documentos introduciendo los datos válidos.	V(comp utadora)	V(250)	V(5)	V(512365)	V(26/11/2012)	V(26/11/2009)	V(5)	El sistema confirma el registro de los datos y muestra una ventana de confirmación: "Los datos han sido guardado"	
EP 1.2	Adicionar AFT nuevos a documentos introduciendo los datos incorrect	I(erf6?)	V(250)	V(5)	V(512365)	V(26/11/2012)	V(26/11/2009)	V(5)	El sistema señala los datos erróneos, permite corregirlos y registrarlos.	

os.	V(mesa)	I(kek4)	V(5)	V(512365)	V(26/11/2012)	V(26/11/2009)	V(5)	El sistema señala los datos erróneos, permite	
	V(mesa)	V(256)	I(años)	V(512365)	V(26/11/2012)	V(26/11/2009)	V(5)	El sistema señala los datos erróneos, permite	
	V(mesa)	V(256)	V(5)	I(primer)	V(26/11/2012)	V(26/11/2009)	V(5)	El sistema señala los datos erróneos, permite	
	V(mesa)	V(256)	V(5)	V(23658)	I(16/11/2009)	V(26/11/2009)	V(5)	El sistema señala los datos erróneos, permite	
	V(mesa)	V(256)	V(5)	V(23658)	V(26/11/2012)	I(26/11/2015)	V(5)	El sistema señala los datos erróneos, permite	

		V(mesa)	V(256)	V(5)	V(23658)	V(26/11/2012)	V(26/11/2009)	l(años)	El sistema señala los datos erróneos, permite
--	--	---------	--------	------	----------	---------------	---------------	---------	---

### Condiciones de ejecución

- Se debe identificar y autenticar ante el sistema y además debe tener los permisos para ejecutar esta acción.
- Se debe seleccionar el subsistema de Logística.
- Se debe seleccionar la opción **Activo Fijo Tangible/Gestión de documentos/Gestionar documentos**
- Se debe haber adicionado al menos un AFT al documento.

### Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
1: Eliminar AFT del documento.	Se elimina el AFT del documento.	EP 1.1: Eliminar AFT del documento correctamente.	<ul style="list-style-type: none"> <li>– Se selecciona el documento.</li> <li>– Se introducen los datos de activos fijos si es alta, un ajuste por sobrante, una apertura o una operación nueva definida con concepto entrada.</li> <li>– Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.2: Eliminar un AFT en el documento cuando no está	– Se selecciona el AFT a eliminar.

		confirmado.	<ul style="list-style-type: none"> <li>- Se solicita confirmación para eliminar el AFT.</li> <li>- Se presiona el botón <b>Eliminar</b>.</li> <li>- Se presiona el botón <b>Aceptar</b>.</li> </ul>
		EP 1.3: Cancelar.	<ul style="list-style-type: none"> <li>- Se selecciona o no Eliminar AFT del documento.</li> <li>- Se presiona el botón <b>Cancelar</b>.</li> </ul>

### Juegos de datos a probar

Id del escenario	Escenario	Respuesta del sistema	Resultado de la prueba
EP 1.1	Eliminar AFT del documento correctamente.	El sistema muestra un mensaje de confirmación: "El AFT del documento fue eliminado satisfactoriamente."	
EP 1.2	Eliminar un AFT en el documento cuando no está confirmado.	El sistema muestra un mensaje de error: "No se puede eliminar el AFT porque no está confirmado el documento."	

EP 1.3	Cancelar.	Se cancela la operación y se cierra la ventana.	
--------	-----------	---	--

### 3.2.1. Resultado de las Pruebas

Durante el transcurso de la etapa de pruebas a la aplicación se detectaron tres no conformidades que fueron resueltas, evitando con esto que impidieran el buen funcionamiento del módulo.

