

**UNIVERSIDAD AMAZÓNICA DE PANDO  
ÁREA DE CIENCIAS Y TECNOLOGÍA  
CARRERA DE INGENIERÍA INFORMÁTICA**

**SISTEMA DE INFORMACIÓN PARA LA ADMINISTRACIÓN  
DE INVENTARIOS DE LA UNIDAD DE ALMACÉN DEL  
GOBIERNO AUTÓNOMO MUNICIPAL DE COBIJA**

**Trabajo de Titulación presentado en conformidad a los requisitos  
para obtener el Título de Ingeniero en Sistemas Informáticos**

**TUTOR:** Lic. Javier Patty Magne

**ASESOR:** MSc. Juan Carlos Huanca Guanca

**Cobija-Pando-Bolivia  
2011**

## **DEDICATORIA:**

A Dios, por darme la vida para poder estudiar esta carrera universitaria, a mi esposa Maday Usnayo Quispe y a mi hija Sara Uziely Chipana Usnayo, por darme su Amor y Fortaleza para seguir adelante, a mi Madre, Tío y Abuelos por el apoyo incondicional que me brindaron durante todo este tiempo, también a todos los docentes de la carrera de Ingeniería Informática por haberme transmitido sus conocimientos, como también a mis compañeros de estudio

# INDICE GENERAL

## CAPITULO I

1.1.	ANTECEDENTES.....	1
1.2.	DESCRIPCION DEL PROBLEMA .....	2
1.3.	SOLUCIÓN PROPUESTA.....	2
1.4.	OBJETIVOS Y ALCANCES DEL PROYECTO .....	3
1.4.1.	Objetivo general .....	3
1.4.2.	Objetivos específicos.....	3
1.4.3.	Alcances .....	3
1.5.	METODOLOGIA Y HERRAMIENTAS UTILIZADAS .....	5
1.6.	RESULTADOS OBTENIDOS .....	7
1.7.	ORGANIZACIÓN DEL DOCUMENTO.....	8

## CAPITULO II

2.1.	SISTEMA DE INFORMACION (SI).....	9
2.1.1.	Categorías de Sistemas de Información .....	9
2.1.1.1.	Sistemas para el procesamiento de transacciones.....	9
2.1.1.2.	Sistema de Información Administrativa .....	9
2.1.1.3.	Sistema para el soporte de Decisiones.....	9
2.1.2.	Bases de Datos.....	10
2.1.2.1.	Modelos de Base de Datos.....	11
2.1.3.	Sistema de Gestión de Base de Datos (SGBD) .....	12
2.1.3.1.	SGBD MySql.....	12
	i. Lenguaje de Definición de Datos (LDD) .....	13
	ii.Lenguaje de Manejo de Datos (LMD).....	13
2.1.4.	Metodología de Técnica de Modelado de Objetos (OMT) .....	14
2.1.4.1.	Modelos OMT.....	15
2.1.4.1.1.	Lenguaje Unificado de Modelado (UML) .....	16
	i. Modelo de Objetos .....	18

ii. Modelo Dinámico.....	19
iii. Modelo Funcional.....	20
2.1.4.1.2. Diagrama de Flujo de Datos.....	21
2.1.4.2. Fases de Desarrollo de la Metodología OMT.....	21
2.1.4.2.1. Fase de Conceptualización.....	21
2.1.4.2.2. Fase de Análisis del Sistema.....	22
2.1.4.2.3. Fase de Diseño del Sistema.....	24
2.1.4.2.4. Fase de Diseño de Objetos.....	28
2.1.4.2.5. Fase de Implementación y Pruebas.....	30
2.1.5. Intranet.....	36
2.1.6. Métricas de Calidad de Software.....	38
2.2. SISTEMA DE ADMINISTRACIÓN DE INVENTARIOS.....	<b>38</b>
2.3. ADMINISTRACIÓN DE ALMACENES.....	<b>38</b>
2.3.1. Inventarios.....	39
2.3.1.1. Métodos de valuación de inventarios.....	39
A. UEPS (últimos en entrar, primeros en salir).....	39
B. PEPS (primeros en entrar, primeros en salir).....	40
2.3.2. Kardex.....	40

### **CAPITULO III**

3.1. FASE DE CONCEPTUALIZACION.....	<b>41</b>
3.1.1. Requerimientos.....	41
3.1.2. Casos de uso.....	44
3.2. FASE DE ANALISIS.....	<b>60</b>
3.2.1. Modelo de Objetos.....	60
i. Identificación de Clases.....	61
ii. Diccionario de datos.....	61
iii. Asociación de Clases.....	62

3.2.2.	Modelo Dinámico.....	62
3.2.3.	Modelo Funcional.....	64
3.3.	<b>FASE DE DISEÑO DEL SISTEMA .....</b>	<b>65</b>
3.3.1.	Módulos del Sistema .....	65
3.3.2.	Arquitectura básica del sistema.....	66
3.3.3.	Procesos o funcionalidades de los módulos .....	66
3.3.4.	Estructura y diseño de la base de datos .....	67
3.4.	<b>FASE DE DISEÑO DE OBJETOS .....</b>	<b>69</b>
3.4.1.	Modelo de Objetos Detallado.....	69
3.4.2.	Modelo Dinámico detallado .....	72
3.4.3.	Modelo Funcional Detallado .....	80
3.5.	<b>FASE DE IMPLEMENTACION Y PRUEBAS.....</b>	<b>85</b>
3.5.1.	Valoración de Calidad del Software.....	92
3.5.1.1.	Calidad del Sistema de Almacén .....	92
a.	Funcionabilidad .....	93
b.	Fiabilidad.....	94
c.	Usabilidad.....	94
d.	Eficiencia.....	95
e.	Mantenimiento.....	95
f.	Portabilidad.....	95
 <b>CAPITULO IV</b>		
4.1.	CONCLUSIONES .....	98
4.2.	RECOMENDACIONES .....	99

## INDICE DE TABLAS

<b>TABLA 1.1:</b> Módulos del Sistema y sus Componentes .....	<b>4</b>
<b>TABLA 1.2:</b> Resumen de las Fases de la Metodología .....	<b>6</b>
<b>TABLA 2.1:</b> Datos de la Persona .....	<b>12</b>
<b>TABLA 2.2:</b> Funciones de las computadoras Cliente y Servidor .....	<b>27</b>
<b>TABLA 3.1:</b> Requerimientos Generales .....	<b>41</b>
<b>TABLA 3.2:</b> Requerimientos de Autenticación de Usuarios .....	<b>42</b>
<b>TABLA 3.3:</b> Requerimientos de Administrar Parámetros del Sistema .....	<b>42</b>
<b>TABLA 3.4:</b> Requerimientos de Administrar Personas .....	<b>42</b>
<b>TABLA 3.5:</b> Requerimientos de Administrar Almacén .....	<b>42</b>
<b>TABLA 3.6:</b> Requerimientos de Reportes.....	<b>43</b>
<b>TABLA 3.7:</b> Descripción del caso de uso Administrar Artículo .....	<b>46</b>
<b>TABLA 3.8:</b> Descripción del caso de uso Elaborar Nota de Entrada.....	<b>52</b>
<b>TABLA 3.9:</b> Descripción del caso de uso elaborar Nota de Salida.....	<b>53</b>
<b>TABLA 3.10:</b> Descripción del caso de uso Inventario Automático .....	<b>54</b>
<b>TABLA 3.11:</b> Descripción del caso de uso Cierre de Gestión .....	<b>55</b>
<b>TABLA 3.12:</b> Descripción de caso de uso Kardex por Artículo .....	<b>58</b>
<b>TABLA 3.13:</b> Descripción del caso de uso Kardex General .....	<b>59</b>
<b>TABLA 3.14:</b> Clases que se Identificaron para el Sistema .....	<b>61</b>
<b>TABLA 3.15:</b> Módulos y Funcionalidades.....	<b>67</b>
<b>TABLA 3.16:</b> Clases del sistema y su descripción.....	<b>69</b>
<b>TABLA 3.17:</b> Escala de Calificación para el Sistema de Almacén.....	<b>93</b>
<b>TABLA 3.18:</b> Resultado de la medida Funcionabilidad.....	<b>93</b>
<b>TABLA 3.19:</b> Resultado de la medida Fiabilidad .....	<b>94</b>
<b>TABLA 3.20:</b> Resultado de la medida Usabilidad .....	<b>94</b>
<b>TABLA 3.21:</b> Resultado de la medida Eficiente .....	<b>95</b>
<b>TABLA 3.22:</b> Resultado de la medida Mantenibilidad .....	<b>95</b>
<b>TABLA 3.23:</b> Resultado de la medida Portabilidad .....	<b>96</b>
<b>TABLA 3.24:</b> Resultados generales de la medida estándar ISO/IEC 9126.....	<b>96</b>



## INDICE DE FIGURAS

<b>FIGURA 2.1:</b> Diagrama de Interacción de Objetos para un Modelo estático .....	<b>18</b>
<b>FIGURA 2.2:</b> Diagrama de Interacción de Objetos con ventana gráfica .....	<b>19</b>
<b>FIGURA 2.3:</b> Diagrama de Transición de Estados .....	<b>20</b>
<b>FIGURA 2.4:</b> Forma Básica del DFD .....	<b>21</b>
<b>FIGURA 2.5:</b> Representación grafica de una Clase.....	<b>22</b>
<b>FIGURA 2.6:</b> Asociación de clases.....	<b>23</b>
<b>FIGURA 2.7:</b> Arquitectura Cliente/Servidor .....	<b>27</b>
<b>FIGURA 2.8:</b> Cliente/Servidor Dos Planos .....	<b>28</b>
<b>FIGURA 2.9:</b> Comparación de Aplicaciones web tradicionales y Aplicaciones web con la tecnología Ajax.....	<b>32</b>
<b>FIGURA 2.10:</b> Esquema del Funcionamiento de la Intranet .....	<b>37</b>
<b>FIGURA 3.1:</b> Diagrama de Casos de Uso General .....	<b>44</b>
<b>FIGURA 3.2:</b> Diagrama casos de uso Autenticación del usuario en el Sistema.....	<b>45</b>
<b>FIGURA 3.3:</b> Diagrama casos de uso Administrar Artículo.....	<b>45</b>
<b>FIGURA 3.4:</b> Diagrama de casos de uso Administrar Almacenes .....	<b>46</b>
<b>FIGURA 3.5:</b> Diagrama de casos de uso Administrar Proveedores .....	<b>47</b>
<b>FIGURA 3.6:</b> Diagrama casos de uso Administrar Rubros .....	<b>47</b>
<b>FIGURA 3.7:</b> Diagrama de casos de uso Administrar Tipos de Ingreso .....	<b>48</b>
<b>FIGURA 3.8:</b> Diagrama de casos de uso Administrar Tipos de Salida .....	<b>48</b>
<b>FIGURA 3.9:</b> Diagrama de casos de uso Administrar Tipo de Documentos.....	<b>49</b>
<b>FIGURA 3.10:</b> Diagrama de casos de uso Administrar Unidades de la Alcaldía.....	<b>49</b>
<b>FIGURA 3.11:</b> Diagrama de casos de uso Administrar Unidades de Medida .....	<b>50</b>
<b>FIGURA 3.12:</b> Diagrama de casos de uso Administrar Usuarios del Sistema .....	<b>50</b>
<b>FIGURA 3.13:</b> Diagrama de casos de uso Administrar Funcionarios del Municipio.....	<b>51</b>
<b>FIGURA 3.14:</b> Diagrama de casos de uso Elaborar Nota de Entrada.....	<b>51</b>
<b>FIGURA 3.15:</b> Diagrama caso de uso elaborar Nota de Salida .....	<b>52</b>
<b>FIGURA 3.16:</b> Diagrama de casos de uso Inventario Manual.....	<b>53</b>
<b>FIGURA 3.17:</b> Diagrama de casos de uso Inventario Automático .....	<b>54</b>
<b>FIGURA 3.18:</b> Diagrama de casos de uso Cierre de Gestión .....	<b>55</b>

<b>FIGURA 3.19:</b> Diagrama de casos de uso Buscar Artículo .....	56
<b>FIGURA 3.20:</b> Diagrama de casos de uso Ver Existencias .....	56
<b>FIGURA 3.21:</b> Diagrama de casos de uso Salidas Trimestrales .....	57
<b>FIGURA 3.22:</b> Diagrama de casos de uso Entradas Trimestrales.....	57
<b>FIGURA 3.23:</b> Diagrama de casos de uso Kardex por Artículo .....	58
<b>FIGURA 3.24:</b> Diagrama de casos de uso Kardex General .....	59
<b>FIGURA 3.25:</b> Diagrama casos de uso cierres de gestión .....	60
<b>FIGURA 3.26:</b> Asociación de Clases .....	62
<b>FIGURA 3.27:</b> Diagrama de Estados del Sistema.....	63
<b>FIGURA 3.28:</b> Diagrama de Flujo de Datos Ingreso de Artículos a Almacén .....	64
<b>FIGURA 3.29:</b> Diagrama de Flujo Salida de Artículos de Almacén .....	65
<b>FIGURA 3.30.:</b> Arquitectura del Sistema .....	<b>66</b>
<b>FIGURA 3.31.:</b> Diseño de la Base de Datos Almacén .....	68
<b>FIGURA 3.32.:</b> Diagrama de clases .....	71
<b>FIGURA 3.33.:</b> Diagrama de Estado del Caso de Uso Nota de Entrada.....	72
<b>FIGURA 3.34.:</b> Diagrama de Estado del caso de uso Nota de Salida.....	73
<b>FIGURA 3.35.:</b> Diagrama de Estado del caso de uso Kardex General .....	74
<b>FIGURA 3.36.:</b> Diagrama de Estado del caso de uso Cierre Gestion.....	75
<b>FIGURA 3.37.:</b> Diagrama de Secuencia del caso de uso Entrada de Articulos .....	76
<b>FIGURA 3.38.:</b> Diagrama de Secuencia Salida de Articulos a Almacen .....	77
<b>FIGURA 3.39.:</b> Diagrama de Secuencia del caso de uso Reportar Kardex .....	78
<b>FIGURA 3.40.:</b> Diagrama de Secuencia Generar Inventario .....	79
<b>FIGURA 3.41.:</b> Diagrama de Secuencia del caso de uso Cierre Gestion.....	80
<b>FIGURA 3.42:</b> Diagrama de Flujo de Datos del caso de uso Salida de Artículos .....	81
<b>FIGURA 3.43:</b> Diagrama de Flujo de Datos del caso de uso Ver Existencias .....	82
<b>FIGURA 3.44:</b> Diagrama de Flujo de Datos Reporte de Kardex General .....	83
<b>FIGURA 3.45:</b> Diagrama de Flujo de Datos Cierre de Gestión.....	84
<b>FIGURA 3.46.:</b> Código Fuente Conectar a Base de Datos .....	85
<b>FIGURA 3.47.:</b> Código Fuente NotaIngresoUno .....	86
<b>FIGURA 3.48.:</b> Código Fuente VerificarIngreso .....	87
<b>FIGURA 3.49.:</b> Código Fuente ValidarIngresoDetalles .....	88

<b>FIGURA 3.50.:</b> Organización del código fuente en el servidor .....	88
<b>FIGURA 3.51.:</b> Pantalla Ingreso al Sistema.....	89
<b>FIGURA 3.52.:</b> Menú principal del Sistema.....	89
<b>FIGURA 3.53.:</b> Creación de la Nota de Entrada a Almacén.....	90
<b>FIGURA 3.54.:</b> Reporte de la Nota de Ingreso a Almacén .....	90
<b>FIGURA 3.55.:</b> Reporte Kardex del un Artículo en Especifico.....	91
<b>FIGURA 3.56.:</b> Formulario Cierre de Gestión.....	91
<b>FIGURA 3.57.:</b> Formulario Nuevo Inventario .....	92

# **CAPITULO I**

## **INTRODUCCIÓN**

---

*Este capítulo presenta: la parte inicial de todo el documento como ser el problema dentro de la institución y el objetivo de cómo solucionar el problema enunciado.*

### **1.1. ANTECEDENTES**

El Gobierno Autónomo Municipal de Cobija, en su estructura organizacional se encuentra la Unidad de Almacén, quien cumple la función administrativa, que comprende actividades y procedimientos relativos a las entradas, salidas, registro, almacenamiento, distribución y control de los bienes de consumo en la entidad pública.

La Unidad de Almacén en la gestión 2009, contaba con un sistema denominado Sistema de Información y Control de Almacén (SICA), que hacía en control de entradas y salidas de artículos que son bienes del Municipio.

A mediados de la gestión 2010 el sistema SICA quedó obsoleto debido a que no existía la clave de acceso, por este motivo la Unidad de Almacén hizo una solicitud a la Unidad de Sistemas, para descifrar la clave de acceso y de esa manera poder trabajar con el sistema.

El sistema SICA, no se llegó a utilizar debido a que no se conocía el funcionamiento del sistema, también a la escasa confiabilidad sobre los datos que contenía.

Al mismo tiempo el sistema SICA tenía, varias observaciones por parte de Auditoría Interna, las cuales son:

- Se tenga concordancia del saldo de kardex con el saldo del inventario general.
- Los kardex deben ser impresos en forma individual y archivada en forma ordenada.
- Los reportes deben ser clasificados por categoría de materiales.
- El inventario debe reportar saldos completos anteriores y actualizados.
- En el kardex de almacén debe registrar el número de documento de ingreso y de salida.
- Se debe utilizar el método PEPS para el registro de los materiales o artículos.

## **CAPITULO I INTRODUCCIÓN**

---

### **1.2. DESCRIPCION DEL PROBLEMA**

La unidad de almacén del gobierno municipal presenta diversos problemas, como ser la elaboración de inventarios de forma manual, también la aplicación de métodos inadecuados para la valuación, al mismo tiempo se utiliza diferentes formatos en el que están registrados los artículos y los registros no están centralizados.

Esto conlleva a la formulación del problema principal.

**“Deficiente Administración de Inventarios de los artículos de la Unidad de Almacén, dependiente del Gobierno Autónomo Municipal de Cobija.”**

Esto ocasiona que la unidad tenga una información descentralizada y redundancia de datos también el desconocimiento de la cantidad existente de cada artículos y en algunos casos se llega a perder, y los inventarios que se generan son incompletos y datos desactualizados (Ver anexo A, Árbol de Problemas).

### **1.3. SOLUCIÓN PROPUESTA**

Como existe una deficiente administración de inventarios dentro de la unidad de almacén y estas deficiencias existen por no hay una lista actualizada de las cantidades existentes, tampoco existe un control de kardex de cada artículo que muestre cuanto de saldo existe en almacén. Por esta razón la solución propuesta es que se desarrolle e implemente un sistema de información que permita centralizar toda la información en un solo formato y que todos tengan acceso a la información necesaria. Lo que se pretende llegar utilizando el sistema son:

- Ayudar a mejorar el control de las entradas y salidas de los artículos de almacén.
- Que exista un registro detallado de todos los artículos que ingresan y salen para su posterior uso o referencia.
- Tener los kardex de artículos actualizados diariamente conforme se va realizando las entradas y salidas.
- Cumplir con las normativas exigidas en cuanto al ingreso de artículos y la salida de datos que se debe aplicar el método de valuación PEPS.
- Realizar el cierre de gestión de manera automática por medio del Sistema.

## **CAPITULO I INTRODUCCIÓN**

---

### **1.4. OBJETIVOS Y ALCANCES DEL PROYECTO**

#### **1.4.1. Objetivo general**

Desarrollar un Sistema de Información para la eficiente Administración de Inventarios de artículos de la Unidad de Almacén del Gobierno Autónomo Municipal de Cobija, haciendo uso de la metodología Object Modeling Technique (OMT)

#### **1.4.2. Objetivos específicos**

- Identificar los procesos de negocio sobre la administración de inventarios de la unidad de almacén
- Determinar las especificaciones de análisis y diseño para el sistema de información en base a la metodología OMT.
- Construir los módulos del sistema de información a partir de las especificaciones del diseño.
- Aplicar pruebas de funcionamiento del sistema de información
- Evaluar el calidad del sistema aplicando las métricas de calidad de software determinadas en la norma ISO 9126

#### **1.4.3. Alcances**

De acuerdo al análisis realizado en la institución bajo la metodología OMT, se llegó a determinar la necesidad de construir los módulos del sistema que se detallan a continuación.

- Módulo Administrar Parámetros del Sistema
- Módulo Administrar Personal
- Módulo de Administración de Almacén
- Módulo Reportes

Los módulos mencionados fueron desarrollados e integrados al Sistema, para que se implemente en la Unidad de Almacén Central dependiente del Gobierno Autónomo Municipal de Cobija.

Los módulos desarrollados se detallan en la siguiente Tabla.

**CAPITULO I**  
**INTRODUCCIÓN**

**TABLA 1.1:** Módulos del sistema y sus componentes

MODULOS	COMPONENTES
Módulo Administrar Parámetros del Sistema	<ul style="list-style-type: none"> <li>• Artículos</li> <li>• Almacenes</li> <li>• Proveedores</li> <li>• Rubros (Categorías)</li> <li>• Tipos de Entrada</li> <li>• Tipos de Salida</li> <li>• Tipos de Documentos</li> <li>• Unidades de la Alcaldía</li> <li>• Unidades de Medida</li> </ul>
Modulo Administrar Personal	<ul style="list-style-type: none"> <li>• Usuarios del Sistema</li> <li>• Personas Habilitadas</li> <li>• Lista de Funcionarios</li> </ul>
Módulo Administrar Almacén	<ul style="list-style-type: none"> <li>• Nota de Entrada</li> <li>• Nota de Salida</li> <li>• Inventario Manual</li> <li>• Inventario Automático</li> <li>• Cierre de Gestión</li> </ul>
Modulo de reportes	<ul style="list-style-type: none"> <li>• Buscar Artículos</li> <li>• Existencias</li> <li>• Salidas Trimestrales</li> <li>• Entradas Trimestrales</li> <li>• Kardex por cada Artículo</li> <li>• Kardex General de Artículos</li> </ul>

Fuente: Elaboración propia

Debido a que no existe la conexión de Red entre la Unidad de Almacén y las Otras Unidades del Municipio, no se llegó a desarrollar el Módulo de Pedidos, pero se llegó a configurar un servidor web en Debían para el funcionamiento del Sistema.

## **CAPITULO I INTRODUCCIÓN**

---

El Sistema fue desarrollado y funciona bajo la normativa referida a la Administración de Almacenes reglamentado por el Decreto Supremo N°. 0181, del 28 de junio del 2009, Normas Básicas del Sistema de Administración de Bienes y Servicios.

### **1.5. METODOLOGIA Y HERRAMIENTAS UTILIZADAS**

Para el desarrollo del Sistema de Administración de Inventarios de la Unidad de Almacén del Gobierno Autónomo Municipal de Cobija, se empleo la metodología OMT (Técnica de Modelado de Objetos), creada por James Rumbaugh y Michael Blaha en 1991.

OMT es una de las metodologías de análisis y diseño orientada a objetos más maduras y eficientes, la gran virtud que aporta esta metodología es su carácter de abierta, que le permite ser de dominio público, esto facilita su evolución para acoplarse a todas las necesidades actuales de la ingeniería de software.

La metodología OMT consta de las siguientes etapas para su construcción.

- a) En la **conceptualización** se identificó los requerimientos o necesidades que tiene la institución, también se identifica las necesidades que tiene el usuario en base a las entrevistas y encuestas que se realizaron, también en esta etapa se crean los diagramas de casos de uso estrechamente relacionados a las funcionalidades del sistema.
- b) En la fase de **análisis del sistema** se diseña el modelo de objetos que consiste en mostrar la estructura estática del sistema, de los objetos del sistema, en el modelo dinámico se muestra el comportamiento del sistema en el tiempo; para esto se utiliza el diagrama de estados, también se utiliza el modelo funcional para mostrar las transformaciones de los valores de los datos por medio de los diagramas de flujos de datos.
- c) Durante el **diseño de sistema**, se organiza el sistema en sub sistemas y se asigna a los sub sistemas tareas o flujos. También se crea el diseño de la base de datos.
- d) Durante el **diseño de objetos** se elabora el modelo de análisis y se proporciona una base detallada para la implantación. Se toman las decisiones necesarias para realizar un sistema sin entrar en los detalles particulares de un lenguaje o base de datos particular. El diseño de objetos inicia un corrimiento en el enfoque de la orientación del mundo

## CAPITULO I INTRODUCCIÓN

---

real del modelo de análisis hacia la orientación en la computadora requerida para una implantación práctica.

- e) En la fase de **codificación** se procede a implementar las clases y los objetos en un lenguaje de programación que en este caso el PHP5.
- f) Las **pruebas** se realizan introduciendo los datos en el sistema y probando su ejecución o si corre o no corre el sistema. Al mismo tiempo se corrige los errores posibles que existen dentro la programación.

En la siguiente Tabla se muestra el resumen de las fases de desarrollo de la metodología OMT.

**TABLA 1.2:** Resumen de las Fases de la metodología

FASE DE LA METODOLOGIA	DESCRIPCION
CONCEPTUALIZACION	<ul style="list-style-type: none"><li>• Se describe los requerimientos para la solución del sistema</li><li>• Se identifican de las necesidades del usuario</li><li>• Se elaboran los Casos de Uso</li></ul>
ANALISIS	<b>En esta etapa se construye</b> <ul style="list-style-type: none"><li>• Modelo Objeto.</li><li>• Modelo Dinámico.</li><li>• Modelo Funcional</li></ul>
DISEÑO DEL SISTEMA	Se determina la arquitectura del sistema en términos de subsistemas
DISEÑO DE OBJETOS	Se refina y optimiza el modelo de análisis, agregando conceptos de programación.
CODIGO	Se implementa las clases de objetos en un lenguaje de programación.
PRUEBAS	Se verifica el comportamiento de las clases y objetos que se encuentran descritos en los escenarios.

Fuente: Elaboración propia

## **CAPITULO I INTRODUCCIÓN**

---

Para el desarrollo del sistema se utilizó las herramientas o tecnologías de desarrollo las cuales son:

- HTML
- AJAX
- CSS
- PHP5
- MYSQL 5.0
- APACHE 2.0

Estas herramientas permitieron estructurar la arquitectura del software como cliente/servidor, donde HTML, AJAX, CSS, son parte del cliente, mientras que PHP5, MYSQL 5 y APACHE 2, forman parte del servidor.

### **1.6. RESULTADOS OBTENIDOS**

Lo que se llegó a obtener con el cumplimiento del proyecto fueron:

- Se llegó a identificar los procesos que lleva la institución para administrar las entradas y salidas de los artículos.
- Se hizo el análisis correspondiente del sistema en base a la metodología OMT.
- Se diseñó la estructura estática y dinámica del sistema en base al análisis realizado.
- Se llegó a construir los módulos correspondientes al Sistema que responde a las funcionalidades requeridas por los involucrados.
- Se realizó la etapa de prueba durante cuatro meses del Sistema en la Unidad de Almacén

Después de la implementación del sistema en la Unidad de Almacén, todas las entradas y salidas de artículos están en proceso de regularización en el Sistema (aspecto que no sucedía anteriormente), lo que se logró alcanzar con el Sistema fueron los siguientes.

- Los artículos ya se encuentran registrados en el sistema con sus respectivos códigos.
- Ya se realizó el primer cierre de gestión en base al inventario físico realizado.
- Las entradas de artículos a Almacén ya se realizan por medio del Sistema

## **CAPITULO I INTRODUCCIÓN**

---

- Las salidas de artículos de Almacén ya se realizan por medio del Sistema
- La existencia de los artículos se va actualizando automáticamente conforme se va haciendo las entradas y salidas.
- El kardex de cada artículo se va generando conforme se realizan las entradas y salidas de Almacén.
- El inventario de los artículos se genera automáticamente por medio del Sistema.

El grado de satisfacción por parte de los destinatarios es de un 90 %, por que se resolvió la mayor parte del problema, la parte administrativa es la que aun tiene dificultades dentro de la Unidad de Almacén.

### **1.7. ORGANIZACIÓN DEL DOCUMENTO**

El documento del presente proyecto se organiza en cuatro capítulos los cuales son.

- En el **capítulo I** se describe la parte introductoria al proyecto de grado.
- En el **capítulo II**, se describe toda la teoría que fundamenta el desarrollo y las herramientas aplicadas.
- En el **capítulo III**, se muestra el desarrollo del proyecto aplicando la metodología adoptada.
- En el **capítulo IV**, se encuentran las conclusiones y recomendaciones sobre el uso del Sistema.