#### Registro de defectos y dificultades detectados

Requisito	No.	No conformidad	Aspecto correspondiente	Etapa de detección	Significativa	No significativa	Recomendación	Estado NC	Respuesta del Equipo de desarrollo
Buscar AFT.	1	No se puede buscar por el campo Grupo.	Buscar AFT.	Primera iteración.	x		Se recomienda revisar la funcionalidad.	Resuelta	Se arregló la búsqueda por Grupo.
Buscar AFT.	2	No se puede buscar por el campo Grupo.	Buscar AFT.	Primera iteración.	x		Se recomienda revisar la funcionalidad.	Resuelta	Se arregló la búsqueda por Grupo.
Confirmar	3	Cuando se da	DCP Confirma	Primera	x		Se recomienda revisar el envío	Resuelta	Se arregló el

Docum ento.	confirmar document o se muestra un mensaje en blanco.	r Docume ntos. EP 1.1	itera ción.			del mensaje.		mensaje.
----------------	---	--------------------------------	----------------	--	--	--------------	--	----------

### Conclusiones Parciales

En el presente capítulo se realizaron las matrices de integración de componentes tanto externa como interna a partir del diagrama de componentes con el objetivo de dar una visión general de cuales son los servicios que consume el módulo de Inventario del subsistema AFT, además se diseñaron casos de pruebas específicos para cada requisito funcional con el objetivo de probar la aplicación mediante las pruebas unitarias, se detectaron un conjunto de no conformidades que fueron resueltas logrando con esto un software listo para satisfacer las necesidades de los clientes.

## *Conclusiones Generales*

Durante el desarrollo del presente trabajo se realizó el diseño del marco teórico de la investigación donde se planteó la situación problemática, el problema a resolver, el objeto de estudio, el campo de acción, la idea a defender, el objetivo general y los objetivos específicos, quedando con esto definidas las acciones a ejecutar en cada fase del trabajo para llevar a buen término la investigación. Se analizaron los procesos de inventario en los AFT así como las herramientas que se utilizaron para el desarrollo de la solución, con el objetivo de conocer el negocio y las razones por las que la dirección del proyecto seleccionó estas herramientas. Se realizó el estudio de los sistemas existentes determinando porque no resultan soluciones factibles para Cuba. Se implementaron las interfaces a partir del prototipo entregado por los analistas, se implementó la capa de negocio que da respuesta a los requisitos propuestos, la capa de acceso a datos, las validaciones y excepciones, los servicios que se necesitan para la implementación de otros módulos y componentes, obteniéndose como resultado el módulo de Inventario del subsistema AFT que cumple con los requisitos propuestos, y que fue validado con pruebas unitarias, con la aplicación de estas se detectaron un conjunto no conformidades que fueron resueltas, logrando con esto una aplicación libre de errores y lista para satisfacer las necesidades de los clientes.



## *Recomendaciones*

Las recomendaciones propuestas para la continuidad del presente trabajo son:

- Ampliar las funcionalidades del módulo con los nuevos requerimientos que surjan por necesidades del cliente.
- Consultar este documento como material de estudio, guía y apoyo para posteriores desarrollos relacionados con los inventarios.

## Referencias Bibliográficas

1. **Wailgum, Thomas.** ABC: An introduction to ERP. [En línea] 2008.  
<http://www.cio.com/research/erp/edit/erpbasics.html>.
2. **Gonzalez Brito, Henry Raúl y Lezcano Lozada, Yuniesky.** *Sistema de Inventario Participativo de la UCI.* [Documento] Ciudad de la Habana : UCI, 2005.
3. **del Toro Ríos, José Carlos y González Brito, Henry Raúl.** *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : s.n., 2009.
4. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
5. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
6. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
7. **Minbas.cu.** Inventario SISCONT5. [En línea] 2007.  
<http://siscont.tm.minbas.cu/Docs/Documentos/Manuales/Inventario%20SISCONT5.pdf>.
8. **Wikidot.com.** Gestión de Inventario. [En línea] 2008. <http://mask.wikidot.com/gestion-del-inventario>.
9. **Fernando Madrigal Hernández.** ERP: ¿Qué es?, Beneficios e impactos en las Compañías. [En línea] 2005. <http://www.monografias.com/trabajos29/beneficios-erp/beneficios-erp.shtml> .
10. ConocimientosWeb. [En línea] <http://www.conocimientosweb.net/portal/term5098.html>.
11. **RodasXXI.cu.** Inventario. [En línea] <http://www.rodasxxi.cu/inventario.php>.
12. **Openbravo.com.** *Características OpenBravo.* [En línea] 2009.  
<http://www.openbravo.com/es/product/erp/features/>.
13. **SAP España.** *SAP: número uno en software ERP.* [En línea] 2007.  
<http://www.sap.com/spain/solutions/business-suite/erp/index.epx>.
14. **Gestión, Centro de Soluciones de.** *Modelo de desarrollo orientado a componentes.* [Documento] La Habana : Universidad de las Ciencias Informáticas, 2008.
15. **freedownloadmanager.org.** *Visual paradigm.* [En línea]  
[http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_\(M%C3%8D\)\\_14720\\_p](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p).