# **CAPITULO I**

## **INTRODUCCIÓN**

---

1.1.	ANTECEDENTES .....	1
1.2.	DESCRIPCION DEL PROBLEMA .....	2
1.3.	SOLUCIÓN PROPUESTA .....	2
1.4.	OBJETIVOS Y ALCANCES DEL PROYECTO .....	3
1.4.1.	Objetivo general.....	3
1.4.2.	Objetivos específicos .....	3
1.4.3.	Alcances.....	3
1.5.	METODOLOGIA Y HERRAMIENTAS UTILIZADAS.....	5
1.6.	RESULTADOS OBTENIDOS.....	7
1.7.	ORGANIZACIÓN DEL DOCUMENTO .....	8

## **CAPITULO II**

### **MARCO TEORICO**

---

Este capítulo presenta las referencias teóricas referentes al proyecto de grado como ser: **definiciones relacionadas a los Sistemas de Información, modelos de desarrollo de software que permiten la administración de información, la metodología utilizada para el desarrollo y definiciones relacionadas a la Administración de Almacenes.**

#### **2.1. SISTEMA DE INFORMACION (SI)**

Según (Joseph & Andreu, 1998) un SI es “el conjunto formal de procesos que, actúan sobre una colección de datos estructurada de acuerdo con las necesidades de una institución. Recopila, elabora y distribuye parte de la misma en las operaciones de ella y en las actividades de dirección y control correspondientes, apoyando al menos en parte, la toma de decisiones necesaria para desempeñar las funciones y procesos de negocio de la institución de acuerdo con su estrategia”. Los sistemas de información se dividen en las siguientes categorías.

##### **2.1.1. Categorías de Sistemas de Información**

Existen diferentes tipos de sistemas de información para satisfacer las necesidades de una institución (Seen James, 2003), los cuales son.

###### ***2.1.1.1. Sistemas para el procesamiento de transacciones***

Sustituye los procedimientos manuales por otros basados en computadoras, trata con procesos de rutina bien estructurados.

###### ***2.1.1.2. Sistema de Información Administrativa***

Ayudan a las autoridades a tomar decisiones y resolver problemas. Se recurren a los datos almacenados como consecuencia del procesamiento de las transacciones, pero también de información, sobre tendencias económicas, demandas, costos de préstamos y otros.

###### ***2.1.1.3. Sistema para el soporte de Decisiones***

Proporciona información a las autoridades que deben tomar decisiones sobre situaciones particulares, apoyan la toma de decisiones en circunstancias que no están bien

## **CAPITULO II**

### **MARCO TEORICO**

---

estructuradas, una decisión se considera no estructurada si no existen procedimientos claros para tomarla.

El sistema de información que se desarrolló en el presente trabajo de grado comparte características tanto de tipo administrativo como de apoyo a la toma de decisiones, por que apoya las autoridades a tomar decisiones y resolver problemas en base a la información que está registrada. Al mismo tiempo proporciona información en un tiempo pertinente para el apoyo en la toma de decisiones.

Las solicitudes de sistemas de información están motivadas por uno de los siguientes tres objetivos generales:

- a) **Resolver un problema.** Actividades, procesos o funciones que en la actualidad, o quizá en el futuro, no satisfacen los estándares de desempeño o las expectativas.
- b) **Aprovechar una oportunidad.** Un cambio para ampliar o mejorar el rendimiento de la institución y su competitividad.
- c) **Dar respuesta a directivos.** Proporcionar información en respuesta a órdenes, solicitudes o mandatos originados por una autoridad legislativa o administrativa, llevar a cabo tareas de cierta manera, o también cambiar la información o tal vez el desempeño.

La información que fluye dentro del SI se almacena en una base de datos para su posterior uso.

#### **2.1.2. Bases de Datos**

La Base de Datos (BD) es justamente la colección de datos recopilados y estructurados que existe durante un periodo de tiempo. Por ejemplo, un libro contable, debido a su estructura, se puede considerar como tal. Una novela, por el contrario, no tiene estructura, y no se suele considerar. Generalmente, un sistema de información consta de una o más BD, junto con los medios para almacenarlas y gestionarlas, sus usuarios y sus administradores.

Una base de datos es un conjunto de datos almacenados de manera que se pueda acceder fácilmente a ellos, con la posibilidad de ordenarlos y relacionarlos. Es un conjunto de información estructurada en registros y almacenada en un soporte electrónico legible desde

## **CAPITULO II**

### **MARCO TEORICO**

---

un ordenador. Cada registro constituye una unidad autónoma de información que puede estar a su vez estructurada en diferentes campos o tipos de datos que se recogen en dicha BD.

La información contenida en una base de datos se distribuye normalmente en registros, la cual puede dividirse en unidades de datos de una misma categoría llamadas campos. Por ejemplo, en un directorio de miembros de una asociación, un registro será la ficha completa de cada uno de los socios. En cada registro se recogerán determinados datos, como el nombre, la profesión, la dirección o el teléfono, cada uno de los cuáles constituye un campo.

Las bases de datos se representan en un modelo de base de datos que se explica en el siguiente apartado.

#### ***2.1.2.1. Modelos de Base de Datos***

Existen diferentes modelos de datos dentro de una base de datos como ser.

- ✓ Modelo Distribuidas
- ✓ Modelo Relacional
- ✓ Modelo Orientado a Objetos

El modelo que se utilizó en el proyecto de grado es el Modelo Relacional, es la que permite representar los datos y la relación de manera simple, para representar y manipular los datos.

La estructura fundamental del modelo relacional es la "relación", es decir una tabla bidimensional constituida por filas (tuplas) y columnas (atributos), las relaciones representan las entidades que se consideran interesantes en la base de datos, cada instancia de la entidad encontrará sitio en una tupla de la relación, mientras que los atributos de la relación representan las propiedades de la entidad.

En la siguiente tabla se observa cómo se representa los dos datos de una persona dentro de una base de datos, donde cuyos atributos describen las características de la personas, cada tupla de la relación representa una persona concreta.

## CAPITULO II

### MARCO TEORICO

---

**TABLA 2.1:** Datos de la Persona

<b>Id</b>	<b>Nombre</b>	<b>Apellido</b>	<b>Nacimiento</b>	<b>Sexo</b>	<b>Estado Civil</b>
1	Juan	Loza	15-06-1971	M	Soltero
2	Isabel	Galvez	23-12-1969	F	Casada
3	Micaela	Ruiz	02-10-1985	F	Soltera

Fuente: Elaboración propia

En la actualidad las bases de datos se asocian con las computadoras, y su gestión no suele ser manual, sino altamente automatizada. La tecnología actual exige la delegación de la gestión de una base de datos a unos tipos de aplicaciones software específicas denominadas Sistemas de Gestión de Bases de Datos (SGBD).

#### **2.1.3. Sistema de Gestión de Base de Datos (SGBD)**

Según (Elmasri y Navathe, 1997) los sistemas de Gestión de Bases de Datos, son aplicaciones que permiten a los usuarios definir, crear y mantener la base de datos y proporciona un acceso controlado a la misma.

Los SGBD es una aplicación que interactúa con los usuarios de los programas de aplicación y la base de datos.

Algunos de los SGBD más conocidos son: SQL, ACCESS, POSTGRESS, MYSQL y otros, para el presente proyecto de grado se utilizó el manejador de base de datos MYSQL.

##### **2.1.3.1. SGBD MySql**

MySql Database Server es un administrador de Base de Datos (DB) de código fuente. Para agregar, acceder y procesar datos guardados en un computador, se necesita un administrador de DB como el MySql. Dado que las computadoras son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de los sistemas de información.

A si como todo SGBD, MySql utiliza el Lenguaje de Definición de Datos y el Lenguaje de Manejo de Datos que se definen a continuación.

MySql utiliza el lenguaje de consultas para manipular los datos que están dentro de la BD.

## **CAPITULO II**

### **MARCO TEORICO**

---

#### *i. Lenguaje de Definición de Datos (LDD)*

Un lenguaje de definición de datos es un lenguaje de programación proporcionado por el sistema de gestión de base de datos que permite definir estructuras de datos utilizando el lenguaje de consultas SQL, las instrucciones que se utilizan en esta categoría son.

- ✓ CREATE DATABASE
- ✓ CREATE TABLE
- ✓ DROP TABLE
- ✓ ALTER TABLE

#### *ii. Lenguaje de Manejo de Datos (LMD)*

Un lenguaje de manejo de datos así como el LDD es un lenguaje de programación para manipular los datos que están almacenados dentro de la DB, utilizando el lenguaje de consultas SQL, las instrucciones que se utilizan en esta categoría son.

- ✓ INSERT
- ✓ SELECT
- ✓ UPDATE
- ✓ DELETE

El software que administra la BD, tiene los siguientes objetivos que se muestran a continuación.

- ✓ Definir la Base de Datos mediante el Lenguaje de Definición de Datos, el cual permite especificar la estructura, tipo de datos y las restricciones sobre los datos, almacenándolo todo en la base de datos.
- ✓ Separar la descripción y manipulación de datos, permitiendo un mayor entendimiento de los objetos, además de flexibilidad de consulta y actualización de los datos.
- ✓ Permitir la inserción, eliminación, actualización, consulta de los datos mediante el Lenguaje de Manejo de Datos, lo que permite resolver el problema que presentan los sistemas de archivos, donde hay que trabajar con un conjunto fijo de consultas o la necesidad de tener muchos programas de aplicaciones.
- ✓ Proporcionar acceso controlado a la base de datos.

## **CAPITULO II**

### **MARCO TEORICO**

---

- a) Seguridad: los usuarios no autorizados no pueden acceder a la base de datos.
  - b) Integridad: mantiene la integridad y consistencia de la base de datos.
  - c) Control de Recurrencia: permite el acceso compartido a la base de datos.
  - d) Control de Recuperación: restablece la base de datos después de producirse un fallo de software o hardware.
- ✓ Gestionar la estructura física de los datos y su almacenamiento, proporcionando eficiencia en las operaciones de la base de datos y el acceso al medio de almacenamiento.
  - ✓ Proporcionar un mecanismo de vistas, que permita a cada usuario tener su propia vista o visión de la base de datos.
  - ✓ Permitir una fácil administración de los datos.

Un Sistema de Información además de orientarse hacia una categoría de información y administrar sus datos con un SGBD, funciona en una intranet dentro de la Institución.

#### **2.1.4. Metodología de Técnica de Modelado de Objetos (OMT)**

Según (Booch, 1996; Jacobson, 1992; Martiny Odell, 1994; Rumbaugh, 1996), la Técnica de Modelado de Objetos (Object Modeling Technique, OMT), se basa en un conjunto de conceptos que definen lo que es la tecnología Orientado a Objetos y una notación gráfica independiente.

La tecnología orientada a objetos propone una forma de pensar abstracta acerca de la resolución de problemas empleando conceptos del mundo real.

OMT es una metodología (de notación gráfica) para el desarrollo orientado a objetos que consiste en construir un modelo de un dominio de aplicación añadiéndole detalles de implementación durante el diseño del sistema.

Esta metodología consta de las siguientes fases:

- Conceptualización
- Análisis del sistema.
- Diseño del sistema.
- Diseño de objetos.
- Implementación y pruebas

## **CAPITULO II**

### **MARCO TEORICO**

---

La metodología emplea los modelos OMT que se integran en la fase de análisis y diseño de objetos.

#### ***2.1.4.1. Modelos OMT***

Un modelo es una abstracción de algo, cuyo objetivo es comprenderlo antes de construirlo. Dado que los modelos omiten los detalles no esenciales es más sencillo manipularlos que manipular la entidad original. La abstracción permite enfrentarse a la complejidad. Los ingenieros, artistas y artesanos han estado construyendo modelos durante miles de años para probar los diseños antes de ejecutarlos.

Para construir sistemas complejos, el desarrollador debe abstraer distintas vistas de los mismos, construir modelos utilizando notaciones precisas, verificar que los modelos satisfacen los requisitos del sistema y añadir, gradualmente, detalles para transformar los modelos en una implementación.

- ✓ La metodología OMT emplea tres clases de modelos para describir el sistema:
- ✓ Modelo de Objetos que describe los objetos del sistema y sus relaciones.
- ✓ Modelo Dinámico que describe las interacciones existentes entre objetos del sistema. Modelo Funcional que describe las transformaciones de datos del sistema.

Todos los modelos son aplicables en su totalidad de las fases del desarrollo y van adquiriendo detalles de implementación a medida que progresa el desarrollo.

Una descripción completa del sistema requiere los tres modelos. Un procedimiento típico de software contiene estos tres aspectos:

- Utiliza estructuras de datos (modelo de objetos),
- Secuencia las operaciones en el tiempo (modelo dinámico)
- Transforma valores (modelo funcional).

Cada modelo referencia a entidades de los otros modelos, los tres están relacionados entre sí. Las interconexiones entre los distintos modelos son limitadas y explícitas. Los buenos diseños aíslan los distintos aspectos del sistema y limitan el acoplamiento entre ellos.

## **CAPITULO II**

### **MARCO TEORICO**

---

El enfoque Orientado a Objetos se centra primordialmente en identificar objetos procedentes del dominio de la aplicación, ajustándoles después los procedimientos.

Para construir el modelo de objetos y el modelo dinámico se hace uso del lenguaje unificado de modelado (UML).

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

UML tiene una notación gráfica que permite representar en mayor o menor medida todas las fases de un proyecto de software basado en computadoras: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos y otros, hasta la implementación y configuración con los diagramas de despliegue.

Existen diversos métodos y técnicas orientadas a objetos, con muchos aspectos en común pero utilizando distintas notaciones, se Presentaban inconvenientes para el aprendizaje, aplicación, construcción y uso de herramientas, etc., además de pugnas entre enfoques y la falta de estandarización en la manera de representar en forma gráfica un modelo, impedía que los diseños gráficos realizados se pudieran compartir fácilmente entre estos distintos diseñadores, lo que genero la creación del UML como estándar para el modelado de sistemas de software, pero con posibilidades de ser aplicado a todo tipo de proyectos.

En el proyecto de grado se utilizó los siguientes diagramas especificados en la metodología.

- ✓ Diagrama de Casos de Uso
- ✓ Diagrama de Clases
- ✓ Diagrama de Estados
- ✓ Diagrama de Secuencia

#### ***a). Casos de Uso***

Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas.

Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo. En este tipo de diagramas intervienen algunos conceptos nuevos como.

## CAPITULO II

### MARCO TEORICO

---

**Actor.-** Es una entidad externa al sistema que se modela y que puede interactuar con él; un ejemplo de actor podría ser un usuario o cualquier otro sistema.

#### *b). Diagrama de Clases*

Los diagramas de clases representan un conjunto de elementos del modelo que son estáticos, como las clases y los tipos, sus contenidos y las relaciones que se establecen entre ellos.

Una clase representa un conjunto de objetos que tienen una estructura, un comportamiento y unas relaciones con propiedades parecidas. Describe un conjunto de objetos que comparte los mismos atributos, operaciones, métodos, relaciones y significativo. Los componentes de una clase son:

**Atributo.** Se corresponde con las propiedades de una clase o un tipo.

**Operación.** También conocido como método, es un servicio proporcionado por la clase que puede ser solicitado por otras clases y que produce un comportamiento en ellas cuando se realiza.

#### *c). Diagrama de Estados*

Representan la secuencia de estados por los que un objeto o una interacción entre objetos pasa durante su tiempo de vida en respuesta a estímulos (eventos) recibidos. Un estado en UML es cuando un objeto o una interacción satisfacen una condición, desarrolla alguna acción o se encuentra esperando un evento.

Cuando un objeto o una interacción pasa de un estado a otro por la ocurrencia de un evento se dice que ha sufrido una transición, existen varios tipos de transiciones entre objetos: simples y complejos. Como todas las metodologías Orientados a Objetos se envían mensajes, en este caso es la acción de la pueden enviar mensajes a uno o varios objetos destino.

#### *d). Diagrama de Secuencia*

Muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tiene lugar. En los diagramas de este tipo intervienen objetos que son instancias concretas

## CAPITULO II MARCO TEORICO

---

de una clase que participa en la interacción. Un diagrama de secuencia representa una forma de indicar el periodo durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento.

Los diagramas de secuencia permiten indicar cuál es el momento en el que se envía o se completa un mensaje mediante el tiempo de transición, que se especifica en el diagrama.

En los siguientes apartados se describe los tres modelos que utiliza la metodología OMT.

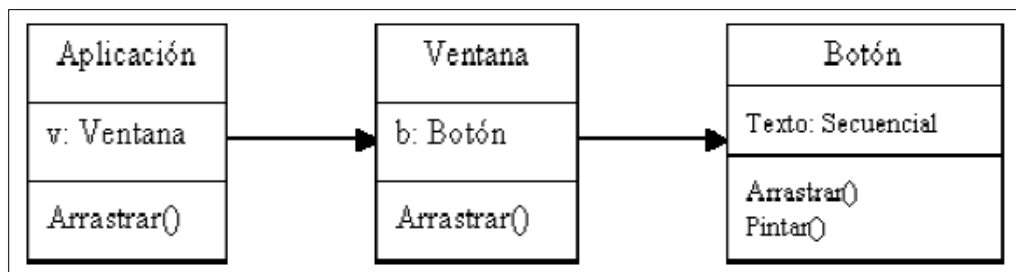
### *i. Modelo de Objetos*

El modelo de objetos describe la estructura estática (de datos), de los objetos del sistema (identidad, atributos y operaciones) y también sus relaciones.

Este modelo de objetos contiene **diagramas de objetos** (Ver Figura 2.1), que son unos grafos, cuyos nodos son clases de objetos y cuyos arcos son relaciones entre las clases. El diagrama contiene clases de objetos organizados en jerarquías, que comparten una estructura y comportamiento comunes y que están asociadas a otras clases, las cuales definen los atributos que lleva cada instancia de objeto y las operaciones que efectúa o sufre cada uno. En cada instancia de la clase se guardan los valores de esos atributos.

El diagrama de comunicación representa la forma en cómo un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen claramente las responsabilidades. La interacción entre objetos se modela gráficamente mediante un diagrama de interacción entre Objetos (Ver Figura 2.2).

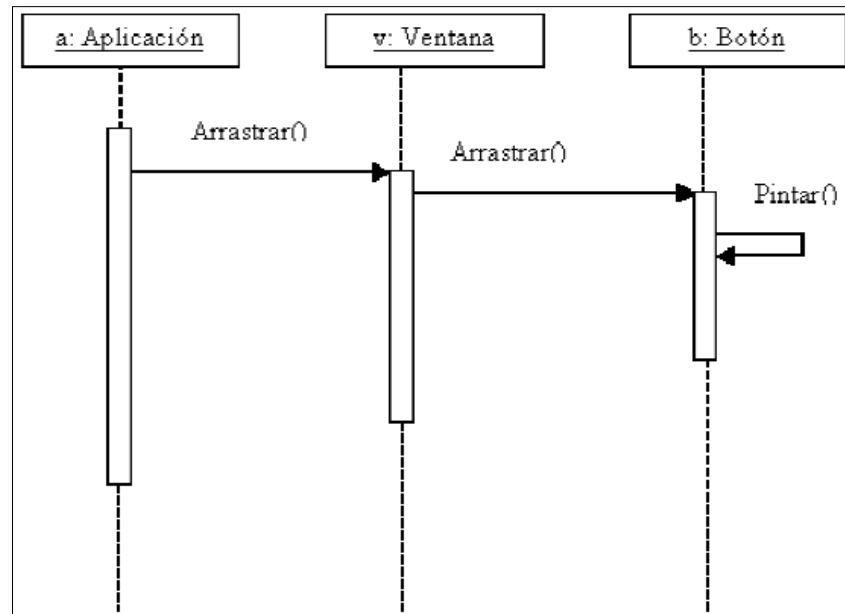
**FIGURA 2.1.** Diagrama de Interacción de Objetos para un Modelo Estático.



Fuente: (James Rumbaugh, 1996)

## CAPITULO II MARCO TEORICO

FIGURA 2.2. Diagrama de Interacción de Objetos con Ventana Gráfica.



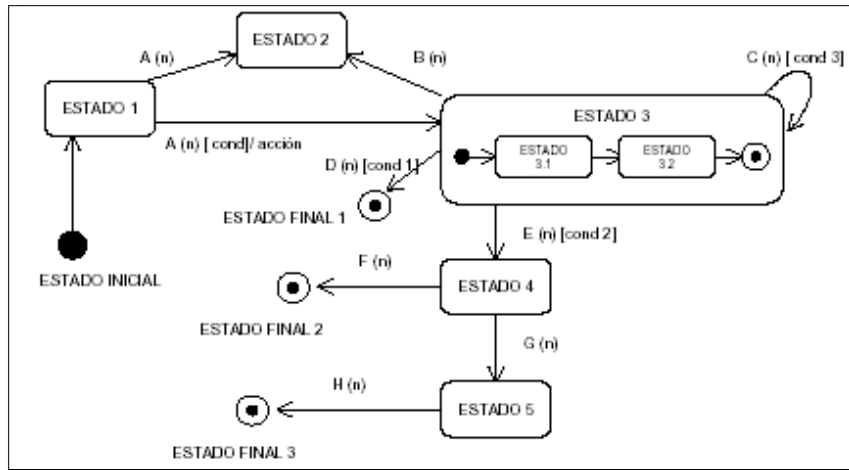
Fuente: (James Rumbaugh, 1996)

### ii. Modelo Dinámico

El modelo dinámico describe los aspectos de comportamiento (de control) de un sistema que cambian con el tiempo. El modelo dinámico se utiliza para especificar e implementar los aspectos del control del sistema. Los modelos dinámicos contienen **diagramas de estados** (Ver Figura 2.3). Un diagrama de estados es un grafo cuyos nodos son estados y cuyos arcos son transiciones entre estados causadas por sucesos o eventos.

Los Diagramas de Transición de Estados (DTE) se utilizan para describir el comportamiento de los objetos estableciendo sus vidas posibles. Una vida válida es una secuencia de eventos que caracteriza un comportamiento correcto para todos los objetos de la clase. Los principales componentes del diagrama son los estados y las flechas, que representan los cambios de estado.

**FIGURA 2.3.** Diagrama de Transición de Estados



Fuente: (James Rumbaugh, 1996)

Se especifican en este modelo la temporización y secuencia de operaciones (sucesos que marcan los cambios, secuencias de los mismos, estados que definen el contexto para los sucesos), y la organización de sucesos y de estados. El modelo dinámico captura el control, aquel aspecto de un sistema que describe las secuencias de operaciones que se producen sin tener en cuenta lo que hagan las operaciones, aquello a lo que afecten o la forma en la que estén implementadas.

**iii. Modelo Funcional**

El modelo funcional describe las transformaciones (de función), de valores de datos que ocurren dentro del sistema, captura lo que hace el sistema, independientemente de cuándo se haga o de la forma en que se haga. Las funciones se invocan como acciones en el modelo dinámico y se muestran como operaciones que afectan a objetos en el modelo de objetos. El modelo funcional contiene **diagramas de flujo de datos** (Ver Figura 2.4).

El Diagrama de Flujo de Datos (DFD) es una técnica gráfica que describe el flujo de información y las transformaciones que se aplican a los datos, conforme se mueven de la entrada a la salida. Es un grafo cuyos nodos son procesos y cuyos arcos son flujos de datos, se muestra las dependencias entre los valores y el cálculo de valores de salida a partir de los de entrada y de funciones, sin considerar cuando se ejecutan las funciones, ni siquiera si llegan a ejecutarse. También se le conoce como un grafo de flujo de datos o un diagrama de burbujas.

**FIGURA 2.4:** Forma básica de DFD.



Fuente: (James Rumbaugh, 1996)

#### ***2.1.4.1.2. Diagrama de Flujo de Datos***

Los diagramas de flujo de datos son un tipo de herramienta de modelado, permiten modelar todo tipo de sistemas, concentrándose en las funciones que realiza, y los datos de entrada y salida de esas funciones.

Un sistema puede representarse empleando varios diagramas de flujos de datos, cada flujo de datos puede representar una parte "más pequeña" del sistema.

Los DFD permiten una partición por niveles del sistema. El nivel más general se representa con un DFD global llamado diagrama de contexto.

El diagrama de contexto DFD representa a todo el sistema con una simple burbuja o proceso, las entradas y salidas de todo el sistema, y las interacciones con los usuarios.

#### ***2.1.4.2. Fases de Desarrollo de la Metodología OMT.***

Para el desarrollo del sistema se aplican las siguientes fases que se detallan a continuación.

##### ***2.1.4.2.1. Fase de Conceptualización***

En la fase de conceptualización se obtiene los requerimientos en base a encuestas y entrevistas realizadas en la institución, a partir de los requerimientos se procede a realizar los diagramas de casos de uso que representan la funcionalidad del sistema, también se llega a identificar los actores que intervienen en el manejo del sistema

## CAPITULO II

### MARCO TEORICO

---

#### 2.1.4.2.2. Fase de Análisis del Sistema

Durante el análisis se construye un modelo en el dominio de la aplicación sin tener en cuenta la implementación que se deberá efectuar posteriormente.

Deberá incluir aquella información que sea significativa desde el punto de vista del mundo real, presenta el aspecto externo del Sistema, comprensible para el cliente del Sistema, que proporciona una buena base para extraer los verdaderos requisitos (congruentes y realizables) en el Sistema.

El análisis comprende lo siguiente:

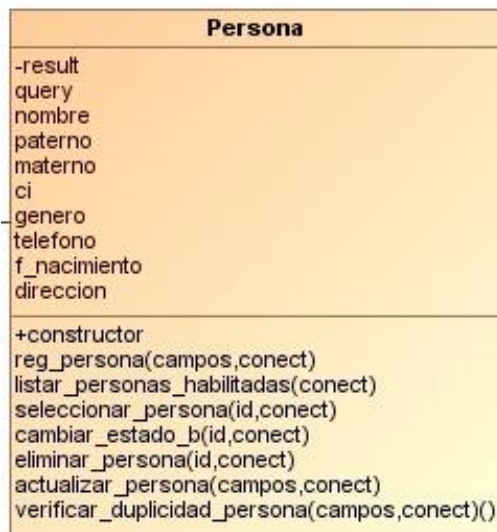
- a) Se construye un modelo de objetos.
- b) Se desarrolla un modelo dinámico.
- c) Se construye un modelo funcional.

Los pasos que se llevan a cabo para el modelo de objetos son:

#### i. Identificar las clases

Las clases se representan por rectángulos con tres compartimientos, en el primero se pone el nombre de la clase, en el segundo los atributos y en el tercero las operaciones o métodos.

FIGURA 2.5: Representación grafica de una Clase



Fuente: Elaboración propia

## CAPITULO II MARCO TEORICO

---

### ii. Preparar un Diccionario de clases (datos)

Un **diccionario de datos** es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

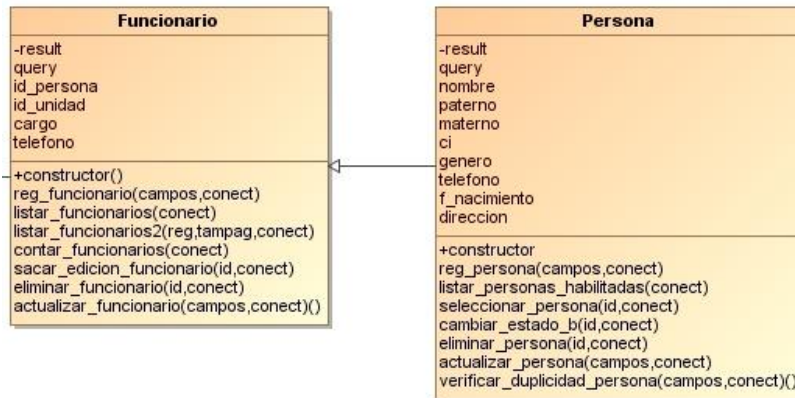
### iii. Identificar las asociaciones entre objetos

Una asociación es una dependencia entre dos o más clases. Las asociaciones, se representan por líneas que unen a las clases sobre las cuales se puede escribir el nombre de la asociación.

También se puede representar la relación " es parte de" o agregación que indica que el objeto está compuesto por objetos de las clases asociadas.

Se denota un pequeño diamante del lado de la clase que agrega y puede incluir multiplicidades.

**FIGURA 2.6:** Asociación de clases



Fuente: Elaboración propia

El documento que genera el modelo de objetos es el siguiente:

*Modelo de objetos = diagrama del modelo de objetos + diccionario de datos*

Los pasos para desarrollar el Modelo Dinámico son:

El Modelo Dinámico se representa por un diagrama de estados. El estado de un objeto es un conjunto de valores para sus atributos, en un cierto momento, el cual cambia al recibir un estímulo llamado evento.

## **CAPITULO II**

### **MARCO TEORICO**

---

- i. Preparar escenarios de una interacción típica.
- ii. Identificar eventos entre objetos preparando un trazado de eventos para cada escenario.
- iii. Construir los diagramas de estados.
- iv. Revisar los eventos entre los objetos para verificar su consistencia

El documento que genera el modelo dinámico es:

*Modelo dinámico = diagramas de estados*

Los pasos para desarrollar el Modelo Funcional son:

- i. Especifica el significado de las operaciones o métodos en el modelo de objetos y de las acciones en el modelo dinámico.
- ii. Muestra como se calculan los valores sin importar la secuencia, las decisiones ni la estructura de los objetos.
- iii. Se utilizan diagramas de flujo de datos para mostrar las dependencias funcionales.

El documento que genera el modelo funcional es:

*Modelo funcional = diagramas de flujo de datos + restricciones*

El documento que se genera en la fase de análisis del sistema contiene:

*Documento de análisis = definición del problema + modelo de objetos + modelo dinámico + Modelo funcional.*

#### **2.1.4.2.3. Fase de Diseño del Sistema**

Durante el diseño del Sistema, se selecciona la estructura de alto nivel del sistema. Existen varias arquitecturas canónicas que pueden servir como un punto de inicio adecuado. El paradigma orientado a objetos no introduce vistas especiales en el diseño del sistema, pero se incluye para tener una cobertura completa del proceso de desarrollo de software. Los pasos son:

- i. *Se organiza el sistema en subsistemas.*

Cada subsistema comparte alguna propiedad en común. Las relaciones entre los subsistemas pueden ser: *cliente – servidor* o *punto a punto*.