16. **Gonzalez Brito, Henry Raúl y Lezcano Lozada, Yuniesky.** *Sistema de Inventario Participativo de la UCI.* [Documento] La Habana : UCI, 2005.
17. **Cavsi.com.** Que es un sistema gestor de bases de datos. [En línea] 2007.  
<http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
18. **Netpecos.org.** *PostgreSql.* [En línea] 2008. [http://www.netpecos.org/docs/mysql\\_postgres/x15.html](http://www.netpecos.org/docs/mysql_postgres/x15.html).
19. **Guia-ubuntu.org.** PgAdmin\_III. [En línea] 2007. [http://guia-ubuntu.org/index.php?title=PgAdmin\\_III](http://guia-ubuntu.org/index.php?title=PgAdmin_III).
20. **Zend.com.** Zend Studio for Eclipse. [En línea] 2007. <http://www.zend.com/products/studio/>.
21. **Cova, Rubén Darío.** *Html.* [En línea] 2004. <http://www.geocities.com/covag/defhtml.html>.
22. **maestrosdelweb.com.** *IntroCSS.* [En línea] 8 de noviembre de 2007.  
<http://www.maestrosdelweb.com/editorial/introcsc/>.
23. **Webexperto.com.** *AJAX.* [En línea] 7 de Julio de 2006.  
<http://www.webexperto.com/articulos/articulo.php?cod=223>.
24. **Json.org.** *Sitio oficial de JSON.* [En línea] 2007. <http://www.json.org/json-es.html>.
25. **Desarrolloweb.com.** *CSS.* [En línea] septiembre de 2008.  
<http://www.desarrolloweb.com/articulos/26.php>.
26. —. *Qué es PHP.* [En línea] 1 de Mayo de 2009. <http://www.desarrolloweb.com/articulos/392.php>.
27. **Ciberaula.com.** *Introducción a Apache.* [En línea] 2008.  
[http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/).
28. **Echarte, Patxi.** *Frameworks de Zend para el desarrollo de aplicaciones PHP.* [En línea] 3 de julio de 2007. <http://www.eslomas.com/index.php/archives/2007/07/03/framework-de-zend-para-el-desarrollo-de-aplicaciones-php/>.
29. **Doctrine-Project.org.** [En línea] 2008. <http://www.doctrine-project.org>.
30. **Wordpress.com.** *Librería ExtJs.* [En línea] 06 de 08 de 2007.  
<http://vargasti.wordpress.com/2007/08/06/libreria-extjs/>.
31. **Firefox, Mozilla.** [En línea] 2009. <http://www.getfirefox.es/firefox-features>.
32. **Osmosislatina.com.** [En línea] 3 de enero de 2009.  
<http://www.osmosislatina.com/subversion/basico.htm>.

## Bibliografía

1. **Wailgum, Thomas.** ABC: An introduction to ERP. [En línea] 2008.  
<http://www.cio.com/research/erp/edit/erpbasics.html>.
2. **Gonzalez Brito, Henry Raúl y Lezcano Lozada, Yuniesky.** *Sistema de Inventario Participativo de la UCI.* [Documento] Ciudad de la Habana : UCI, 2005.
3. **del Toro Ríos, José Carlos y González Brito, Henry Raúl.** *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : s.n., 2009.
4. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
5. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
6. —. *Documento Visión. Proyecto ERP-Cuba.* [Documento] La Habana : UCI, 2009.
7. **Minbas.cu.** Inventario SISCONT5. [En línea] 2007.  
<http://siscont.tm.minbas.cu/Docs/Documentos/Manuales/Inventario%20SISCONT5.pdf>.
8. **Wikidot.com.** Gestión de Inventario. [En línea] 2008. <http://mask.wikidot.com/gestion-del-inventario>.
9. **Fernando Madrigal Hernández.** ERP: ¿Qué es?, Beneficios e impactos en las Compañías. [En línea] 2005. <http://www.monografias.com/trabajos29/beneficios-erp/beneficios-erp.shtml> .
10. ConocimientosWeb. [En línea] <http://www.conocimientosweb.net/portal/term5098.html>.
11. **RodasXXI.cu.** Inventario. [En línea] <http://www.rodasxxi.cu/inventario.php>.
12. **Openbravo.com.** *Características OpenBravo.* [En línea] 2009.  
<http://www.openbravo.com/es/product/erp/features/>.
13. **SAP España.** *SAP: número uno en software ERP.* [En línea] 2007.  
<http://www.sap.com/spain/solutions/business-suite/erp/index.epx>.
14. **Gestión, Centro de Soluciones de.** *Modelo de desarrollo orientado a componentes.* [Documento] La Habana : Universidad de las Ciencias Informáticas, 2008.
15. **freedownloadmanager.org.** *Visual paradigm.* [En línea]  
[http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_\(M%C3%8D\)\\_14720\\_p](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p).