La descomposición se puede organizar por capas horizontales o particiones verticales (cada uno proporciona un servicio).

- ii. *Se identifica la concurrencia inherente en el problema.*

El modelo dinámico permite identificar la concurrencia en el sistema.

## **CAPITULO II**

### **MARCO TEORICO**

---

*iii. Se asignan los subsistemas a procesadores y a tareas.*

Cada subsistema se asigna a un procesador buscando satisfacer necesidades de rendimiento y minimizando la comunicación de los procesadores.

*iv. Se selecciona la estrategia para implementar y administrar los almacenes de datos (archivos o bases de datos).*

*v. Selección de la implementación del control del software.*

Existen dos tipos de control:

- El control interno. Esta dado por el flujo de control en el programa o proceso.
- El control externo. Esta dado por sucesos externos, los cuales pueden ser:
  - Control por procedimientos.
  - Control por sucesos.
  - Concurrentes

El documento que nos proporciona el diseño del sistema es:

*Documento de diseño del sistema = estructura de la arquitectura básica del sistema + decisiones estratégicas de alto nivel*

#### ***a. Arquitectura del Software***

El desarrollo de aplicaciones informáticas cada vez más complejas ha puesto de manifiesto la necesidad de potenciar el diseño de la arquitectura de los sistemas software, esto debido, a que tanto el tamaño como la complejidad aumenta, los algoritmos y estructuras de datos ya no constituyen el problema mayor para el diseño del producto de software, aparecen otros problemas con sistemas que contienen a una cantidad considerable de componentes, que presentan un sin número de dificultades tanto: De tipo estructural que incluye la organización y el control global; Protocolos de comunicación; Sincronización y acceso a los datos; Determinación del funcionamiento para el diseño de los elementos que compondrán el sistema; Distribución física; Composición del diseño de elementos; Rendimiento; Selección entre alternativas de diseño entre otros.

Debido al amplio contexto en el que se encuentra la arquitectura del software, existen diferentes definiciones que dan una muestra clara de lo que cada autor entiende, los apuntes del libro “Ingeniería del Software: un enfoque práctico” puntualiza que la arquitectura del

## **CAPITULO II**

### **MARCO TEORICO**

---

software alude a: “La estructura global del software y las maneras en que esa estructura proporciona integridad conceptual a un sistema”.

Otro de los conceptos sobre arquitectura software indica: “la arquitectura del sistema es un conjunto organizado de elementos y se utiliza para especificar las decisiones estratégicas acerca de la estructura y funcionalidad del sistema, las colaboraciones entre sus distintos elementos y su despliegue físico para cumplir responsabilidades definidas”.

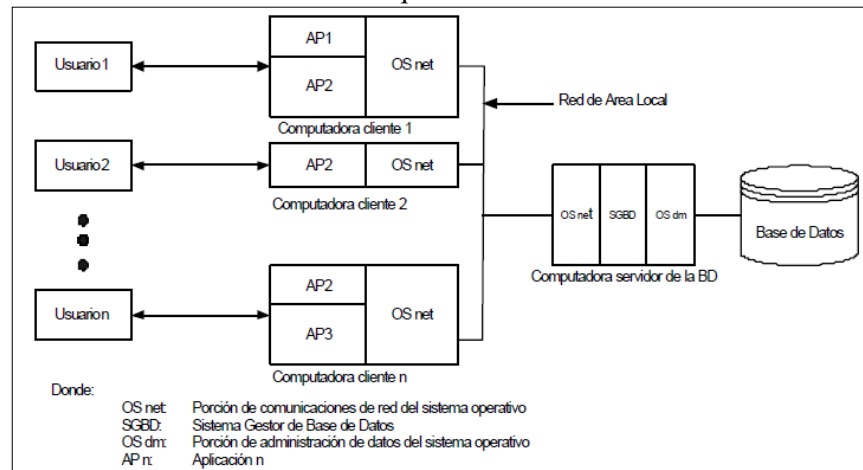
Según (Jacobson Ivar, Booch Grady y Rumbaugh James, 1996) definen que: “la arquitectura en un sistema software se describe mediante diferentes vistas del sistema en construcción, donde incluye los aspectos estáticos y dinámicos más significativos del sistema”.

#### ***b. El Modelo Cliente/Servidor***

Según (Pressmann, 2004) Para el desarrollo de la arquitectura planteada, se tomó en cuenta el modelo cliente servidor donde: “El cliente envía una solicitud que invoca procedimientos remotos en el centro servidor. Los procedimientos remotos pueden ser un conjunto de sentencias SQL (“Standard Query Language” Trad. “Lenguaje estándar de consulta”); se produce una transacción cuando una solicitud da lugar a la ejecución de procedimientos remotos y a la transmisión del resultado devuelto al cliente”.

El Modelo cliente servidor es un caso particular de la computación cooperativa, a la cual añade estructura, estableciendo una diferencia entre las funciones que deben ser compartidas y las funciones particulares a un nodo de la red. Las funciones compartidas residen en los nodos servidores y las funciones particulares en los nodos cliente, en la siguiente figura se muestra un ejemplo de la arquitectura cliente servidor en el cual cada uno de los n usuarios tiene su propia computadora de procesamiento de aplicaciones.

**FIGURA 2.7:** Arquitectura Cliente/Servidor



Fuente: Kroenke, David M.

Por otra parte se consideran las funciones tanto del Cliente, como del Servidor, que se especifican en la siguiente Tabla.

**TABLA 2.2.** Funciones de las computadoras Cliente y Servidor

**Funciones del Cliente:**

- Administrar la interfaz de usuario
- Aceptar datos del usuario.
- Procesar la lógica de la aplicación.
- Generar las solicitudes para la base de datos.
- Transmitir las solicitudes de la base de datos al servidor.
- Recibir los resultados del servidor.
- Dar formato a los resultados.

**Funciones del Servidor:**

- Aceptar las solicitudes de la base de datos de los clientes.
- Procesar las solicitudes de la base de datos.
- Llevar a cabo la verificación de integridad de la base de datos.
- Mantener los datos generales de la base de datos.
- Proporcionar control de acceso concurrente.
- Recuperar y optimizar el procesamiento de consultas y actualizaciones.

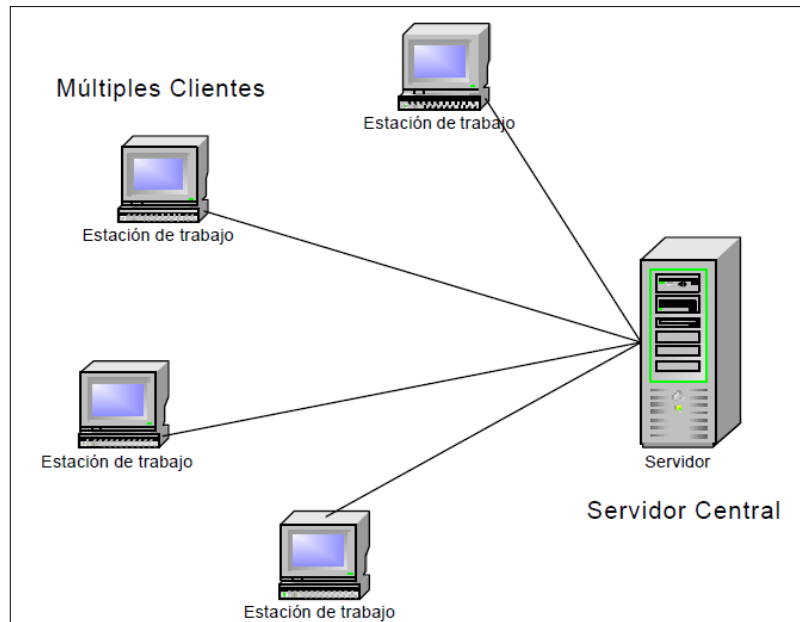
Fuentes: Kroenke, David M., op. cit. Pág. 484

## CAPITULO II MARCO TEORICO

---

Esta clasificación del modelo cliente servidor se basa en la distribución de los procesos y elementos entre sus componentes, centrándose en la parte física del mismo, en el que la administración de la interfaz gráfica se asocia a los clientes PC y la seguridad e integridad de los datos quedan asociados a ambientes mainframe o por lo menos a servidores locales y/o centrales, es el caso del SIGA (Sistema de Gestión de Almacén) que cuenta con una plataforma física ya definida, donde se implementó la arquitectura propuesta en el presente proyecto, como se muestra en la siguiente figura, donde los clientes son conectados a través de una Red LAN (“Local Area Network”, Trad. Red de Área Local”), a un Servidor Web Local, el cual, dependiendo del usuario que se conecte muestra el menú correspondiente. En la siguiente figura se muestra de manera resumida el modelo cliente/servidor, donde varios equipos como cliente están conectados a un servidor.

**FIGURA 2.8** Cliente/Servidor dos Planos



Fuente: <http://www.ar.geocities.com/>

### **2.1.4.2.4. Fase de Diseño de Objetos**

Durante el diseño de objetos se elabora el modelo de análisis y se proporciona una base detallada para la implantación. Se toman las decisiones necesarias para realizar un sistema sin entrar en los detalles particulares de un lenguaje o base de datos particular. El diseño de objetos inicia un corrimiento en el enfoque de la orientación del mundo real del modelo de análisis hacia la orientación en la computadora requerida para una implantación práctica.

## **CAPITULO II**

### **MARCO TEORICO**

---

Los pasos son:

- i. Se refinan las operaciones para el modelo de objetos a partir de los demás modelos:
  - ✓ Se busca una operación para cada proceso del modelo funcional.
  - ✓ Se define una operación para cada suceso del modelo dinámico.
- ii. Se diseñan algoritmos para implementar las operaciones y las estructuras de datos
- iii. Se optimizan las vías de acceso a los datos.
- iv. Se implementa el control del software completando la aproximación propuesta en el diseño del sistema.

Existen tres estrategias básicas para implementar el control:

- ✓ Construir un sistema controlado por procedimientos.
  - ✓ Crear un motor de máquina de estados que responde a una tabla de transiciones y acciones. (Se recomienda para ambientes dirigidos por eventos)
  - ✓ Establecer un control como tareas concurrentes. (Se requiere de un lenguaje que soporte la concurrencia)
- v. Se ajusta la estructura de clases incrementando la herencia.
  - vi. Se diseña la implementación de las asociaciones.

Las asociaciones conforman el pegamento en el modelo de objetos y proporcionan las vías de acceso entre los objetos. La implementación se hace dependiendo del tipo de asociación:

- ✓ Asociaciones unarias. Estas asociaciones se establecen solamente en una sola dirección y se pueden implementar por medio de apuntadores y si la multiplicidad es de "muchos", por medio de un conjunto de apuntadores.
- ✓ Asociaciones bidireccionales. Este tipo de asociaciones se pueden implementar de diferentes maneras:
  1. Atributos en una dirección.
  2. Como atributos en ambas direcciones.
  3. Implementar como un objeto separado por medio de diccionarios.

## CAPITULO II

### MARCO TEORICO

---

vii. Se determina la representación exacta de los atributos de los objetos.

Se empaquetan las clases y las asociaciones en módulos, el empaquetamiento implica:

- Ocultar la información interna a los ojos externos (construir cajas negras con interfaces claras).
- Determinar la coherencia de entidades, es decir, que cada clase o módulo debe de hacer una cosa y bien.
- Construcción de los módulos. Cada módulo debe de tener una cohesión funcional, esto es, un propósito bien definido.

El documento que se genera el diseño de objetos es:

*Documento de diseño de objetos = modelo de objetos detallado + modelo dinámico detallado + modelo funcional detallado.*

#### ***2.1.4.2.5. Fase de Implementación y Pruebas***

Durante la **implementación** se codifica, tanto las estructuras en el dominio de la aplicación como las estructuras en el dominio de la solución. La base que la sustenta es el **diseño de objetos**.

El código puede ser una simple transición de las decisiones de diseño a las características propias de un lenguaje.

La etapa de prueba es una actividad para determinar si el sistema está siendo construido correctamente. Tanto la implementación como las pruebas son dos etapas que están involucradas durante el análisis y diseño. Lo que significa que el análisis, diseño, la implementación y las pruebas están relacionadas durante el ciclo de vida de un sistema, lo cual da paso a tener una forma incremental de desarrollo, debido a que en cada etapa se pueden agregar características que tal vez en un primer nivel de abstracción se dejaron fuera. Para la codificación del sistema se utilizaron herramientas de desarrollo.

#### ***A. Herramientas de Desarrollo***

De acuerdo a la tecnología adoptada para el funcionamiento del Sistema, las herramientas de desarrollo que se utilizaron para el presente proyecto fueron los que se describen a continuación.

## **CAPITULO II**

### **MARCO TEORICO**

---

#### ***i. Lenguaje de etiquetas HTML***

Se optó por utilizar HTML para desarrollar las vistas del sistema con el que usuario interactuará constantemente por ejemplo con los formularios y las listas y reportes.

HTML, siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

Según (Miguel Ángel Alvares, 2009) HTML es el lenguaje con el que se definen las páginas web, básicamente se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web.

HTML se creó en un principio con el objetivo de divulgar información con texto y algunas imágenes. No se pensó que llegara a ser utilizado para crear área de ocio y consulta con carácter multimedia (lo que es actualmente la web), de modo que, el HTML se creó sin dar respuesta a todos los posibles usos que se le iba a dar y a todos los colectivos de gente que lo utilizarían en un futuro. Sin embargo, pese a esta deficiente planificación, se ha incorporado modificaciones con el tiempo, esto es lo que se conoce como los estándares del HTML.

#### ***ii. Lenguaje de Script AJAX***

El término AJAX se presentó por primera vez "Ajax: A New Approach to Web Applications", publicado por (Jesse James Garrett, 2005).

Hasta ese momento, no existía un término normalizado que hiciera referencia a un nuevo tipo de aplicación web que estaba apareciendo.

En realidad, el término AJAX es un acrónimo de *Asynchronous JavaScript + XML*, que se puede traducir como "JavaScript asíncrono + XML".

“Ajax no es una tecnología en sí mismo, en realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes.”

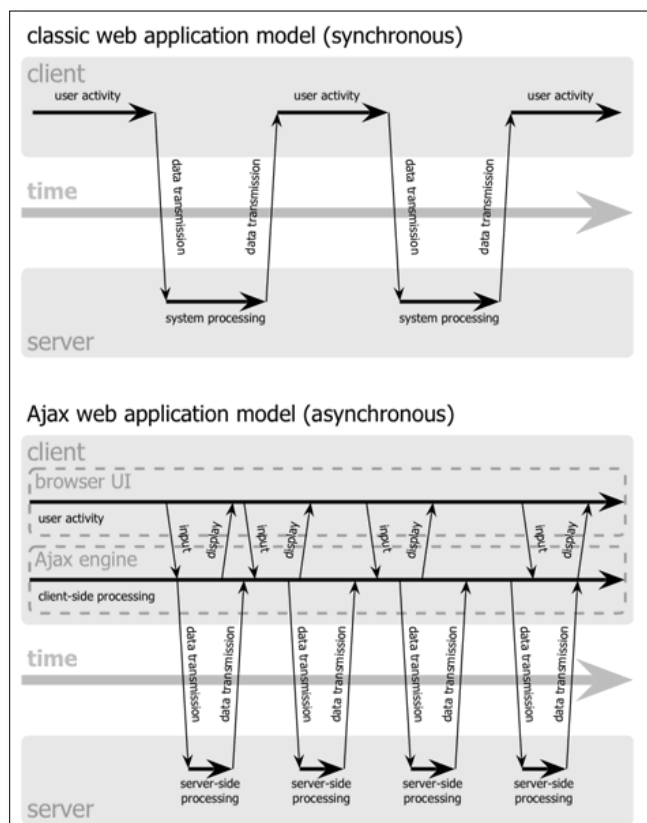
## CAPITULO II MARCO TEORICO

Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías.

El siguiente esquema muestra la diferencia más importante entre una aplicación web tradicional y una aplicación web creada con AJAX.

**FIGURA 2.9:** Comparación entre las comunicaciones síncronas de las aplicaciones web tradicionales y las comunicaciones asíncronas de las aplicaciones AJAX



Fuente: Adaptive Path

Las aplicaciones web tradicionales desencadenan las peticiones del usuario directamente al servidor, una vez procesada la petición el servidor devuelve una nueva página HTML al navegador del usuario.

## **CAPITULO II**

### **MARCO TEORICO**

---

Las aplicaciones construidas con AJAX eliminan la recarga constante de paginas mediante la creación de un elemento intermedio entre el usuario y el servidor, la nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor.

#### ***iii. Lenguaje de hojas de estilos CSS***

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

La separación de los contenidos y su presentación presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados “*documentos semánticos*”). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.

Mientras que el lenguaje HTML/XHTML se utiliza para *marcar* los contenidos, es decir, para designar lo que es un párrafo, lo que es un titular o lo que es una lista de elementos, el lenguaje CSS se utiliza para definir el aspecto de todos los contenidos, es decir, el color, tamaño y tipo de letra de los párrafos de texto, la separación entre titulares y párrafos, la tabulación con la que se muestran los elementos de una lista y otros.

#### ***iv. Lenguaje de programación PHP5***

El lenguaje de programación PHP (Hypertext Preprocessor) es un lenguaje script (no se compila para conseguir códigos máquina si no que existe un intérprete que lee el código y se encarga de ejecutar las instrucciones que contiene éste código), para el desarrollo de páginas web dinámicas del lado del servidor, cuyos fragmentos de código se intercalan fácilmente en páginas HTML, debido a esto, y a que es de Open Source (código abierto), es el más popular y extendido en la web.

PHP fue desarrollado originalmente por (Rasmus Ledford, 1994) como un CGI escrito en Perl que permitía la interpretación de un número limitado de comandos.

## **CAPITULO II**

### **MARCO TEORICO**

---

El sistema fue denominado Personal Home Page Tools y consiguió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas.

La última versión es PHP5 fue lanzado el 13 de julio del 2004, que utiliza el motor Zend-2 y presenta mejoras significativas y un entorno de programación orientado a objetos mucho más completo, que permite que el PHP proporcione un alto rendimiento a las aplicaciones Web empresariales a nivel de las plataformas J2EE y .NET.

El lenguaje de Programación a partir de su versión 5 adopta la mayor parte de las características de la Programación Orientada a Objetos.

#### ***v. Programación Orientada a Objetos***

De acuerdo a (Álvarez, 2001), la Programación Orientada a Objetos (POO) es una forma más cercana a como se puede expresar las cosas en la vida real. Con la POO se tiene que pensar las cosas de una manera distinta en términos de objetos, propiedades, métodos y otras. Es una técnica que aumenta la velocidad de desarrollo y hace que el mantenimiento sea más fácil.

La orientación a objetos mejora el diseño, desarrollo y mantenimiento del software ofreciendo una solución a largo plazo a los problemas y preocupaciones que han existido desde el comienzo: la falta de portabilidad del código y reusabilidad, que es difícil de modificar, ciclos de desarrollo largos y técnicas de codificación no intuitivas.

Según (Castells, 2005), la programación orientada a objetos reúne ciertas características como ser la: **abstracción, encapsulación, herencia y polimorfismo.**

#### ***a) Abstracción***

Consiste en la generalización conceptual de un determinado conjunto de objetos y de sus atributos y propiedades, dejando en un segundo término los detalles concretos de cada objeto, básicamente pasar del plano material (cosas que se tocan) al plano mental (cosas que se piensan).

## **CAPITULO II**

### **MARCO TEORICO**

---

#### ***b) Encapsulación***

Se refiere a la capacidad de agrupar y condensar en un entorno con límites bien definidos distintos elementos. Cuando se habla de encapsulación en general siempre se refiere, a la encapsulación abstracta.

#### ***c) Herencia***

Se aplica sobre las clases. O sea, de alguna forma las clases pueden tener descendencia, y ésta heredará algunas características de las clases "padres". Si se dispone las clases con un formato de árbol genealógico, se tendrá lo que se denomina una estructura jerarquizada de clases.

La herencia se implementa mediante un mecanismo que se denomina derivación de clases: las superclases pasan a llamarse clases base, mientras que las subclasses se constituyen en clases derivadas. El mecanismo de herencia está fuertemente entroncado con la reutilización del código en POO. Una clase derivada posibilita, el fácil uso de código ya creado en cualquiera de las clases base ya existentes.

#### ***d) Polimorfismo***

Esta propiedad, como su mismo nombre sugiere múltiples formas, se refiere a la posibilidad de acceder a un variado rango de funciones distintas a través del mismo interfaz. O sea, que, en la práctica, un mismo identificador puede tener distintas formas (distintos cuerpos de función, distintos comportamientos) dependiendo, en general, del contexto en el que se halle inserto.

El polimorfismo se puede establecer mediante la sobrecarga de identificadores y operadores, la ligadura dinámica y las funciones virtuales. El término sobrecarga se refiere al uso del mismo identificador u operador en distintos contextos y con distintos significados.

Además de las características de la POO, existen los conceptos básicos que también son parte este paradigma de programación las cuales son: **clases, objetos, métodos.**

## CAPITULO II

### MARCO TEORICO

---

#### *e) Clases*

Las clases son declaraciones de objetos, también se podrían definir como abstracciones de objetos. Esto quiere decir que la definición de un objeto es la clase. Cuando programamos un objeto y definimos sus características y funcionalidades en realidad lo que estamos haciendo es programar una clase.

#### *f) Objetos*

Según (Alberto Restrepo) "Un objeto es cualquier cosa que se ofrece a la vista y afecta los sentidos. Es una entidad tangible que exhibe algún comportamiento bien definido". Una definición más acorde con el paradigma de OO es: "objeto es cualquier cosa real o abstracta creada como una instancia de una clase que se pueda sentir, ver, manipular o imaginar. Por ejemplo, un diploma, una organización, un avión y un vuelo de avión, un vehículo y otros.

#### *g) Métodos*

Un **método** consiste en la implementación en una clase de un protocolo de respuesta a los mensajes dirigidos a los objetos de la misma. La respuesta a tales mensajes puede incluir el envío por el método de mensajes al propio objeto y aun a otros, también como el cambio del estado interno del objeto.

#### **2.1.5. Intranet**

Según (Patricio Pastor, 1998), la intranet es una red privada que se han creado utilizando las normas y protocolos de Internet. Aportan la interfaz de exploración del [www](#)<sup>1</sup>, a la información y servicios almacenados en una red de área local (LAN<sup>2</sup>) corporativa. .

Las intranets son atractivas por que disminuyen el costo de mantenimiento de una red interna y, al mismo tiempo, aumenta la productividad, ya que ofrece a los usuarios acceso más eficaz a la información y a los servicios que necesitan.

La diferencia entre intranet e internet no es mucha, básicamente la localización de la información y quien accede a ella, el internet es de acceso público, global, abierta a

---

<sup>1</sup>.- World Wide Web – Red de Redes en el mundo

<sup>2</sup>.- LAN ( Local Area Network) o Red de Área Local

## CAPITULO II

### MARCO TEORICO

---

cualquiera que tenga conexión a internet, mientras que la intranet está restringida a aquellas personas que están conectadas a la red local de la institución.

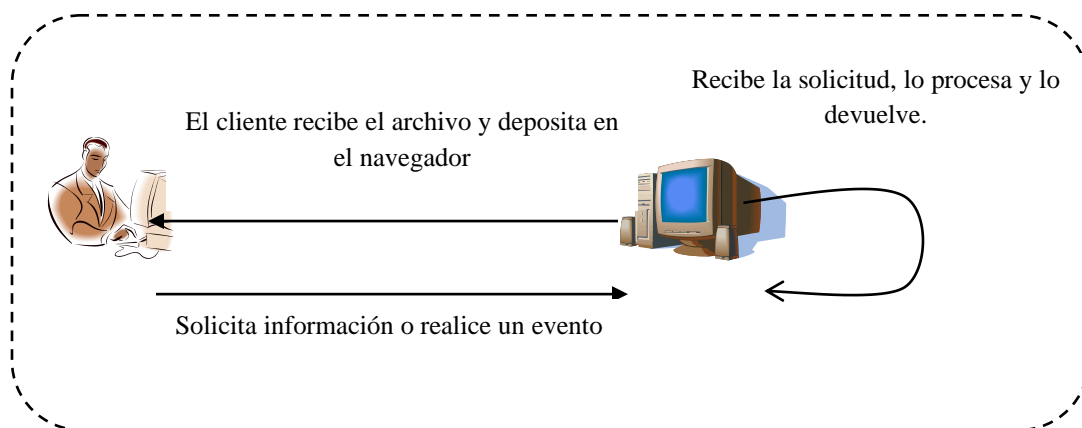
Además de eso, ambas funcionan esencialmente de la misma manera, con la misma tecnología TCP/IP para regular el tráfico de datos, del mismo modo, desde el punto de vista de las aplicaciones (e-mail, transferencia de archivos FTP, además de la Web).

Dentro de la intranet el acceso a la información es muy similar al acceso a internet, para comprender mejor el proceso se describe de la siguiente manera.

Todas las máquinas cliente (PCs locales conectados a una máquina Servidor de Web), utilizan un software especial, un navegador (**navegador**; por ejemplo, Mozilla Firefox o Internet Explorer).

- El usuario hace presión en un hipervínculo, que envía una petición de información al servidor, el hipervínculo contiene el nombre del servidor donde está localizada la información, y el nombre del archivo que ha de ser cargado en el navegador entonces.
- El servidor lee la petición, vuelca el archivo, y lo manda al cliente.
- El cliente recibe el archivo y lo deposita en la ventana del navegador (siempre que dicho archivo sea compatible con el web, por ejemplo, con formato HTML, ASP, PHP y otros). Puede ocurrir también que el archivo se deposite en discos o en aplicaciones especiales.

**FIGURA 2.10:** Esquema del funcionamiento de la intranet.



Fuente: Elaboración propia

## **CAPITULO II**

### **MARCO TEORICO**

---

El Sistema que se desarrollando en el presente proyecto grado, funciona en la red interna del municipio específicamente en la unidad de almacén donde existen tres equipos clientes y un servidor al cual se conecta mediante un Hub<sup>3</sup> (concentrador).

#### ***2.1.6. Métricas de Calidad de Software***

Conjunto de propiedades y de características de un producto o servicio, que le confieren aptitud para satisfacer las necesidades explícitas o implícitas (ISO/IEC 9126).

La Organización Internacional para la Estandarización (ISO), define la usabilidad según la Norma ISO/IEC 9126 como: *“La usabilidad se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso”*.

Esta definición hace énfasis en los atributos internos y externos del producto, los cuales contribuyen a su usabilidad, funcionalidad y eficiencia. La usabilidad depende no sólo del producto sino también del usuario. Por ello un producto no es en ningún caso específicamente usable, sólo tendrá la capacidad de ser usado en un contexto particular y por usuarios particulares. La usabilidad no puede ser valorada estudiando un producto de manera aislada (Bevan, 1994), para ver la norma completa (Ver Anexo B).

## **2.2. SISTEMA DE ADMINISTRACIÓN DE INVENTARIOS**

El sistema de administración de almacenes comprende en administrar los insumos o bienes que tiene una institución, esto se comprende mejor con la definición de administración de almacenes que se detalla a continuación.

## **2.3. ADMINISTRACIÓN DE ALMACENES**

Un almacén es un área operativa de la Dirección Administrativa, que tiene un sólo responsable de su administración, cada institución organiza el número necesario de almacenes, en función de las características técnicas y cualidades de los bienes.

---

<sup>3</sup> Un **Hub** es un dispositivo de red que funciona como un repartidor.

## **CAPITULO II**

### **MARCO TEORICO**

---

La administración de almacenes, es la función administrativa que comprende actividades y procedimientos relativos al ingreso, registro, almacenamiento, distribución, medidas de salvaguarda y control de los bienes de consumo en la entidad pública, además comprende el manejo de inventarios.

#### **2.3.1. Inventarios**

Los inventarios sirven para representar la existencia de bienes muebles e inmuebles que tiene la institución para comerciar con ellos, comprándolos y vendiéndolos tal cual o procesándolos primero antes de venderlos, en un período económico determinado, para tener un inventario actualizado se toma en consideración los métodos de valuación de inventarios.

##### ***2.3.1.1. Métodos de valuación de inventarios***

Existen dos sistemas de inventarios, el sistema periódico y el sistema permanente. Cuando la institución utiliza el sistema permanente, debe recurrir a diferentes métodos de valuación de inventarios, entre los que existen son el Método PEPS, Método UPES.

Según la norma administrativa de almacenes del estado boliviano se debe utilizar uno de los siguientes métodos: método PEPS o método UEPS, para el presente proyecto grado se opto utilizar el método PEPS.

#### **A. UEPS (últimos en entrar, primeros en salir)**

En este método lo que se hace es darle salida a los productos que se compraron recientemente, con el objetivo de que en el inventario final queden aquellos productos que se compraron de primero. Este es un método muy útil cuando los precios de los productos aumentan constantemente, cosa que es muy común en los países con tendencias inflacionarias.

El tratamiento que se la da a las devoluciones en compras es el mismo que se le da en el método PEPS, es decir que se le da salida del inventario por el valor de adquisición, esto debido a que como es apenas lógico, el producto se devuelve por el valor que se pago a la hora de adquirirlo.

## **CAPITULO II**

### **MARCO TEORICO**

---

#### **B. PEPS (primeros en entrar, primeros en salir)**

Este método consiste básicamente en darle salida del inventario a aquellos productos que se adquirieron primero, por lo que en los inventarios quedarán aquellos productos comprados más recientemente.

En cualquiera de los métodos las compras no tienen gran importancia, puesto que estas ingresan al inventario por el valor de compra y no requiere procedimiento especial alguno.

Si lo que se devuelve es un producto vendido a un cliente, este se ingresa al inventario nuevamente por el valor en que se vendió, pues se supone que cuando se hizo la venta, esos productos se les asignó un costo de salida según el método de valuación de inventarios manejado por la institución.

Para el proyecto se utiliza el método PEPS, porque es conveniente al momento de analizar hacer las adquisiciones de los bienes y que estos mismos salgan primero porque si se queda puede que el bien pierda su valor económico y el uso adecuado.

#### **2.3.2. Kardex**

Kardex de inventario es un registro que permite visualizar, información resumida acerca de las transacciones de cada artículo.

Puede realizar un seguimiento de los movimientos de los inventarios y de los costos de los bienes dentro del almacén.

Estos costos se calculan mediante los métodos Promedio continuo, UEPS continuo y PEPS continuo utilizando la unidad de medida definida para el artículo.

## **CAPITULO II**

### **MARCO TEORICO**

---

#### **INDICE CAPITULO II**

2.1. SISTEMA DE INFORMACION (SI) .....	9
2.1.1. Categorías de Sistemas de Información .....	9
2.1.1.1. Sistemas para el procesamiento de transacciones.....	9
2.1.1.2. Sistema de Información Administrativa .....	9
2.1.1.3. Sistema para el soporte de Decisiones.....	9
2.1.2. Bases de Datos.....	10
2.1.2.1. Modelos de Base de Datos.....	11
2.1.3. Sistema de Gestión de Base de Datos (SGBD) .....	12
2.1.3.1. SGBD MySql.....	12
i. Lenguaje de Definición de Datos (LDD) .....	13
ii. Lenguaje de Manejo de Datos (LMD).....	13
2.1.4. Metodología de Técnica de Modelado de Objetos (OMT) .....	14
2.1.4.1. Modelos OMT.....	15
2.1.4.1.1. Lenguaje Unificado de Modelado (UML).....	16
i. Modelo de Objetos .....	18
ii. Modelo Dinámico.....	19
iii. Modelo Funcional.....	20
2.1.4.1.2. Diagrama de Flujo de Datos .....	21
2.1.4.2. Fases de Desarrollo de la Metodología OMT.....	21
2.1.4.2.1. Fase de Conceptualización .....	21
2.1.4.2.2. Fase de Análisis del Sistema .....	22
2.1.4.2.3. Fase de Diseño del Sistema .....	24
2.1.4.2.4. Fase de Diseño de Objetos .....	28
2.1.4.2.5. Fase de Implementación y Pruebas .....	30
2.1.5. Intranet.....	36
2.1.6. Métricas de Calidad de Software.....	38
2.2. SISTEMA DE ADMINISTRACIÓN DE INVENTARIOS .....	38
2.3. ADMINISTRACIÓN DE ALMACENES .....	38
2.3.1. Inventarios .....	39
2.3.1.1. Métodos de valuación de inventarios.....	39
A. UEPS (últimos en entrar, primeros en salir).....	39

**CAPITULO II**  
**MARCO TEORICO**

---

B. PEPS (primeros en entrar, primeros en salir) ..... 40

2.3.2. Kardex ..... 40

### CAPITULO III DESARROLLO DEL PROYECTO

---

*Este capítulo presenta: el desarrollo del proyecto en base a la metodología adoptada a través del que se desarrolla cada una de las fases hasta llegar al producto final.*

#### 3.1. FASE DE CONCEPTUALIZACION

En la fase de conceptualización se identificaron los requerimientos los cuales se representan en diagramas de caso de uso, describiendo las funcionalidades del sistema como tal.

##### 3.1.1. Requerimientos

En esta etapa se identificó los requerimientos en base a las entrevistas y encuestas realizadas en la Unidad de Almacén Central (Ver Anexo F).

Los requerimientos generales pueden verse en la siguiente Tabla.

**TABLA 3.1:** Requerimientos Generales

<b>Ref.</b>	<b>Requerimientos</b>	<b>Tipo</b>
1	Reportar Artículos por Rubro	Evidente
2	Realice las notas de ingreso a almacenes	Evidente
3	Realice las notas de salida a almacenes	Evidente
4	Reportar los saldos existentes en almacén	Evidente
5	Reportar el Kardex General por rubro	Evidente
6	Reportar de Kardex por articulo	Evidente
7	Incorporar la técnica de Valuación de inventarios Primeros en Entrar Primeros en Salir PEPS	Evidente
8	Que el reporte de los inventarios finales que estén expresados (actualizados) de acuerdo con la unidad de fomento a la vivienda (UFV'S)	Evidente
9	Los reportes deben ser clasificados por categoría de materiales	Evidente

Fuente: Elaboración Propia en base a la encuesta realizada

A partir de los requerimientos generales se realizó un análisis más profundo para poder obtener los requerimientos específicos con el detalle que permita identificar las funcionalidades del sistema.

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

**TABLA 3.2:** Requerimientos de Autenticación de Usuarios

Ref.	Requerimientos	Tipo
1	Cada usuario que interactúe con el sistema de tener una identificación de usuario y su contraseña	Evidente
2	Una vez ingresado al sistema el usuario tiene acceso al menú.	Evidente
3	Cada usuario debe tener diferentes opciones de acuerdo al rol del usuario	Evidente

Fuente: Elaboración Propia

**TABLA 3.3:** Requerimientos de Administrar Parámetros del Sistema

Ref.	Requerimientos	Tipo
1	En el menú administrar parámetros del sistema se debe mostrar las opciones para ingresar datos.	Evidente
2	El sistema debe permitir registrar los siguientes datos como: artículos, almacenes, proveedores, rubros, tipos de ingreso, tipos de salida, unidades de la alcaldía y otros	Evidente
3	El sistema debe permitir ver los datos ingresados.	Evidente

Fuente: Elaboración Propia

**TABLA 3.4:** Requerimientos de Administrar Personas

Ref.	Requerimientos	Tipo
1	El sistema debe permitir registrar usuarios que estarán involucrados con el manejo del sistema y asignar el rol respectivo.	Evidente
2	El sistema también debe permitir registrar funcionarios del municipio, esto para realizar la entrega de material.	Evidente
3	El sistema debe permitir ver los datos ingresados.	Evidente

Fuente: Elaboración Propia

**TABLA 3.5:** Requerimientos de Administrar Almacén

Ref.	Requerimientos	Tipo
1	El sistema permite registrar las entradas de los materiales, elaborando una nota de entrada donde se registra los siguientes datos; tipo ingreso, tipo de documento, el proveedor, almacén, fecha ingreso, N° . de factura del proveedor, N° . de comprobante de pago contabilidad, detalle de ingreso y el valor de la UFV de compra.	Evidente
2	También permite registrar el artículo que está ingresando juntamente la cantidad y su precio unitario.	Evidente

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

3	Después de ingresar los datos, el sistema debe permitir calcular el sub total y el total de manera automática.	Evidente
4	El sistema debe permitir validar los datos ingresados en la nota de entrada	Evidente
5	El sistema debe permitir imprimir la nota de ingreso para su descargo.	Evidente
6	El sistema debe permitir ver las notas de ingreso anteriormente registradas.	Evidente
7	El sistema debe permitir realizar inventario automáticamente en base a las existencias actuales	Evidente
8	El sistema debe permitir realizar cierre de gestión en base al inventario del cuarto trimestre.	Evidente

Fuente: Elaboración Propia

**TABLA 3.6:** Requerimientos de Reportes

Ref.	Requerimientos	Tipo
1	Debe permitir buscar el articulo y mostrar la cantidad existente del mismo según la gestión que sea necesaria	Evidente
2	Debe reportar las existencias de todos los artículos ordenando por categoría (rubro)	Evidente
3	Debe permitir reportar las salidas trimestrales que se realizaron durante una gestión.	Evidente
4	Debe permitir reportar las entradas trimestrales que se realizaron durante una gestión.	Evidente
5	El sistema debe permitir reportar el kardex de un artículo en específico para ver las E y S del artículo según la gestión.	Evidente
6	Debe permitir reportar el kardex general de todos los artículos activos ordenados por categoría (rubro).	Evidente
7	Debe permitir ver los cierres de gestión realizadas anteriormente.	Evidente
8	El sistema debe permitir imprimir todos los reportes realizados para evidenciar los datos que contiene el sistema	Evidente

Fuente: Elaboración Propia

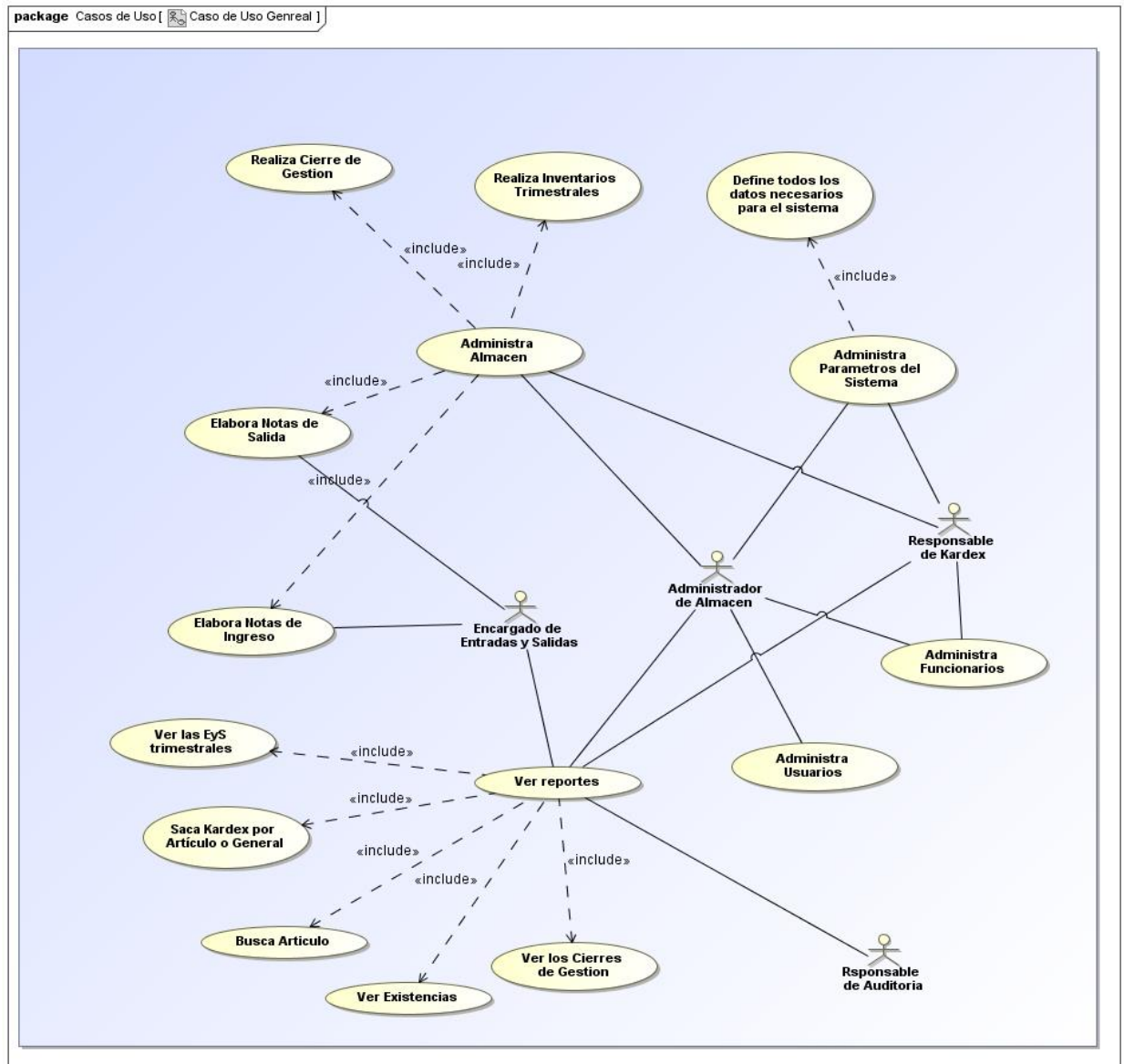
**3.1.2. Casos de uso**

Para poder ver el panorama general del sistema se elaboró un Caso de Uso General donde muestran los actores que intervienen en el sistema y que opciones tienen cada uno con sus respectivos roles.

A continuación se muestra el caso de uso general.

### CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.1: Diagrama de casos de uso general

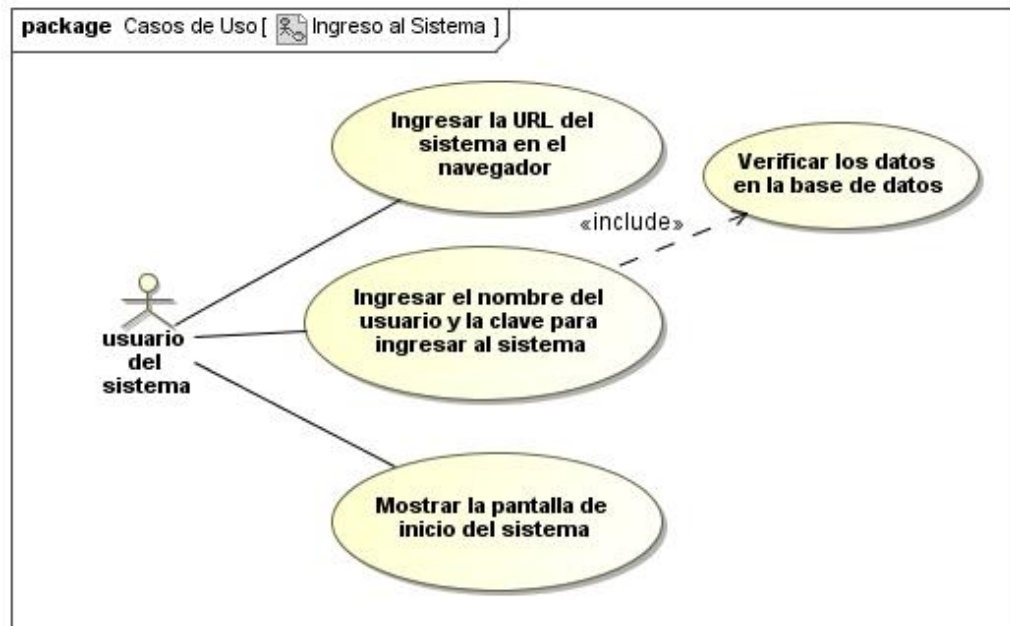


Fuente: Elaboración Propia

Después de elaborar el caso de uso general se elaboran los casos de uso más detallados que se muestran a continuación.

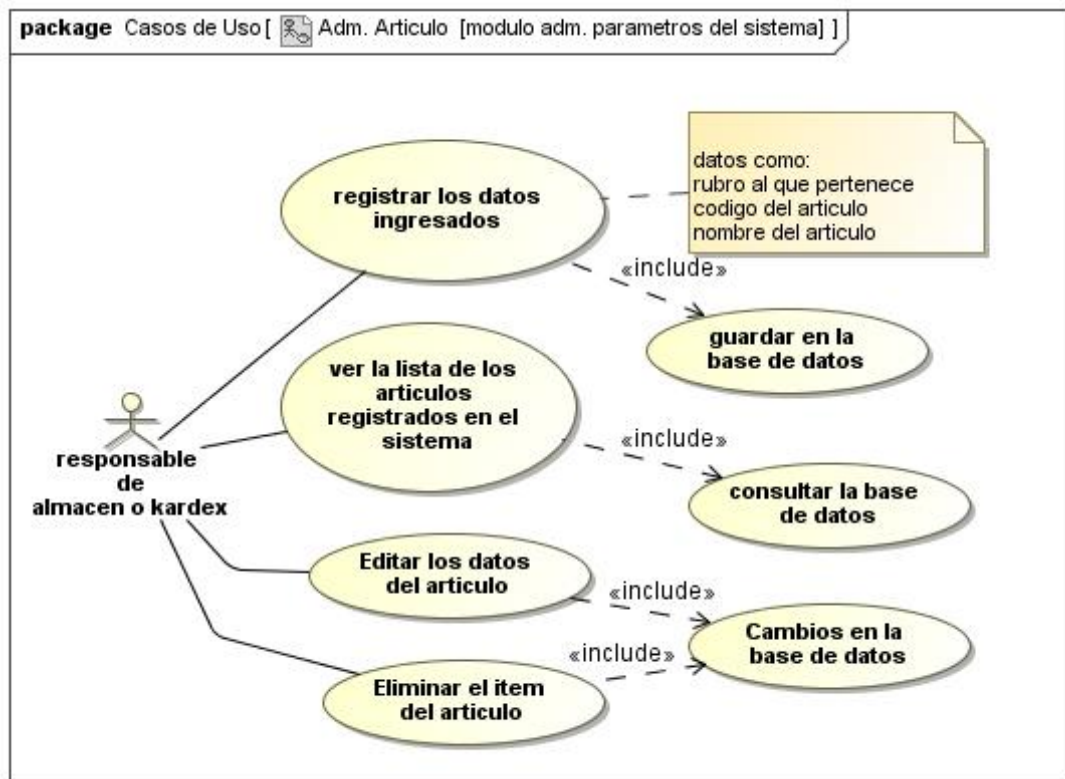
### CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.2: Diagrama casos de uso Autenticación del usuario en el Sistema



Fuente: Elaboración propia

FIGURA 3.3: Diagrama casos de uso Administrar Artículo



Fuente: Elaboración propia

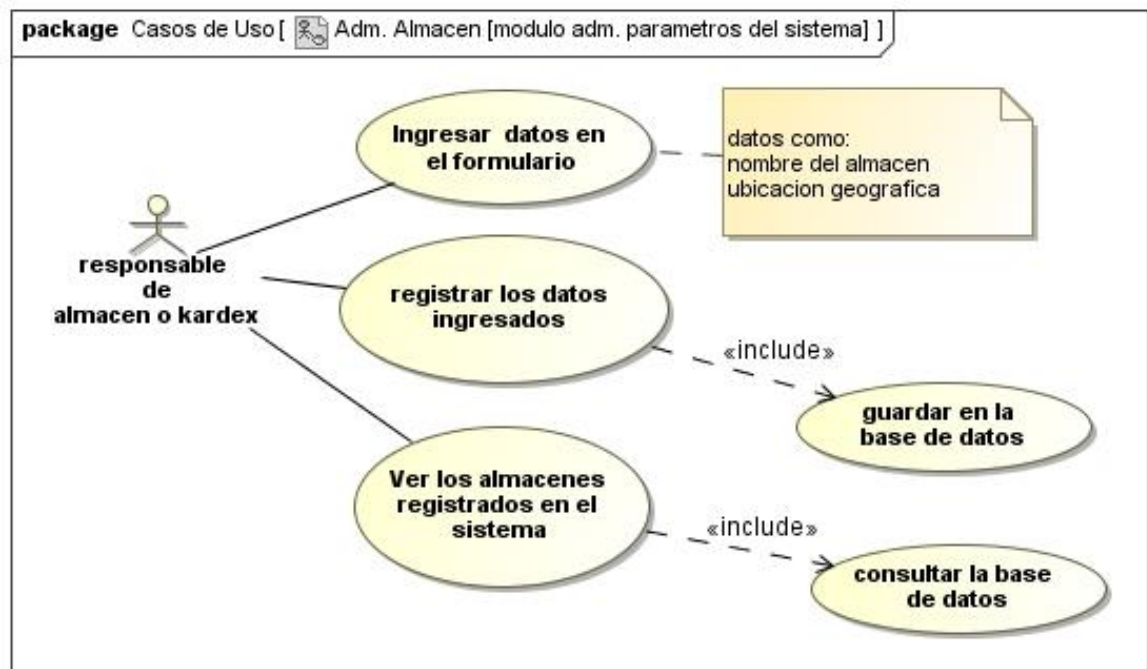
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**TABLA 3.7:** Descripción del caso de uso Administrar Artículo

Caso de uso	Descripción
Registra los datos ingresados	Cuando se agrega un nuevo artículo se registra los siguientes datos: rubro al que pertenece, código del artículo, nombre del artículo y su descripción.
Listar los artículos registrados	En la lista de Artículos simplemente se observa una lista de todos los artículos registrados y ordenados por rubros.
Editar datos del artículo	Permite editar al código del artículo y el nombre del artículo.
Eliminar el ítem del artículo	Elimina el artículo de la base de datos.

Fuente: Elaboración Propia

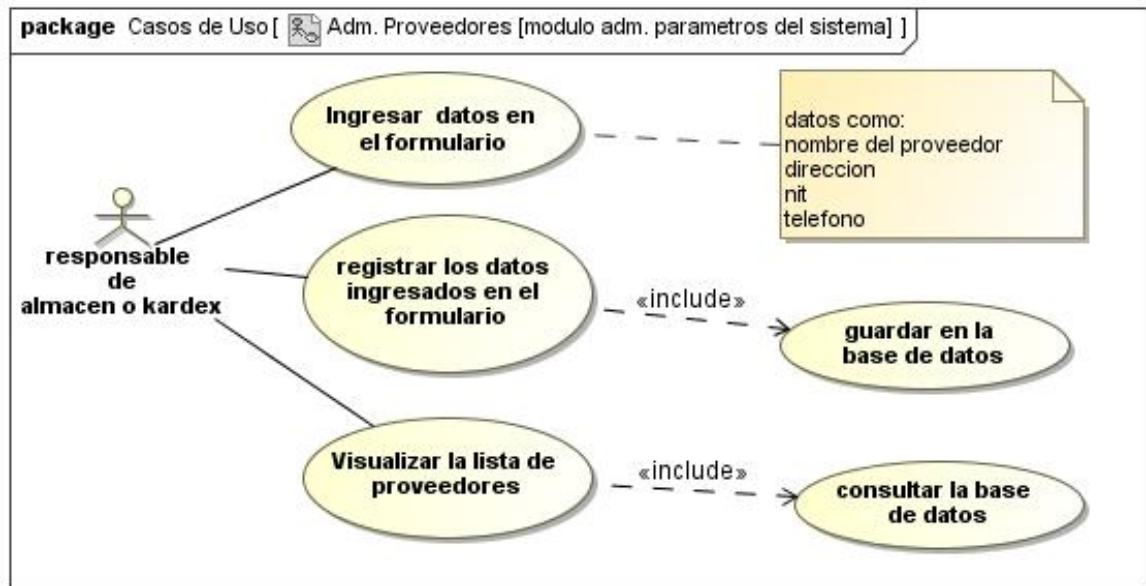
**FIGURA 3.4:** Diagrama de casos de uso Administrar Almacenes



Fuente: Elaboración Propia

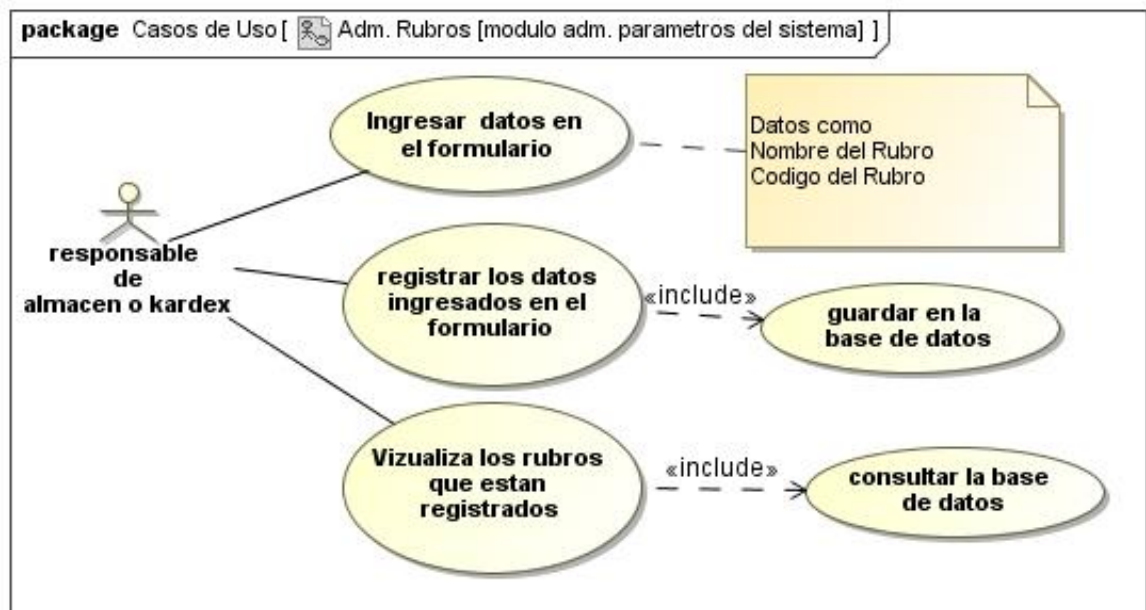
### CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.5:** Diagrama de casos de uso Administrar Proveedores



Fuente: Elaboración Propia

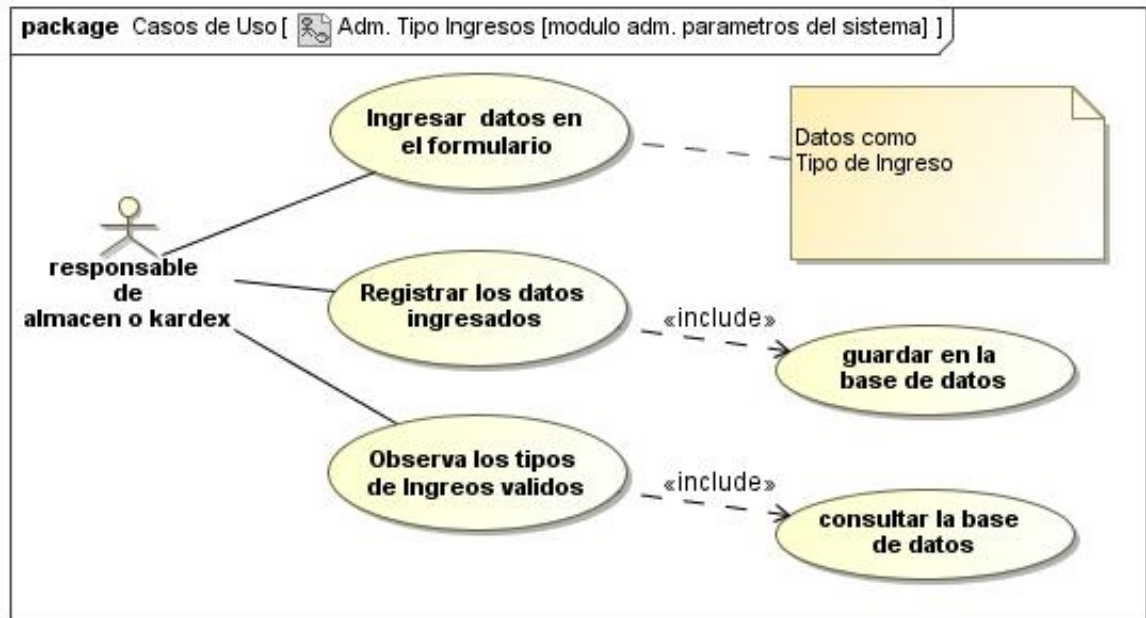
**FIGURA 3.6:** Diagrama casos de uso Administrar Rubros



Fuente: Elaboración Propia

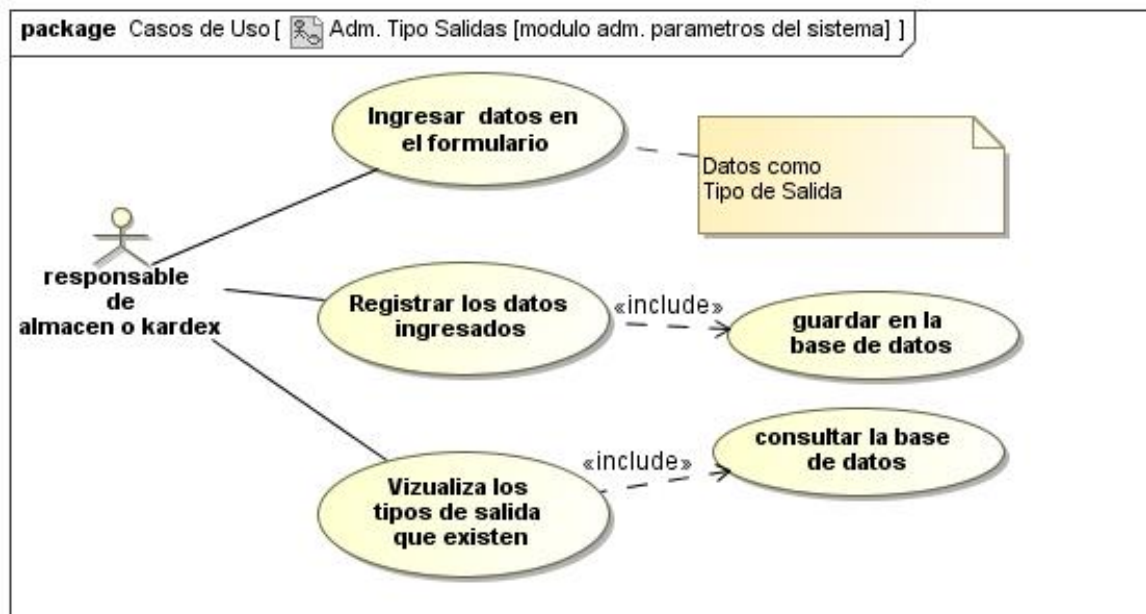
### CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.7:** Diagrama de casos de uso Administrar Tipos de Ingreso