16. **Gonzalez Brito, Henry Raúl y Lezcano Lozada, Yuniesky.** *Sistema de Inventario Participativo de la UCI.* [Documento] La Habana : UCI, 2005.
17. **Cavsi.com.** Que es un sistema gestor de bases de datos. [En línea] 2007.  
<http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
18. **Netpecos.org.** *PostgreSql.* [En línea] 2008. [http://www.netpecos.org/docs/mysql\\_postgres/x15.html](http://www.netpecos.org/docs/mysql_postgres/x15.html).
19. **Guia-ubuntu.org.** PgAdmin\_III. [En línea] 2007. [http://guia-ubuntu.org/index.php?title=PgAdmin\\_III](http://guia-ubuntu.org/index.php?title=PgAdmin_III).
20. **Zend.com.** Zend Studio for Eclipse. [En línea] 2007. <http://www.zend.com/products/studio/>.
21. **Cova, Rubén Darío.** *Html.* [En línea] 2004. <http://www.geocities.com/covag/defhtml.html>.
22. **maestrosdelweb.com.** *IntroCSS.* [En línea] 8 de noviembre de 2007.  
<http://www.maestrosdelweb.com/editorial/introcss/>.
23. **Webexperto.com.** *AJAX.* [En línea] 7 de Julio de 2006.  
<http://www.webexperto.com/articulos/articulo.php?cod=223>.
24. **Json.org.** *Sitio oficial de JSON.* [En línea] 2007. <http://www.json.org/json-es.html>.
25. **Desarrolloweb.com.** *CSS.* [En línea] septiembre de 2008.  
<http://www.desarrolloweb.com/articulos/26.php>.
26. —. *Qué es PHP.* [En línea] 1 de Mayo de 2009. <http://www.desarrolloweb.com/articulos/392.php>.
27. **Ciberaula.com.** *Introducción a Apache.* [En línea] 2008.  
[http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/).
28. **Echarte, Patxi.** *Frameworks de Zend para el desarrollo de aplicaciones PHP.* [En línea] 3 de julio de 2007. <http://www.eslomas.com/index.php/archives/2007/07/03/framework-de-zend-para-el-desarrollo-de-aplicaciones-php/>.
29. **Doctrine-Project.org.** [En línea] 2008. <http://www.doctrine-project.org>.
30. **Wordpress.com.** *Librería ExtJs.* [En línea] 06 de 08 de 2007.  
<http://vargasti.wordpress.com/2007/08/06/libreria-extjs/>.
31. **Firefox, Mozilla.** [En línea] 2009. <http://www.getfirefox.es/firefox-features>.
32. **Osmosislatina.com.** [En línea] 3 de enero de 2009.  
<http://www.osmosislatina.com/subversion/basico.htm>.

## *Glosario de Términos*

AFT: Activo Fijo Tangible.

Framework: Conjunto de APIs y herramientas destinadas a la construcción de un determinado tipo de aplicaciones de manera generalista.

Implementación: Proceso por el cual se escribe (en un lenguaje de programación), se prueba, se depura y se mantiene el código fuente de un programa informático.

Logística: Conjunto de medios y métodos necesarios para llevar a cabo la organización de una empresa o de un servicio.

Módulo: A efectos prácticos, hablar de *programas* es lo mismo que hablar de *productos de software* o hablar de *módulos de software*. Cada *módulo* es una parte del sistema, que se instala y funciona por separado, entrelazándose con otros módulos con los que intercambia información.

Plataforma: Entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones.

Software: Se refiere a los programas y datos almacenados en un ordenador.

- Los programas dan instrucciones para realizar tareas al hardware o sirven de conexión con otro software.