Fuente: Elaboración Propia

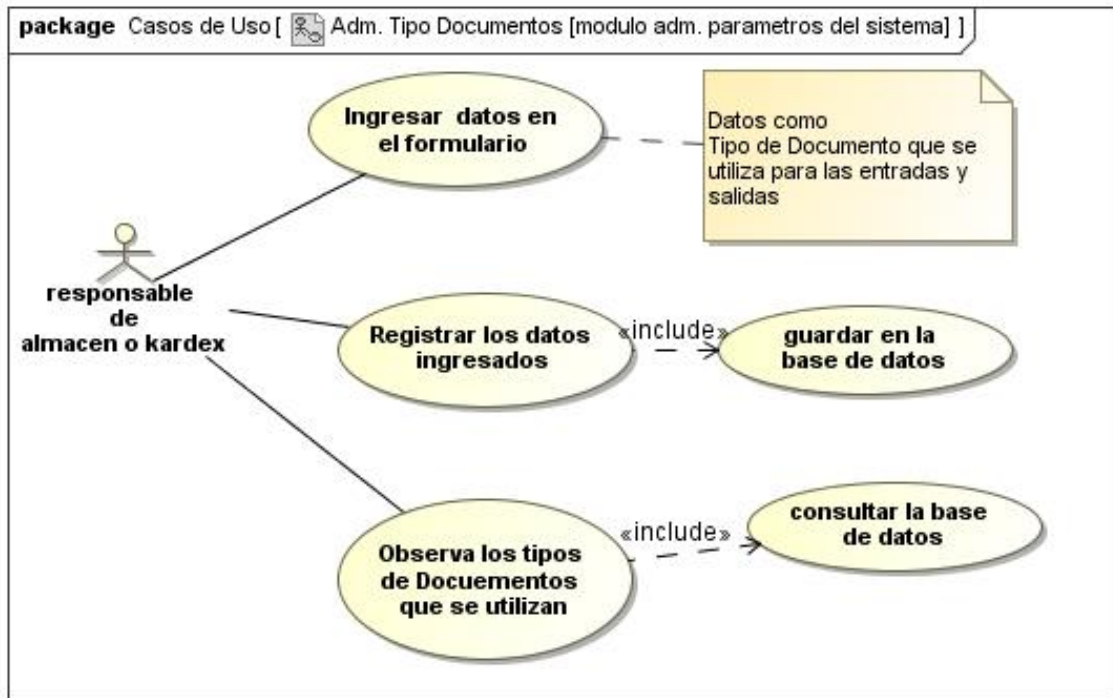
**FIGURA 3.8:** Diagrama de casos de uso Administrar Tipos de Salida



Fuente: Elaboración Propia

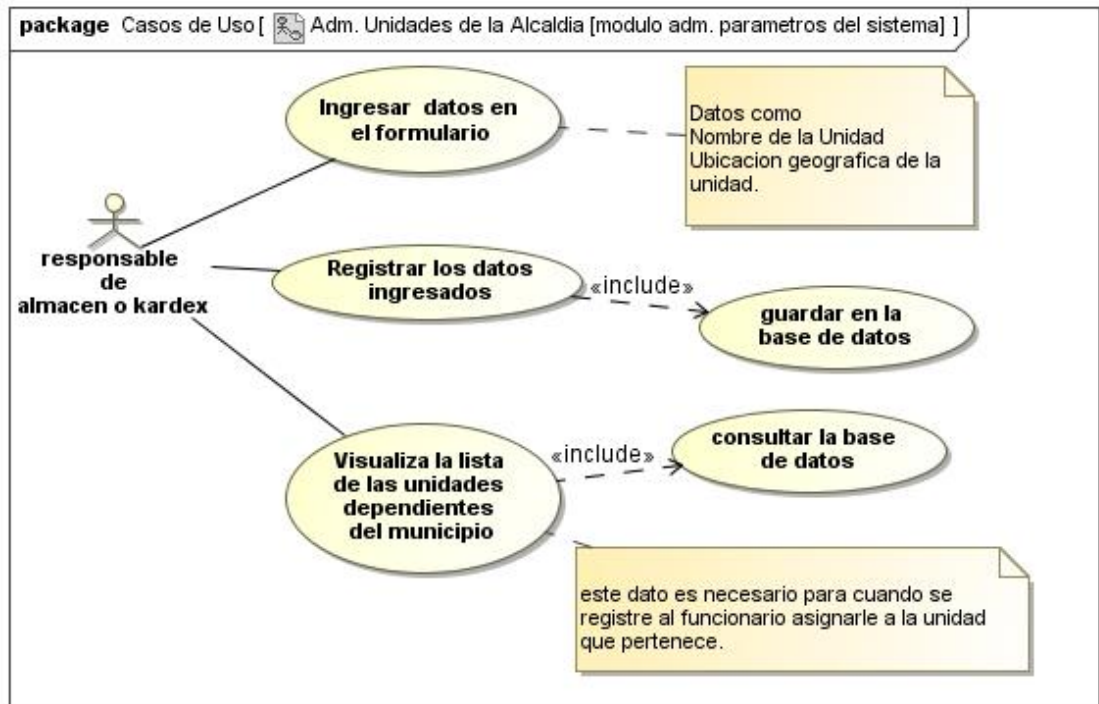
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.9:** Diagrama de casos de uso Administrar Tipo de Documentos



Fuente: Elaboración Propia

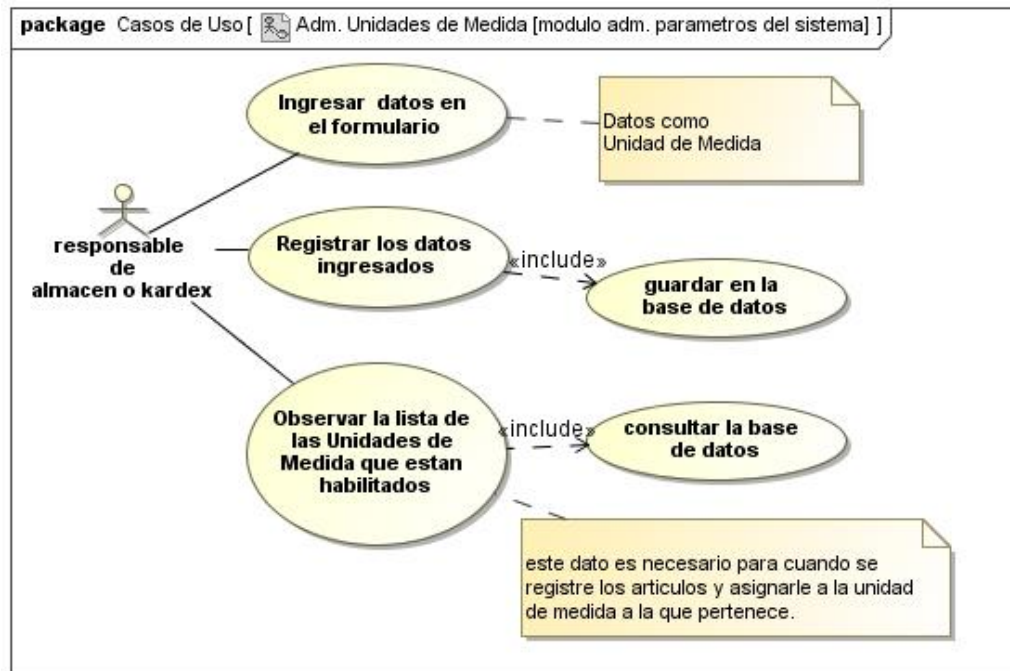
**FIGURA 3.10:** Diagrama de casos de uso Administrar Unidades de la Alcaldía



Fuente: Elaboración Propia

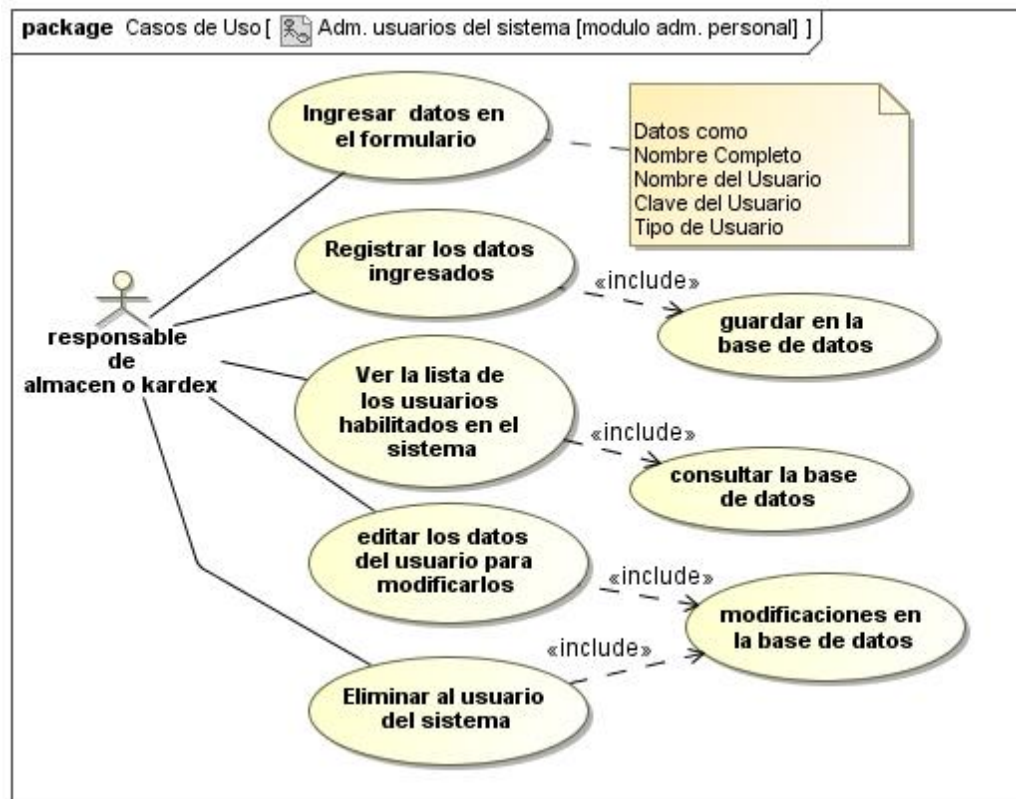
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.11:** Diagrama de casos de uso Administrar Unidades de Medida



Fuente: Elaboración Propia

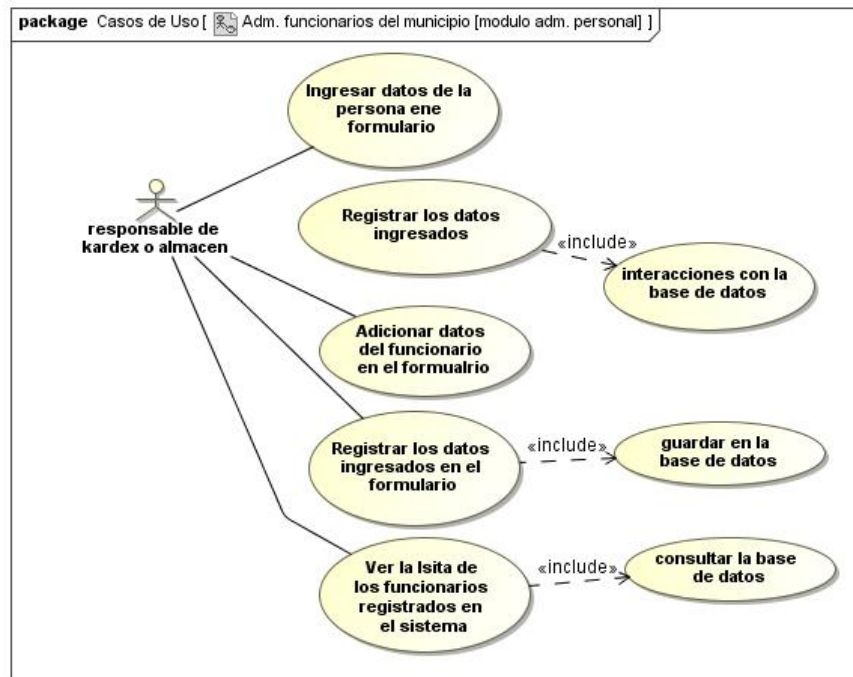
**FIGURA 3.12:** Diagrama de casos de uso Administrar Usuarios del Sistema



Fuente: Elaboración Propia

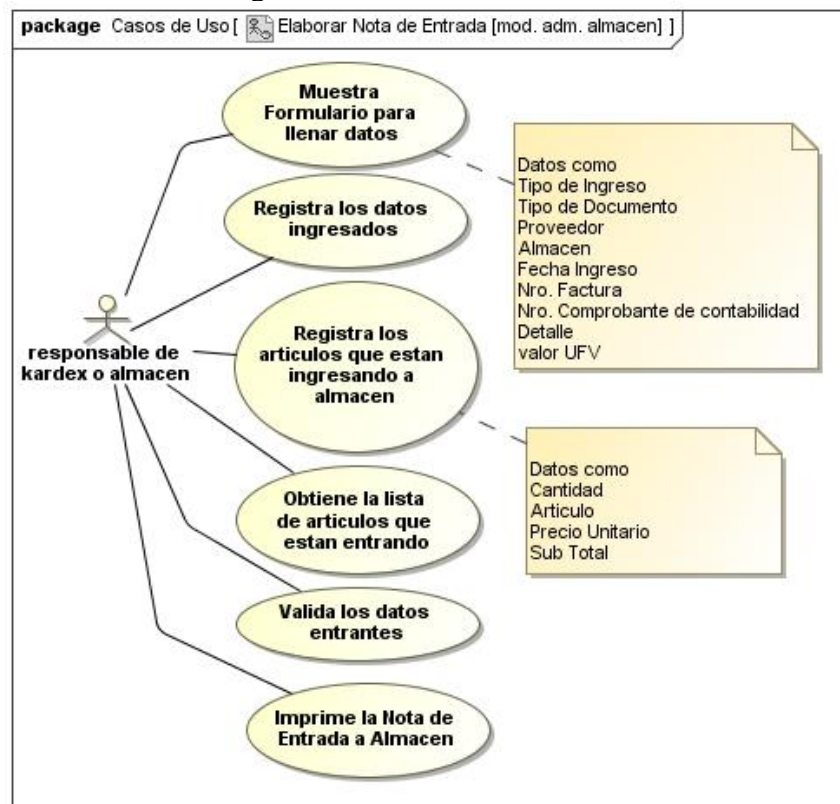
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.13:** Diagrama de casos de uso Administrar Funcionarios del Municipio



Fuente: Elaboración Propia

**FIGURA 3.14:** Diagrama de casos de uso Elaborar Nota de Entrada



Fuente: Elaboración Propia

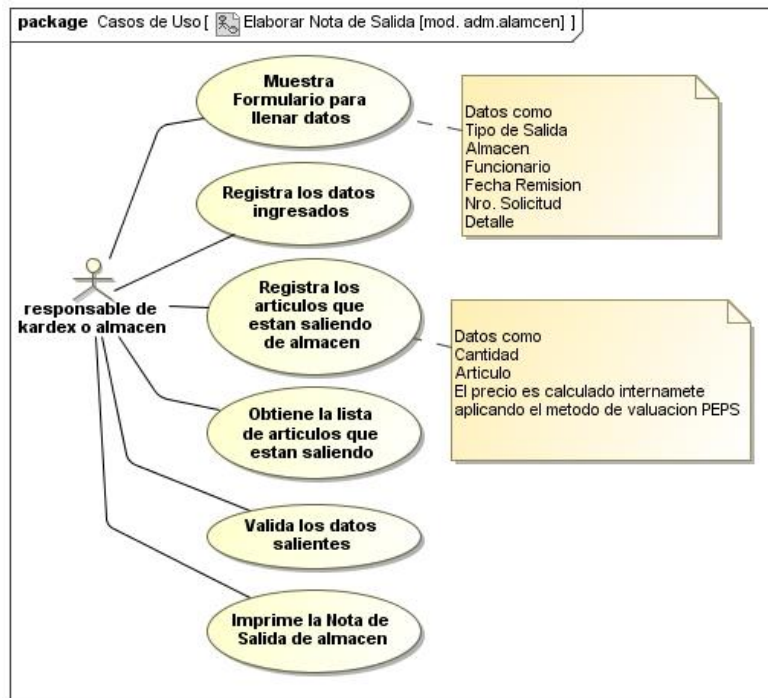
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**TABLA 3.8:** Descripción del caso de uso Elaborar Nota de Entrada

Caso de uso	Descripción
Muestra el formulario para llenar los datos.	Se introduce los datos necesarios para elaborar la nota de entrada. Los datos que se llenan son: tipo ingreso, tipo documento, proveedor, almacén, fecha ingreso, valor de la UFV de compra y otros.
Registra los datos ingresados	Almacena los datos ingresados en el formulario
Registra los artículos que van a ingresar.	Se ingresa la cantidad, artículo, precio unitario de los artículos que ingresaran a almacén, mientras sigue existiendo mas artículos para ingresar se sigue con los mismos pasos.
Obtiene la lista de artículos que están ingresando.	Mientras se ingresa los datos del artículo en la parte inferior se observa la lista de los artículos ingresados.
Valida los datos	Cuando ya se termina de ingresar todos los artículos que ingresar a almacén se procede a la validación de de estos datos.
Imprime la nota de entrada	Después de validar la nota de entrada se imprime como constancia de que ingreso a almacén.

Fuente: Elaboración Propia

**FIGURA 3.15:** Diagrama caso de uso elaborar Nota de Salida



Fuente: Elaboración Propia

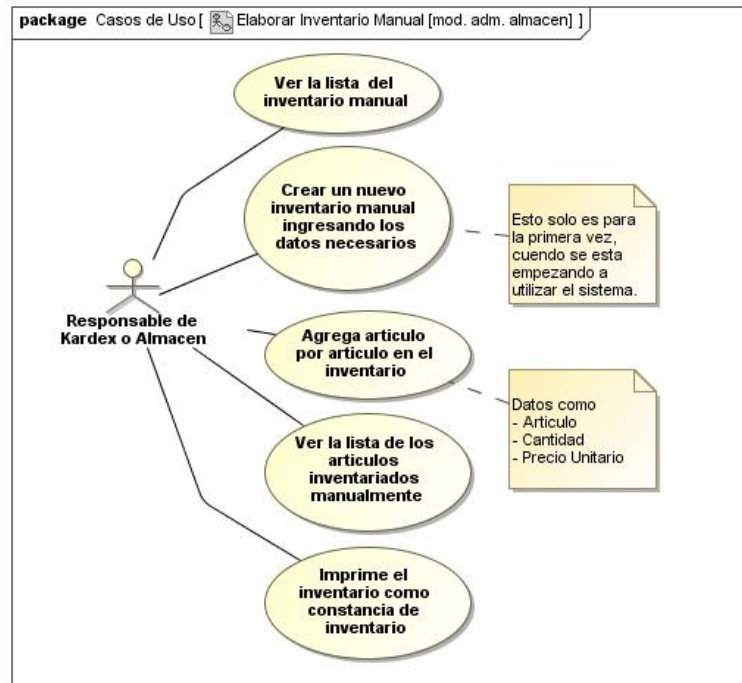
**CAPITULO III  
DESARROLLO DEL PROYECTO**

**TABLA 3.9:** Descripción del caso de uso elaborar Nota de Salida

Caso de uso	Descripción
Muestra el formulario para llenar los datos	Se llena los datos necesarios para elaborar la nota de salida los datos que se llena son: tipo de salida, almacén, funcionario, fecha remisión, nro. de solicitud y detalle de entrega.
Registra los datos ingresados	Almacena los datos ingresados
Registra los artículos que están saliendo de almacén.	Para despachar los artículos se ingresa la cantidad y el articulo y el sistema automáticamente va actualizando la existencia y generando el kardex del articulo.
Obtiene la lista de artículos que están saliendo.	Mientras se ingresa los datos del artículo en la parte inferior se observa la lista de los artículos que están saliendo.
Valida los datos	Cuando ya se termina de ingresar todos los artículos que salen de almacén se procede a la validación de estos datos.
Imprime la nota de salida	Después de validar la nota de salida se imprime como constancia de que salió tales artículos.

Fuente: Elaboración Propia

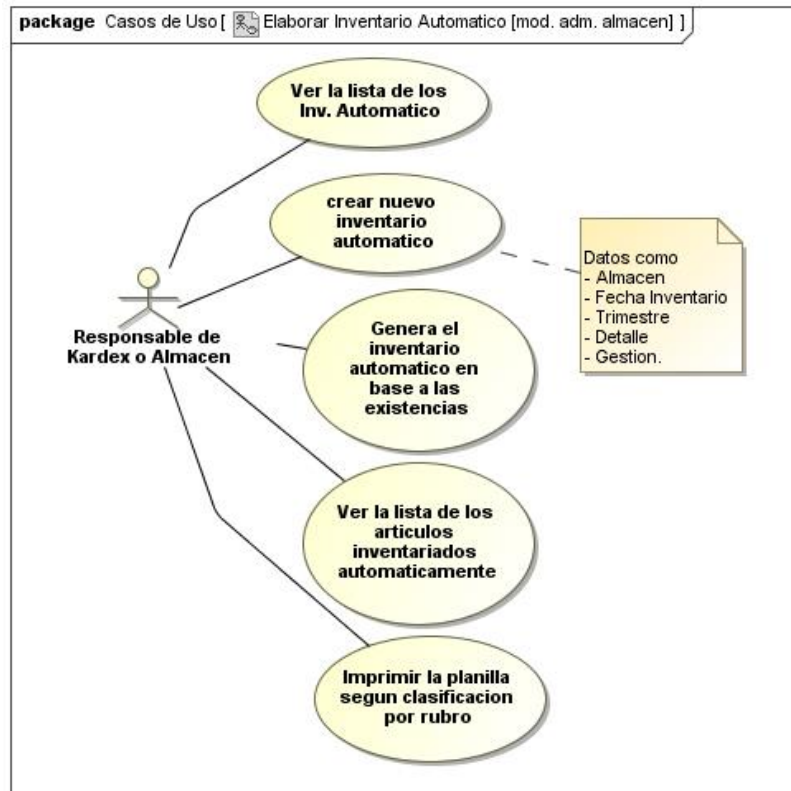
**FIGURA 3.16:** Diagrama de casos de uso Inventario Manual



Fuente: Elaboración Propia

**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.17:** Diagrama de casos de uso Inventario Automático



Fuente: Elaboración Propia

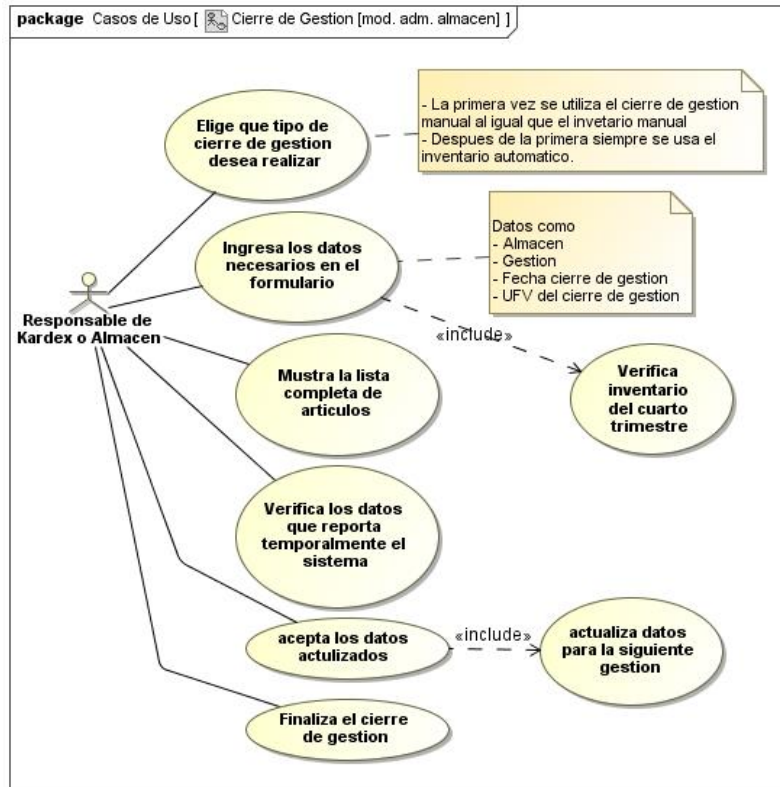
**TABLA 3.10:** Descripción del caso de uso Inventario Automático

<b>Caso de uso</b>	<b>Descripción</b>
Ver lista de los Inventarios	El responsable de kardex ve la lista de los inventarios generados anteriormente.
Crea nuevo inventario automático.	Para generar otro inventario se debe crear antes un inventario con datos generales.
Generar el inventario automático en base a la existencia.	Una vez creado el inventario se procede a generar el inventario de cada uno de los artículos en base a la existencia actuales del artículo.
Ver la lista de los artículos inventariados automáticamente.	Después de haber generado el inventario se muestra en la interfaz el resultado del inventario agrupado por rubros.
Imprime la planilla según su clasificación	El inventario reportado se puede imprimir como constancia de la cantidad en existencia.

Fuente: Elaboración Propia

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

**FIGURA 3.18:** Diagrama de casos de uso Cierre de Gestión



Fuente: Elaboración Propia

**TABLA 3.11:** Descripción del caso de uso Cierre de Gestión

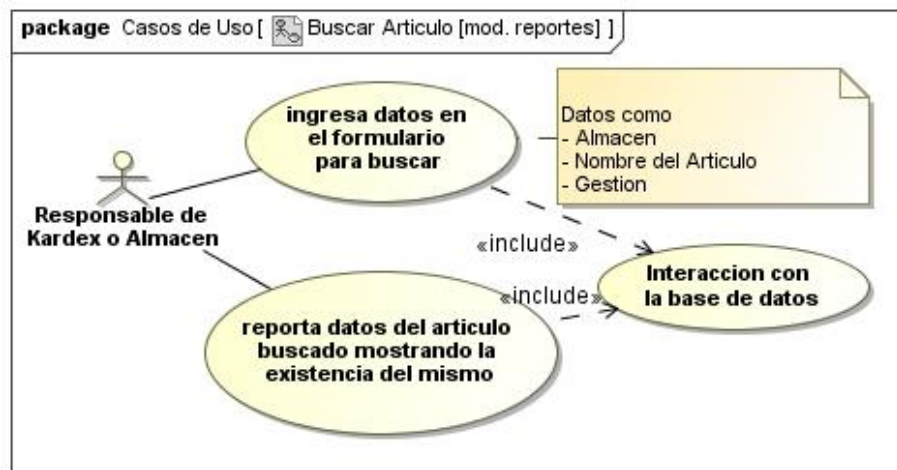
Caso de uso	Descripción
Elige que tipo de cierre de gestión desea realizar	El usuario elige el tipo de cierre de gestión que desea realizar puede ser manual o automático, se aplica manual la primera vez, después solamente se utilizara el automático.
Ingresa los datos necesarios para el cierre de gestión.	En el formulario de del cierre de gestión se ingresa los siguientes datos: almacén, gestión al que corresponde, fecha del cierre de gestión y la UVF del cierre de gestión.
Observa la lista de todos los artículos actualizados según la UFV.	Después de aceptar el sistema muestra todos los artículos con sus saldos actualizados según la UVF del 31 de diciembre.
Verifica los datos reportados por el sistema.	El sistema muestra los datos para que el responsable verifique los datos reportados.

**CAPITULO III  
DESARROLLO DEL PROYECTO**

Acepta los datos actualizados	Después de afirmar los datos reportados, el usuario acepta estos datos y se actualiza las existencias para la siguiente gestión.
Finaliza el cierre de gestión.	Finaliza el cierre de gestión de todos los artículos.

Fuente: Elaboración Propia

**FIGURA 3.19:** Diagrama de casos de uso Buscar Artículo



Fuente: Elaboración Propia

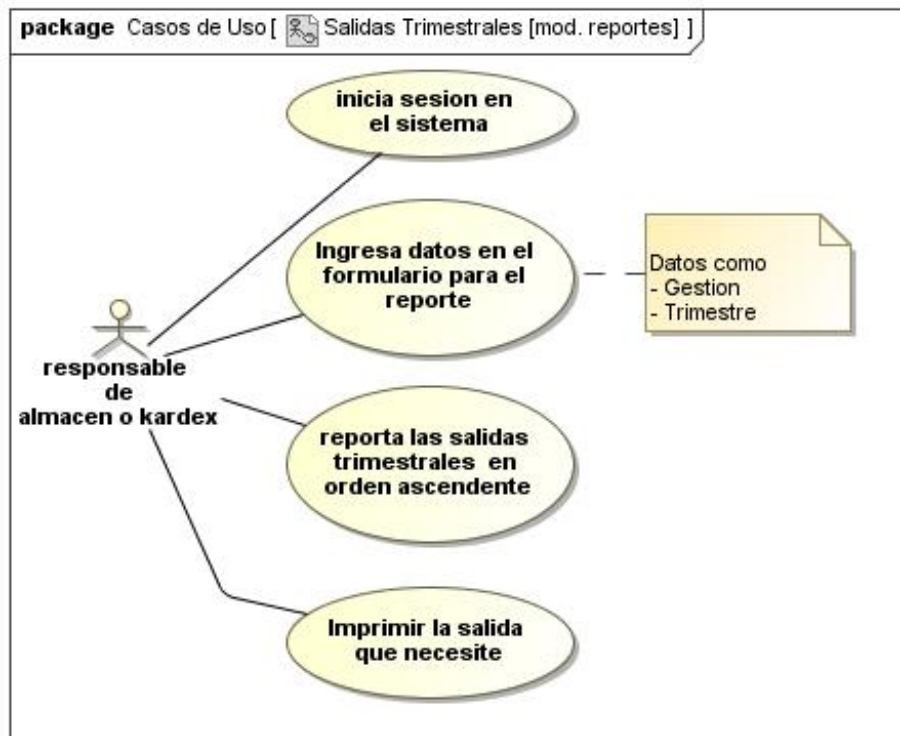
**FIGURA 3.20:** Diagrama de casos de uso Ver Existencias



Fuente: Elaboración Propia

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

**FIGURA 3.21:** Diagrama de casos de uso Salidas Trimestrales



Fuente: Elaboración Propia

**FIGURA 3.22:** Diagrama de casos de uso Entradas Trimestrales



Fuente: Elaboración Propia

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

**FIGURA 3.23:** Diagrama de casos de uso Kardex por Artículo



Fuente: Elaboración Propia

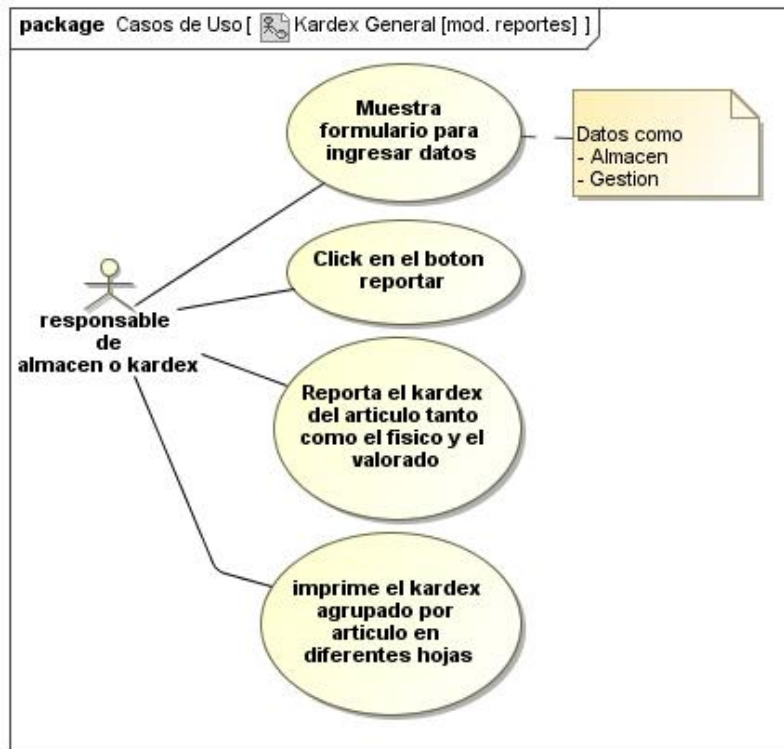
**TABLA 3.12:** Descripción de caso de uso Kardex por Artículo

<b>Caso de uso</b>	<b>Descripción</b>
Ingresa los datos para reportar el kardex del artículo.	El responsable para reportar el kardex de un artículo en específico ingresa los siguientes datos: almacén, nombre artículo, gestión.
Reporta el kardex del artículo ingresado.	Al hacer click en el botón reportar el sistema muestra en la parte inferior del formulario.
Imprime el kardex del articulo	El reporte también se puede imprimir para una constancia.

Fuente: Elaboración Propia

**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.24:** Diagrama de casos de uso Kardex General



Fuente: Elaboración Propia

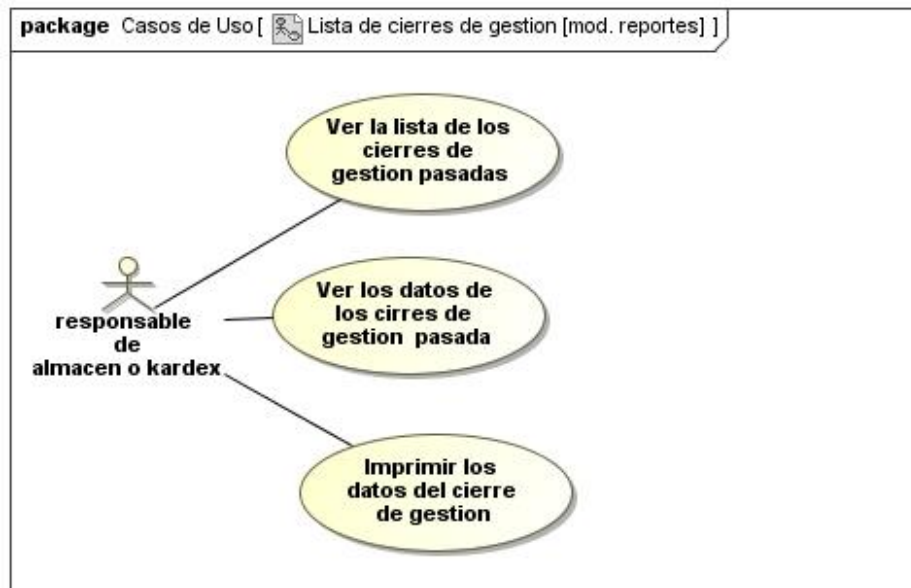
**TABLA 3.13:** Descripción del caso de uso Kardex General

<b>Caso de uso</b>	<b>Descripción</b>
Ingresa los datos para reportar el Kardex general.	Para poder reportar el Kardex en el formulario. Se debe ingresar el almacén y la gestión del que quiere reportar.
Ejecutar en el botón reportar	Al presionar sobre el botón, el sistema empieza a actuar.
Reporte del kardex en general.	El sistema reporta el Kardex de todos los artículos.
Imprime el kardex del articulo	Al finalizar el reporte este mismo se puede imprimir como una constancia.

Fuente: Elaboración Propia

## CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.25:** Diagrama casos de uso cierres de gestión



Fuente: Elaboración Propia

Cada uno de los casos de uso mostrados anteriormente establecen las funcionalidades del Sistema de Almacén, esto ayudará a seguir con la siguiente fase donde se desarrolla la metodología.

### 3.2. FASE DE ANALISIS

Como se estableció en la Metodología, en la fase de análisis se construye los tres modelos OMT para describir cómo funciona el Sistema de Administración de Inventarios, esta fase se divide en tres partes los cuales son:

- ✓ Modelo de Objetos
- ✓ Modelo Dinámico
- ✓ Modelo Funcional

#### 3.2.1. Modelo de Objetos

Los pasos que se realizaron para obtener el modelo de objetos fueron: la identificación de clases, elaboración de un diccionario de datos y la asociación de clases.

### CAPITULO III DESARROLLO DEL PROYECTO

---

#### *i. Identificación de Clases*

Las clases que se identificaron fueron las siguientes.

**TABLA 3.14:** Clases que se Identificaron para el Sistema

✓ Cálculos	✓ Persona
✓ Artículo	✓ Usuario
✓ Rubros	✓ Funcionario
✓ UnidadesMedida	✓ Unidades
✓ EntradasSalidas	✓ Proveedor
✓ Kardex	✓ Existencia
✓ Inventario	✓ CierreGestion

Fuente: Elaboración Propia

#### *ii. Diccionario de datos*

Los datos que se incorporaron en el diccionario son los siguientes.

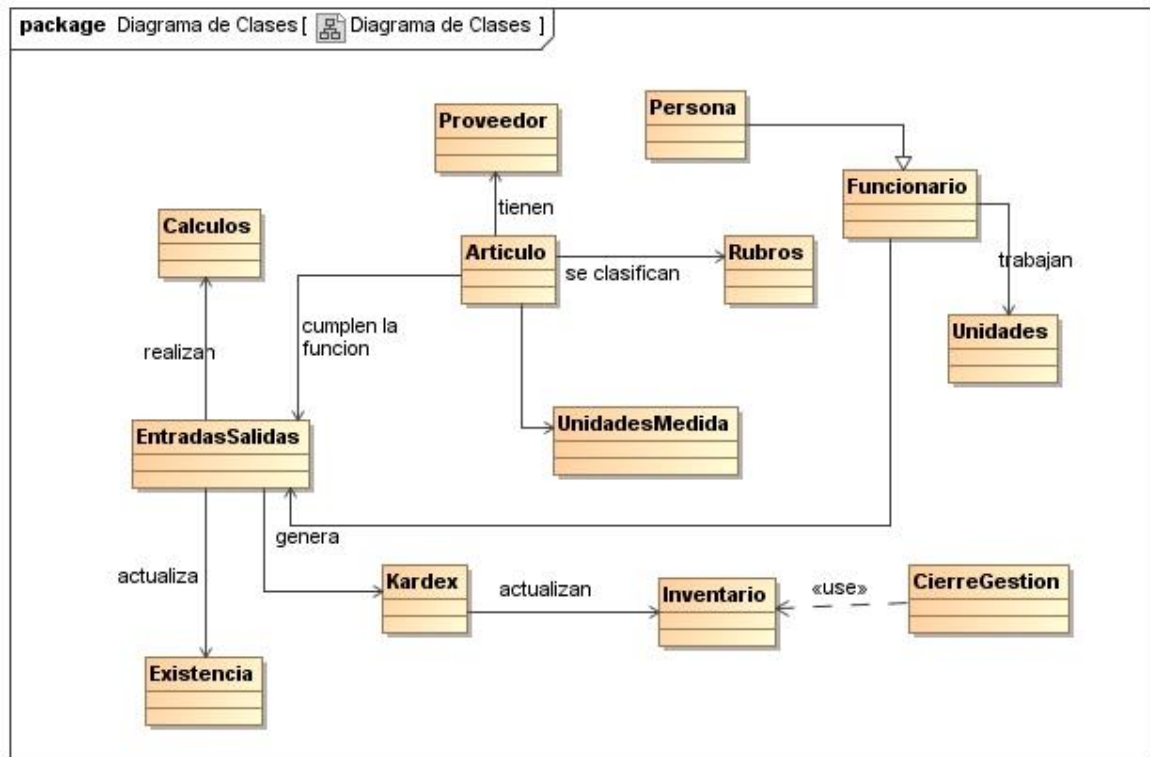
- ✓ **Responsable de almacén.-** Persona encargada de administrar los bienes de la institución, directa responsable de elaborar informes de entrada y salida.
- ✓ **Responsable de kardex.-** Persona encargada de realizar el kardex de cada uno de los artículos activos en la Unidad de Almacén.
- ✓ **Responsable de Entradas y Salidas.-** Persona encargada de realizar la entrega y la recepción física de los artículos que están dentro de almacén.
- ✓ **Rubro.-** Es la clasificación y la forma de organizar los artículos ya que estos son muchos.
- ✓ **Artículo.-** Es un bien que pertenece a la institución el cual es utilizado por el Municipio.
- ✓ **Unidad de Medida.-** La unidad de medida va ligada con los artículos porque cada artículo tiene su propia unidad de medida.
- ✓ **Cierre Gestión.-** Permite capturar los artículos existentes en almacén y actualizar estos para la siguiente gestión según el valor de la UFV de compra y de cierre.
- ✓ **Existencias.-** Es determinar la cantidad existente de concreto articulo en particular o de todos en general.

## CAPITULO III DESARROLLO DEL PROYECTO

### iii. Asociación de Clases

En esta parte se muestra la asociación de clases para su mejor entendimiento se muestra en la siguiente imagen.

**FIGURA 3.26:** Asociación de Clases



Fuente: Elaboración Propia

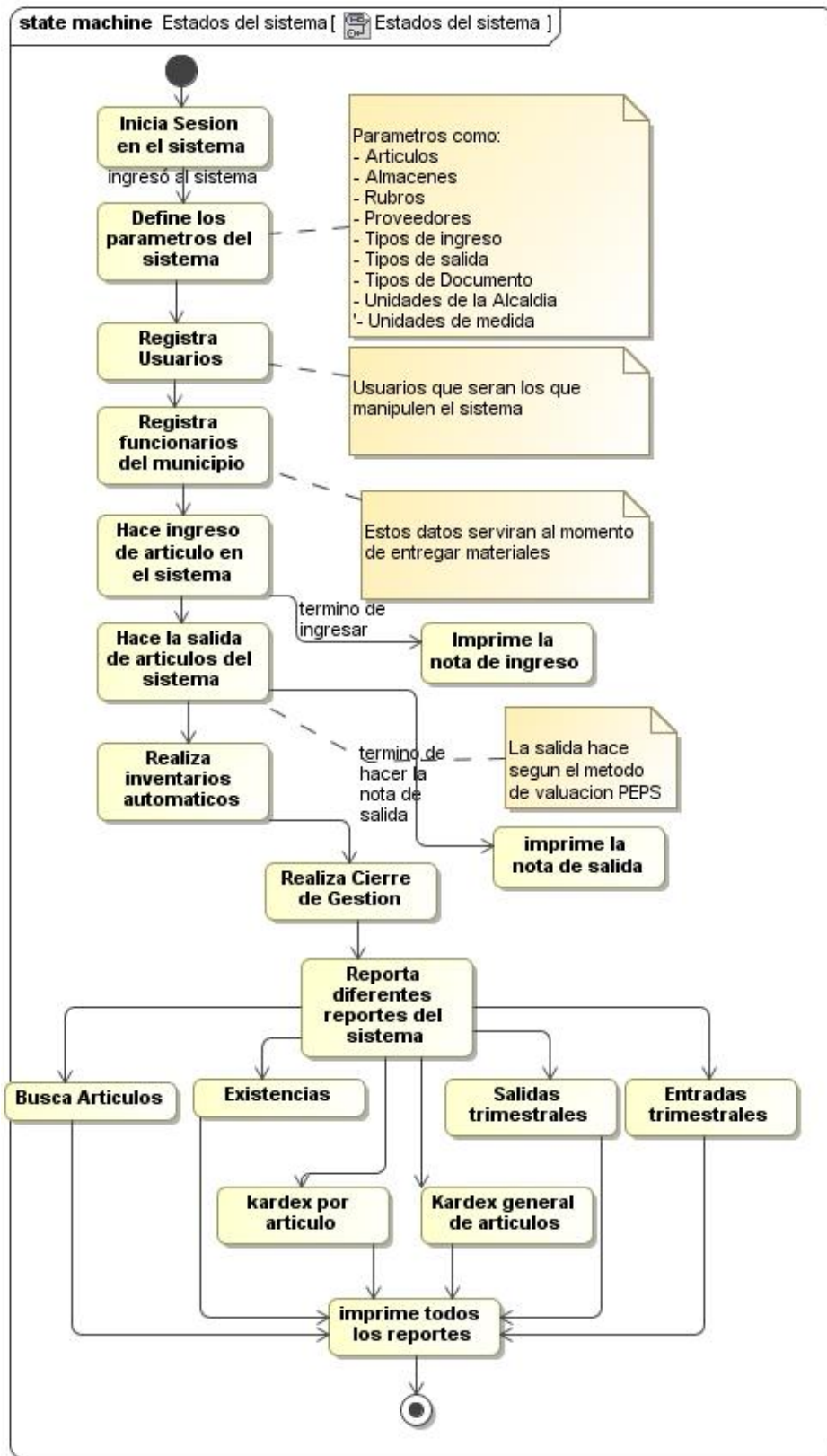
En el diagrama de clases básicamente muestra las relaciones que existen entre ellas.

### 3.2.2. Modelo Dinámico

En esta fase se elaboró el diagrama de estados general del Sistema, para tener un panorama general de los estados que tendrá el sistema así como se observa en la siguiente imagen.

**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.27: Diagrama de Estados del Sistema**



Fuente: Elaboración propia

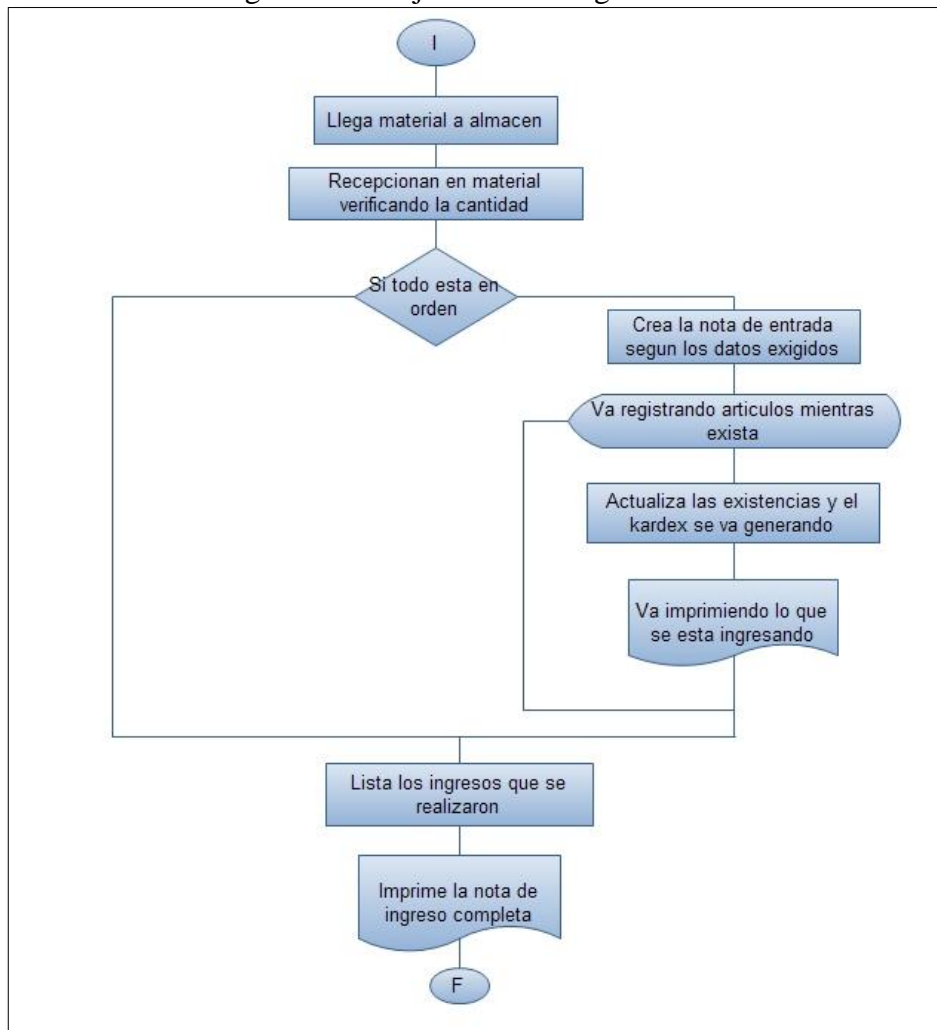
### CAPITULO III DESARROLLO DEL PROYECTO

En la fase de diseño de objetos se muestra a más detalle los diagramas de estado complementando con el diagrama de secuencia ya que estos diagramas permiten mostrar la dinámica del sistema.

#### 3.2.3. Modelo Funcional

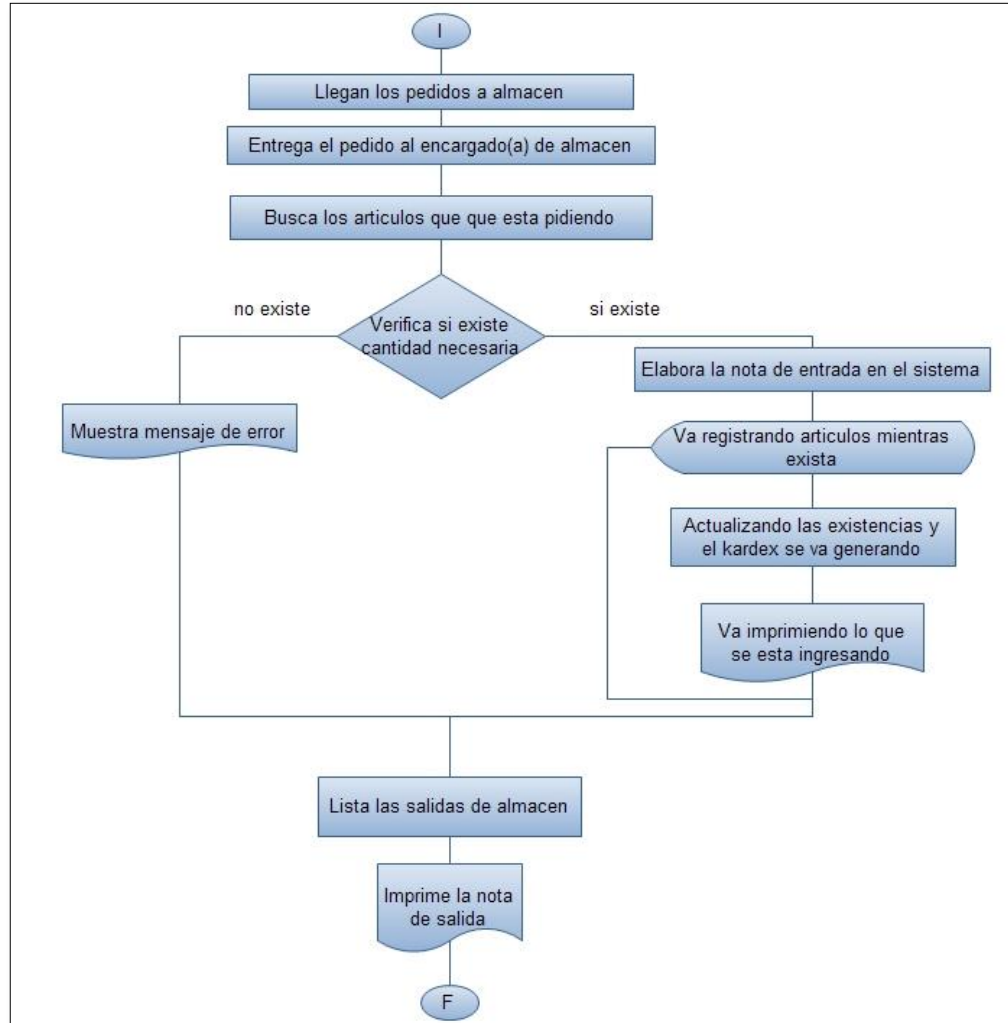
El modelo funcional permite mostrar el flujo de la información, para lo cual se creó el diagrama de flujo general, para mostrar el flujo de la información al momento de realizar los ingresos y salidas de almacenes por medio del sistema así como se muestra en las siguientes figuras.

**FIGURA 3.28:** Diagrama de Flujo de Datos Ingreso de Artículos a Almacén



Fuente: Elaboración propia

**FIGURA 3.29:** Diagrama de Flujo Salida de Artículos de Almacén



Fuente: Elaboración propia

### **3.3. FASE DE DISEÑO DEL SISTEMA**

En la fase de diseño del sistema se llegó a determinar los módulos que serán parte del sistema en su totalidad, también se llegó a definir que arquitectura que se empleará en su funcionalidad.

#### **3.3.1. Módulos del Sistema**

Los módulos que se definieron para el funcionamiento del sistema de información son los siguientes:

### CAPITULO III DESARROLLO DEL PROYECTO

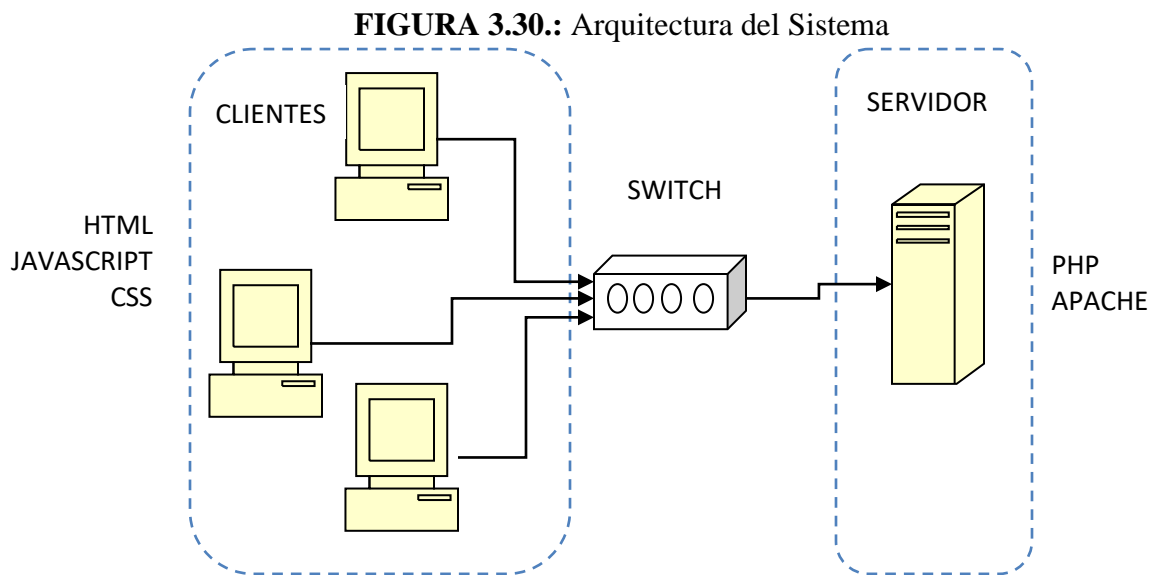
---

- ✓ Modulo Administrar Parámetros del Sistema
- ✓ Modulo Administrar Personas
- ✓ Módulo Administrar Almacén
- ✓ Módulo Reportes

Los módulos mencionados están implementados y en etapa de prueba en la Unidad de Almacén.

#### 3.3.2. Arquitectura básica del sistema

El Sistema tiene una arquitectura funcional cliente/servidor, o conocido de otra forma como Sistema orientado a la web, tal como se observa en la siguiente Figura.



Fuente: Elaboración propia

#### 3.3.3. Procesos o funcionalidades de los módulos

Las funcionalidades que se llegaron a desarrollar en cada uno de los módulos del sistema se muestran en el siguiente Tabla.

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

**TABLA 3.15.: Módulos y Funcionalidades**

MODULOS	FUNCIONALIDADES
Módulo Administrar Parámetros del Sistema.	<ul style="list-style-type: none"> <li>• Artículos</li> <li>• Almacenes</li> <li>• Proveedores</li> <li>• Rubros (Categorías)</li> <li>• Tipos de Entrada y Salida</li> <li>• Tipos de documentos</li> <li>• Unidades de la Alcaldía</li> <li>• Unidades de medida</li> </ul>
Modulo Administrar Personal	<ul style="list-style-type: none"> <li>• Usuarios del Sistema</li> <li>• Personas habilitadas</li> <li>• Lista de funcionarios</li> </ul>
Módulo Administrar Almacén	<ul style="list-style-type: none"> <li>• Nota de Entrada y Salida</li> <li>• Inventario Manual</li> <li>• Inventario Automático</li> <li>• Cierre de Gestión</li> </ul>
Modulo de reportes	<ul style="list-style-type: none"> <li>• Buscar Artículos</li> <li>• Existencias</li> <li>• Salidas trimestrales</li> <li>• Entradas trimestrales</li> <li>• Kardex por cada artículo</li> <li>• Kardex general de artículos</li> <li>• Cierres de gestión</li> </ul>

Fuente: Elaboración propia

**3.3.4. Estructura y diseño de la base de datos**

Para que el sistema guarde toda la información, se diseñó la base de datos con el nombre de Almacén, la misma que cuenta con 22 tablas y está diseñada en el manejador de base de datos MYSQL, la base de datos se encuentra en la tercera forma normal así como se ve en la siguiente Figura.



### CAPITULO III DESARROLLO DEL PROYECTO

---

#### 3.4. FASE DE DISEÑO DE OBJETOS

En la etapa de diseño de objetos así como la metodología indica que se aplica: el modelo de objetos detallado, modelo dinámico detallado, modelo funcional detallado las mismas que se describen en los siguientes apartados.

##### 3.4.1. Modelo de Objetos Detallado

En la primera parte se hace un listado de todas las clases que son parte del sistema las mismas que son creadas bajo el lenguaje de programación PHP.

Las clases del sistema son:

**TABLA 3.16:** Clases del sistema y su descripción

➤ Cálculos	En la clase Cálculos están los métodos que permiten calcular los totales, sub totales y precios actualizados.
➤ Artículo	En la clase Artículo están los métodos que permiten registrar los artículos, listar y otros métodos.
➤ Rubros	En la clase Rubros están los métodos que permiten registrar los rubros, listar, verificar duplicidad y otros.
➤ UnidadesMedida	La clase UnidadesMedida registra las diferentes unidades de medida y lista las unidades.
➤ EntradasSalidas	La clase EntradasSalidas registra las entradas y las salidas que se realizan en el sistema.
➤ Kardex	La clase Kardex registra las entradas y las salidas ordenado por fecha para hacer el respectivo balance.

**CAPITULO III  
DESARROLLO DEL PROYECTO**

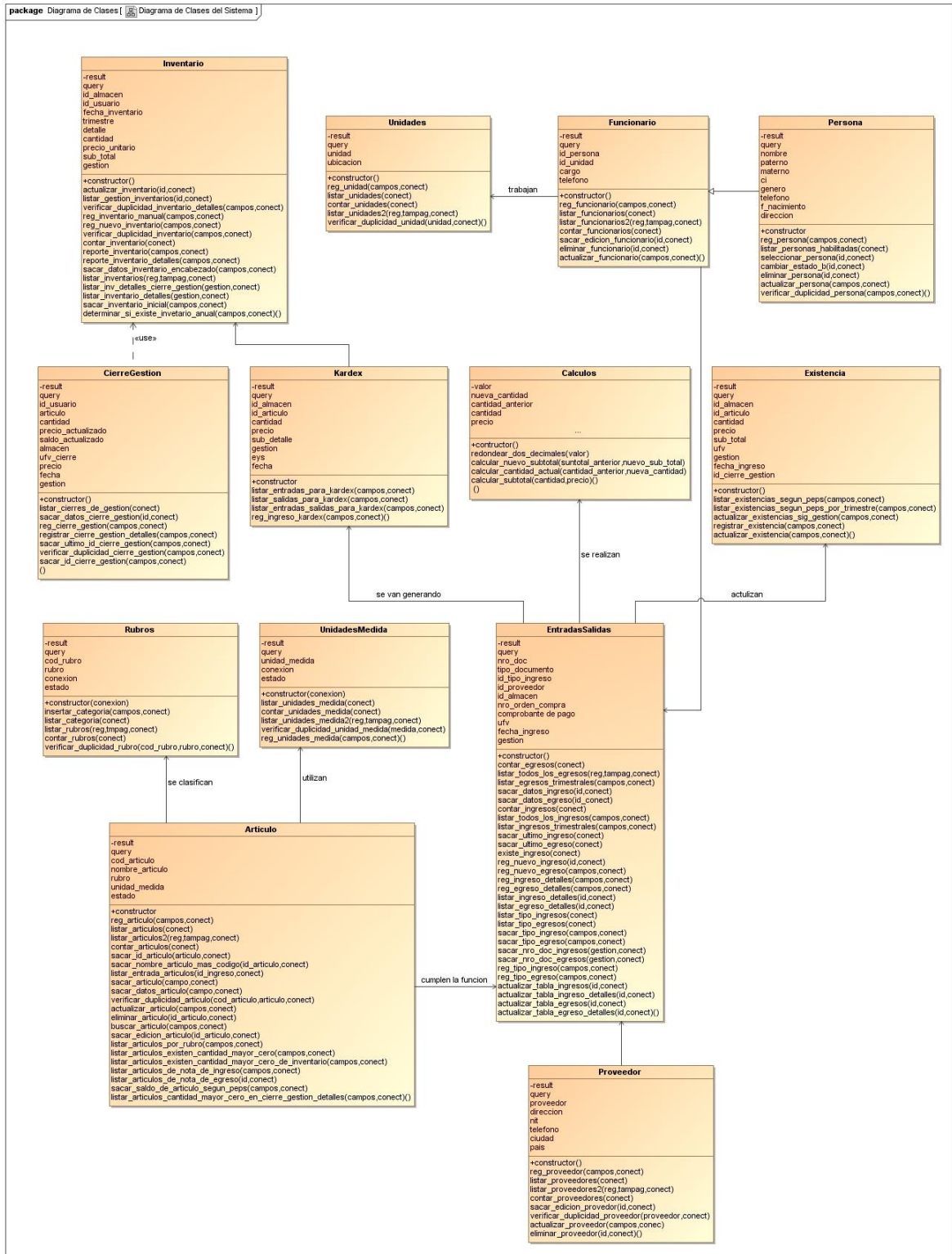
➤ Inventario	La clase Inventario crea nuevos inventarios, también genera los inventarios automáticamente en base a las existencias.
➤ CierreGestion	La clase CierreGestion permite crear nuevos cierres de gestión y cerrar el cierre de gestión de manera automática.
➤ Persona	La clase Persona registra los datos de la persona y también permite listar las personas habilitadas.
➤ Usuario	La clase Usuario permite registrar a los usuarios que serán parte del sistema también permite listar y modificar los datos del usuario.
➤ Funcionario	La clase Funcionario permite registrar a la persona como funcionario y también permite obtener la lista de todos funcionarios habilitados.
➤ Unidades	La clase Unidades registra datos de las unidades de la alcaldía.
➤ Proveedor	La clase Proveedor registra a los proveedores con sus datos respectivos.
➤ Existencia	La clase Existencia muestra una lista completa de todos los artículos más la cantidad existente.
➤ DBManager	La clase DBManager es el que se conecta a la base de datos enviando los parámetros como ser el usuario y la clave.

Fuente: Elaboración Propia

Después de hacer el detalle de las clases en la siguiente figura se muestra el diagrama de clases más las relaciones que existen entre ellas.

# CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.32.: Diagrama de clases



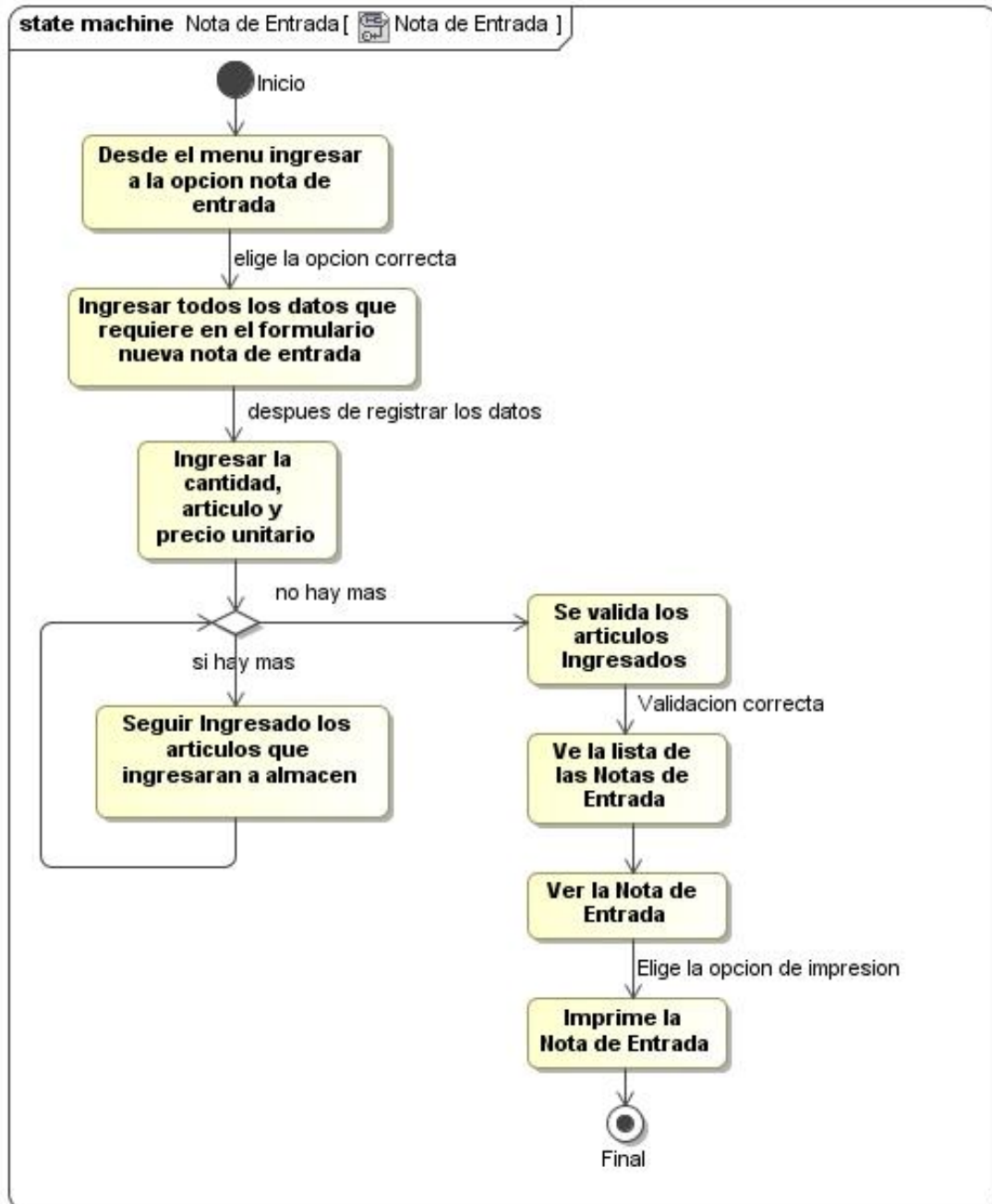
Fuente: Elaboracion propia

### CAPITULO III DESARROLLO DEL PROYECTO

#### 3.4.2. Modelo Dinámico detallado

Los siguientes diagramas de estado muestran los estados del sistema, esta parte los casos más sobresalientes del Sistema.

FIGURA 3.33.: Diagrama de Estado del Caso de Uso Nota de Entrada

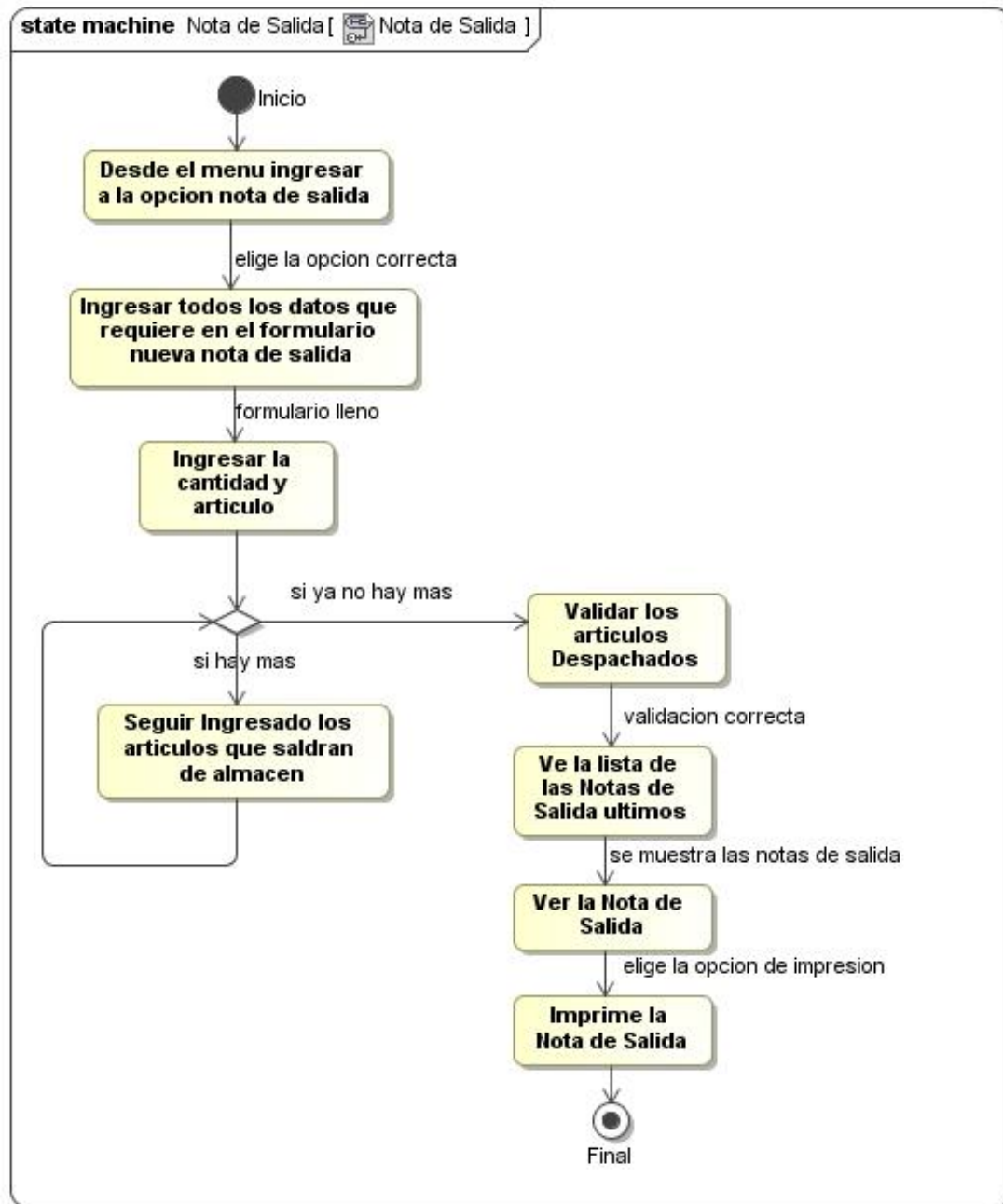


Fuente: Elaboracion propia

### CAPITULO III DESARROLLO DEL PROYECTO

En el siguiente diagrama se observa los estados del sistema cuando se realiza la nota de salida.

**FIGURA 3.34.:** Diagrama de Estado del caso de uso Nota de Salida

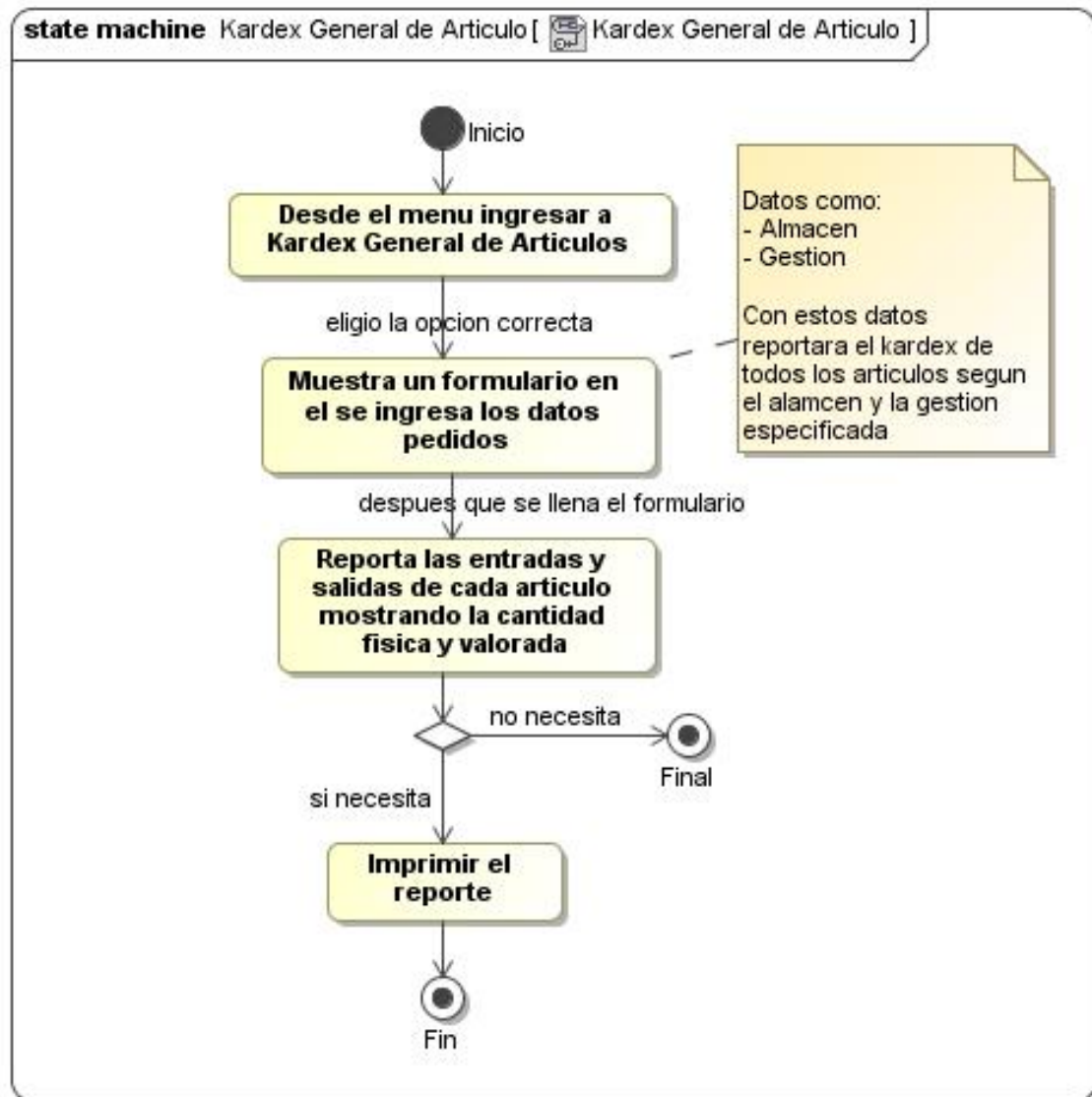


Fuente: Elaboracion propia

En la siguiente figura se muestra el diagrama de estado Kardex General, describe los estados por el que pasa para reportar el kardex de todos los articulos que estan activos dentro de Almacén.

### CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.35.: Diagrama de Estado del caso de uso Kardex General

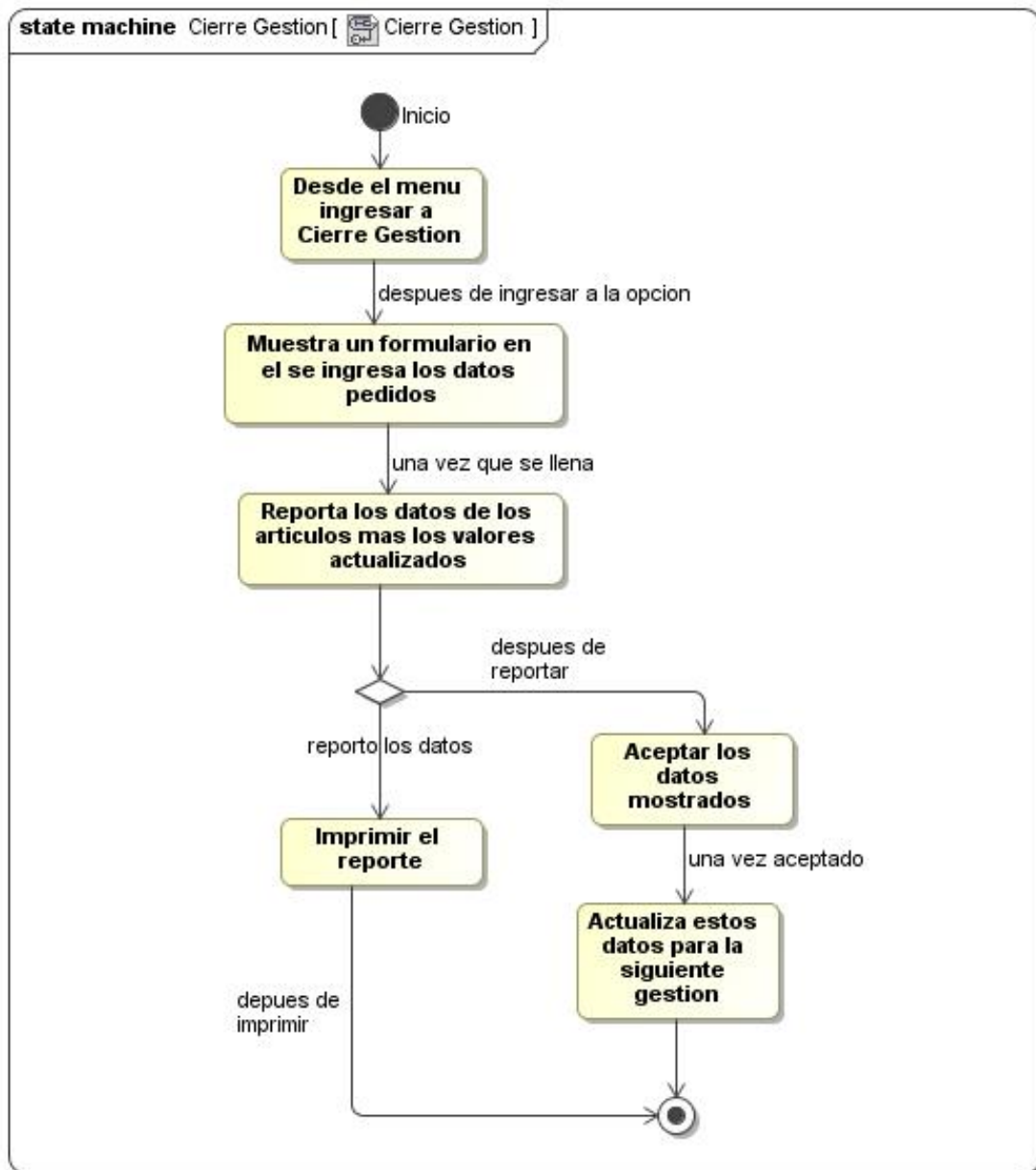


Fuente: Elaboracion propia

El diagrama de estado cierre de gestion describe todos los estados posibles por el cual pasa el sistema para archivar los datos y al mismo tiempo actualizar para la siguiente gestion.

### CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.36.: Diagrama de Estado del caso de uso Cierre Gestion



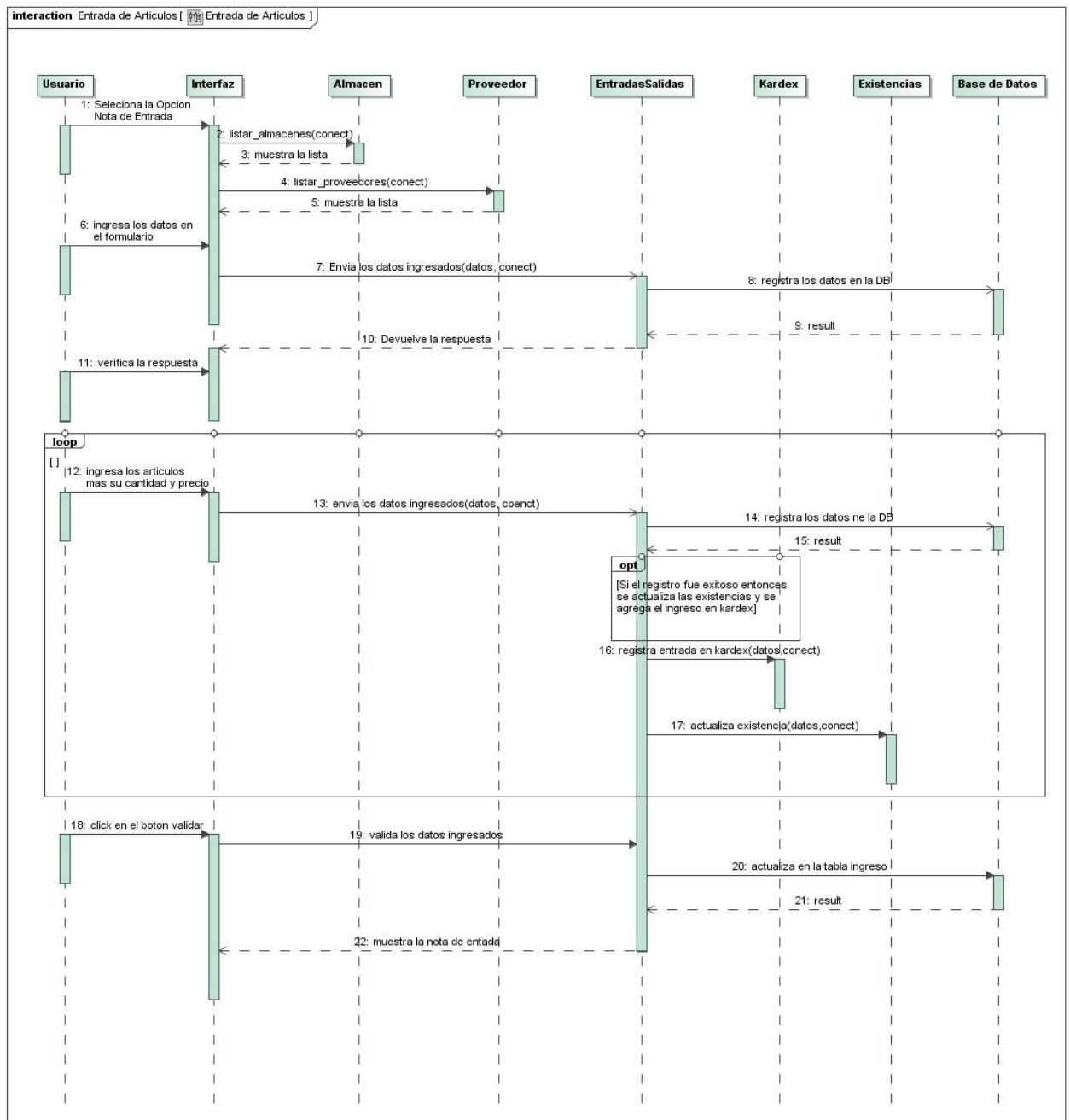
Fuente: Elaboración propia

En esta fase también se vio la necesidad de elaborar los diagramas de secuencia para mostrar a más detalle la dinámica del sistema, donde se observa los mensajes que se pasan entre los objetos para obtener un resultado.

En las siguientes Figuras se muestran los diagramas de secuencia de los casos más particulares del Sistema.

# CAPITULO III DESARROLLO DEL PROYECTO

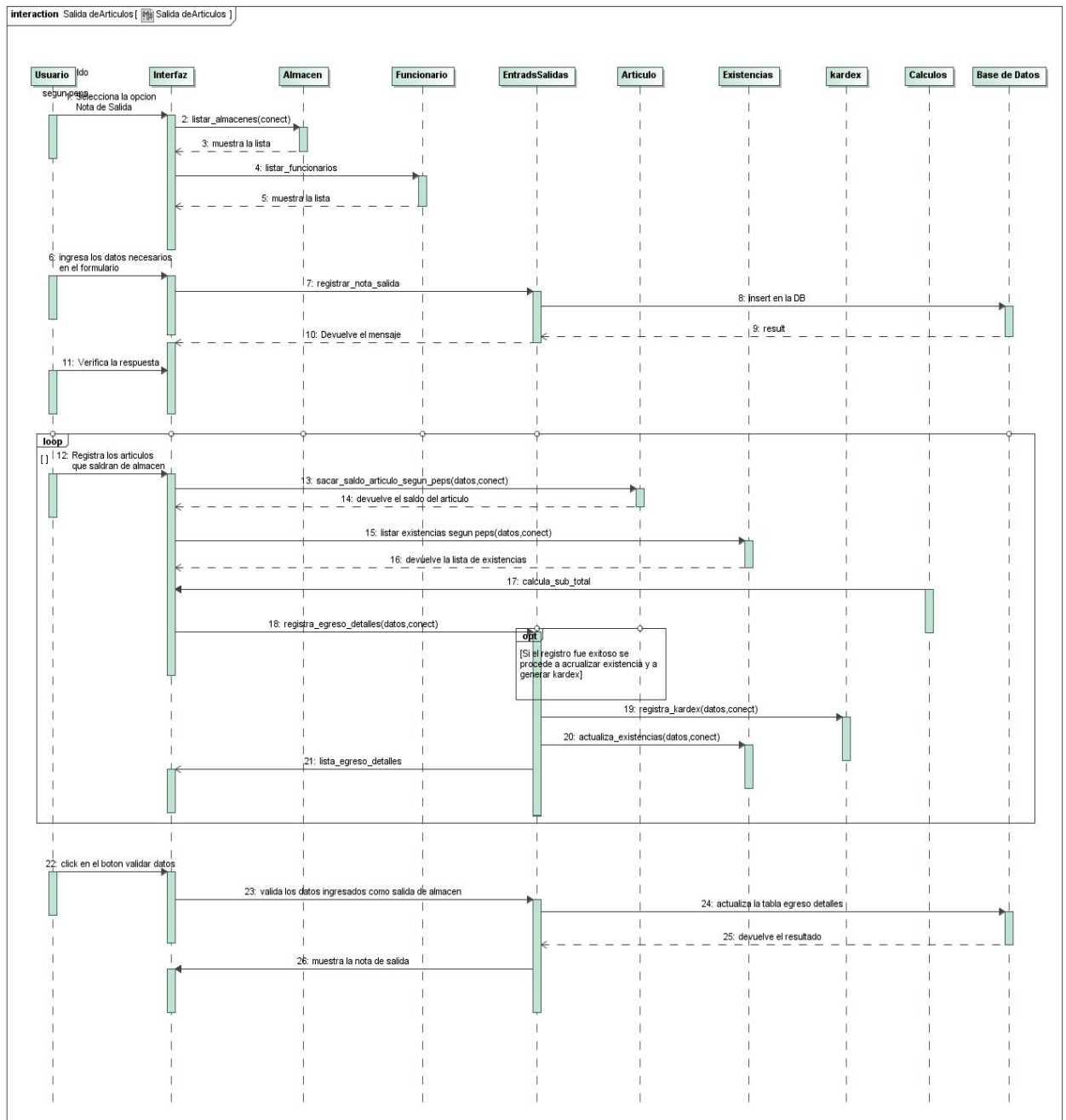
FIGURA 3.37.: Diagrama de Secuencia del caso de uso Entrada de Articulos



Fuente: Elaboración propia

# CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.38.: Diagrama de Secuencia Salida de Articulos a Almacen

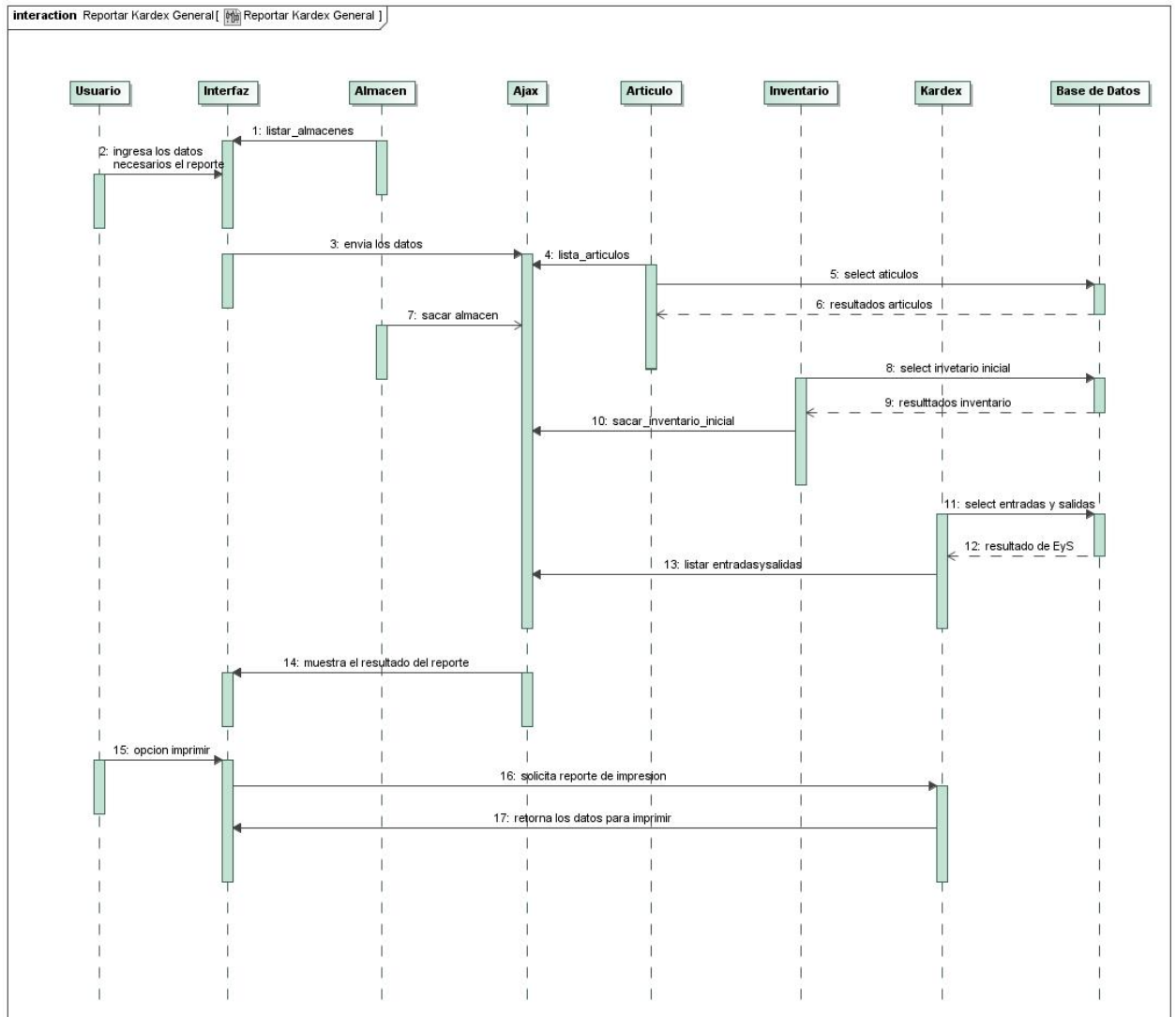


Fuente: Elaboración propia

### CAPITULO III DESARROLLO DEL PROYECTO

En la siguiente Figura se muestra el diagrama de secuencia la misma que muestra en reportar Kardex, la interaccion entre objetos enviandose mensajes.

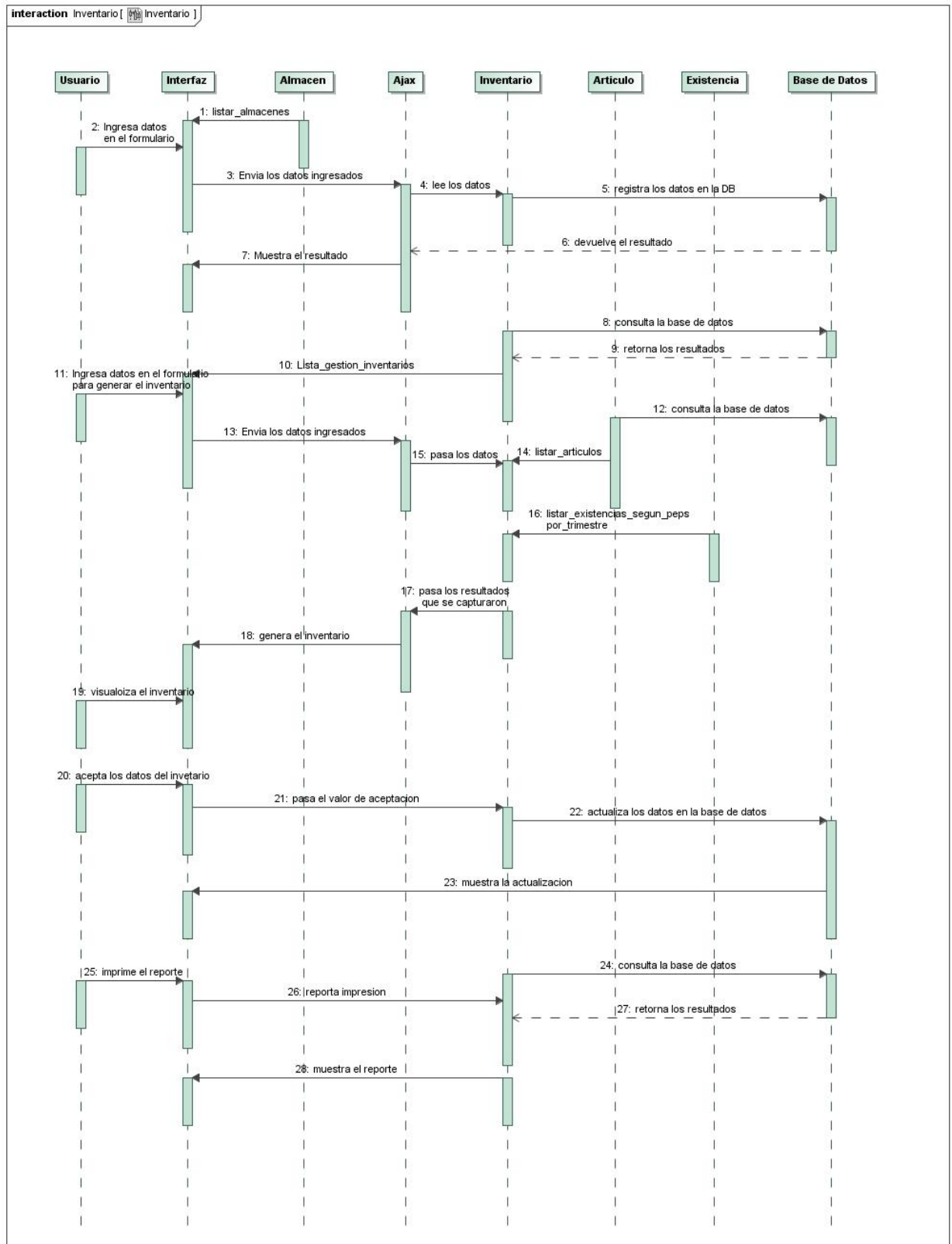
**FIGURA 3.39.:** Diagrama de Secuencia del caso de uso Reportar Kardex



Fuente: Elaboración propia

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

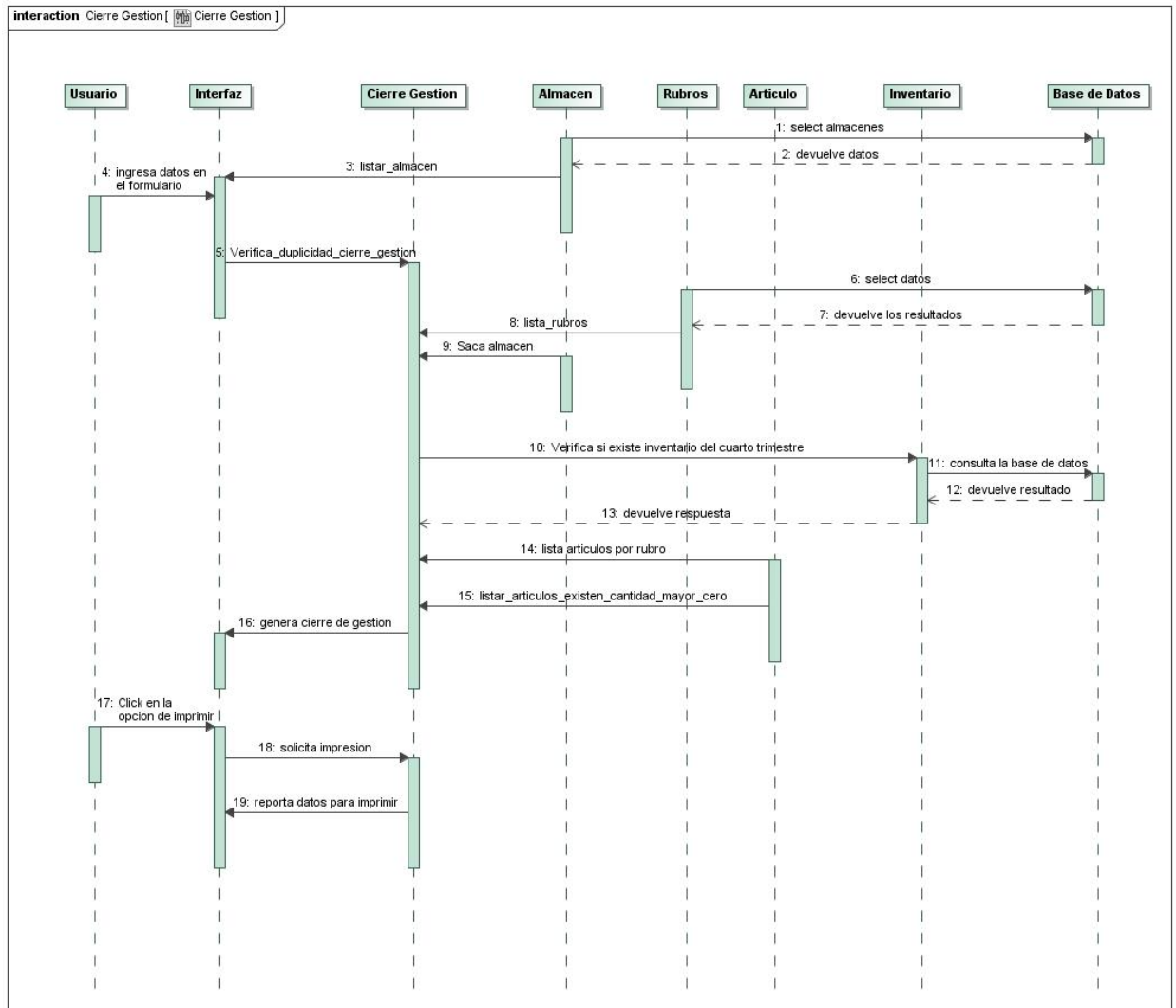
**FIGURA 3.40.: Diagrama de Secuencia Generar Inventario**



Fuente: Elaboración propia

# CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.41.: Diagrama de Secuencia del caso de uso Cierre Gestion



Fuente: Elaboración propia

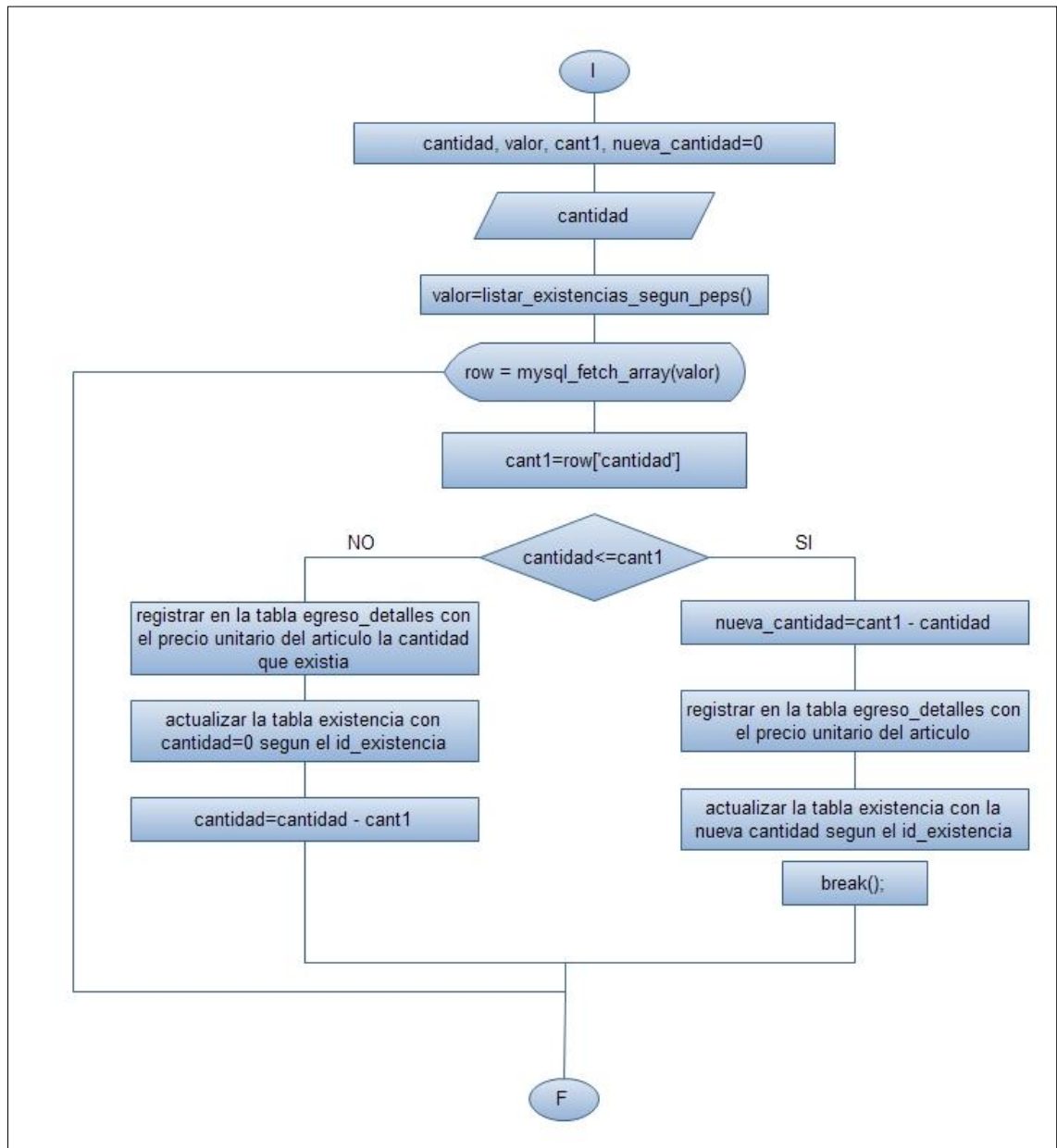
De esta manera se elaboró los diagramas de secuencia que muestra la dinámica de los objetos dentro del sistema en este caso solo se elaboró de los casos más sobresalientes del sistema.

### 3.4.3. Modelo Funcional Detallado

En el modelo funcional se llega a describir el flujo de la información, para esto se elaboró los diagramas de flujos de datos (DFD), para las funcionalidades más sobresalientes que se detectaron y a los que se dedico más tiempo en el análisis y desarrollo del sistema. Tal como se observa en las siguientes Figuras.

### CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.42:** Diagrama de Flujo de Datos del caso de uso Salida de Artículos



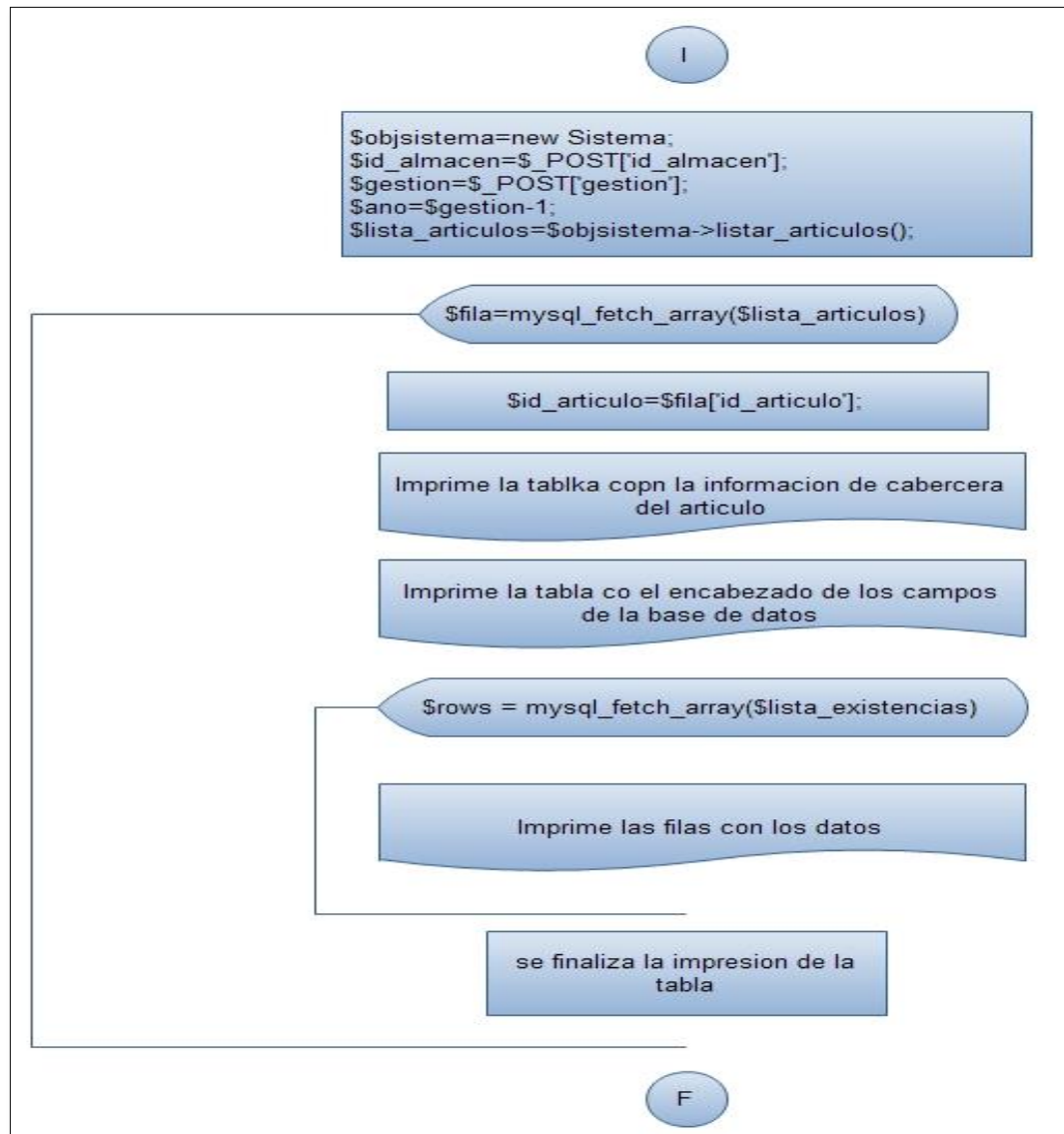
Fuente: Elaboración propia

El diagrama de flujo de la figura anterior muestra como se realiza las salida de artículos de almacén tomando en cuenta el método de valuación PEPS (primeros en entrar primeros en salir), esta forma de valuación esta normado en el (Decreto Supremo 0181, 2099).

Cuando se realiza la función de entrada y salida de artículos de almacén se calculan las existencias dentro del Sistema, las mismas que se observan en la interfaz del Sistema.

### CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.43:** Diagrama de Flujo de Datos del caso de uso Ver Existencias de Artículos

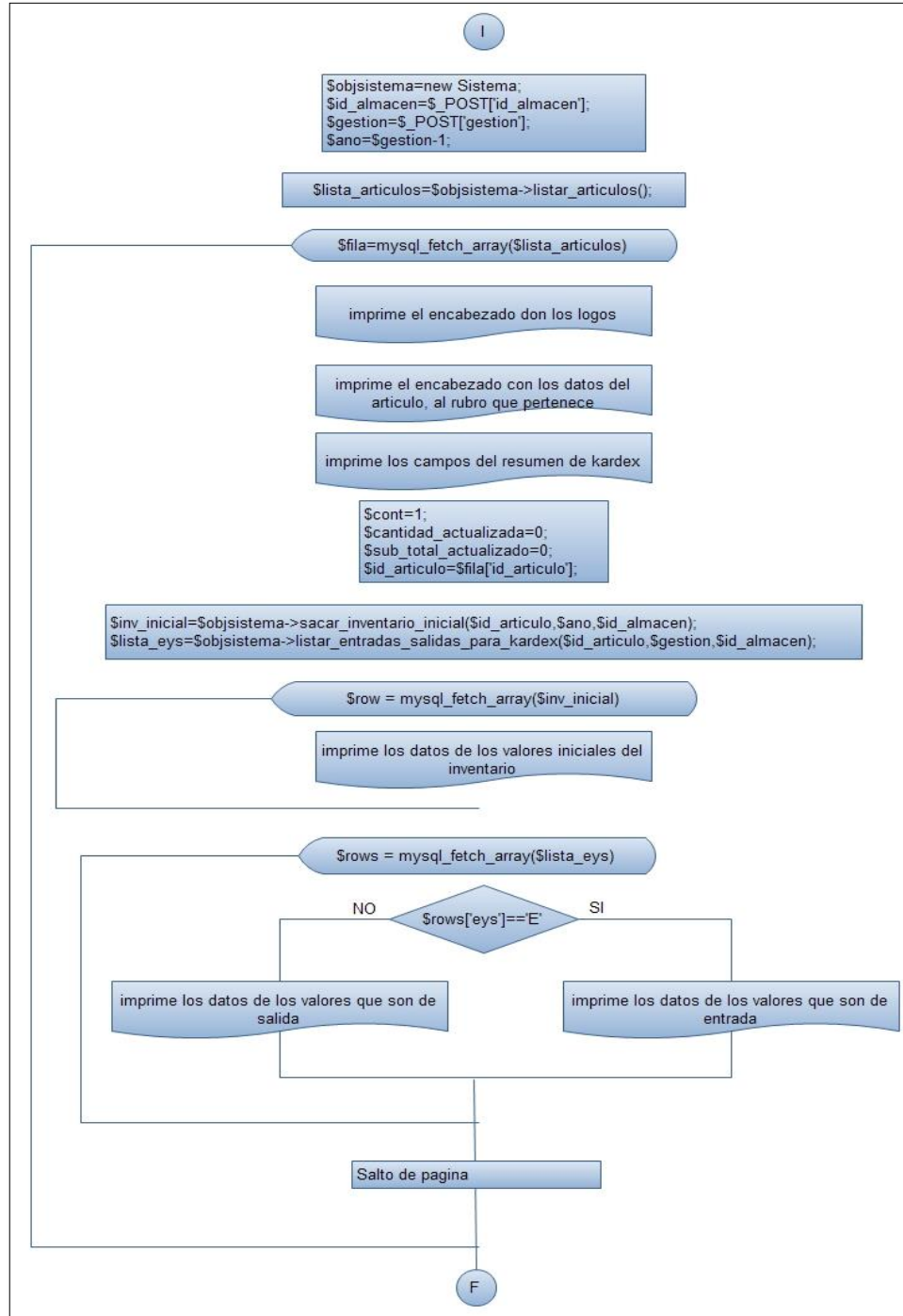


Fuente: Elaboración propia

En la siguiente Figura se muestra el diagrama de flujo que permite mostrar el Kardex general de todos los artículos agrupados por rubros<sup>1</sup>.

<sup>1</sup> Rubros. Se conoce como una clasificación en categorías de artículos

**FIGURA 3.44:** Diagrama de Flujo de Datos Reporte de Kardex General

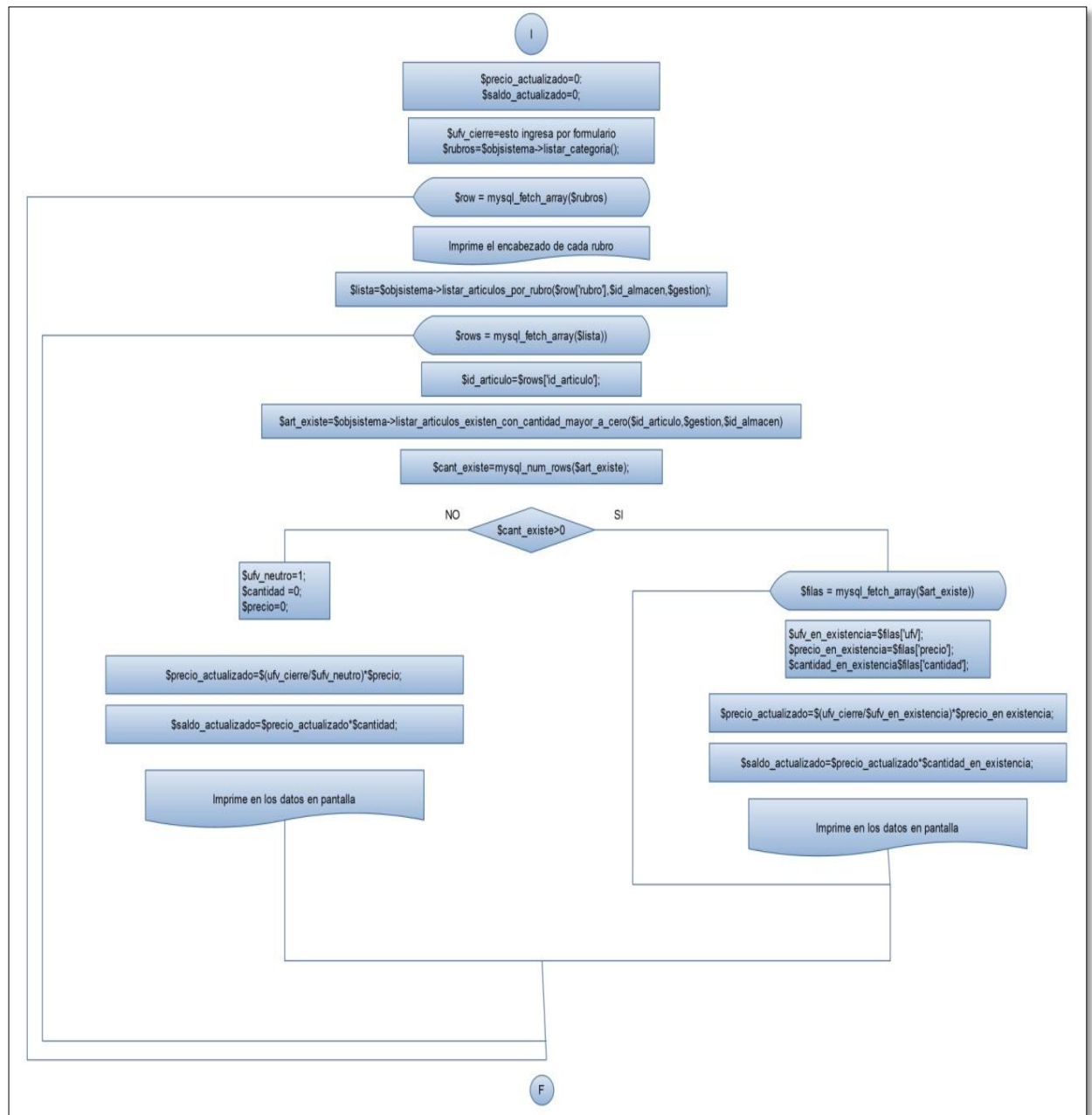


Fuente: Elaboración propia

En la siguiente Figura se observa cómo se realiza el cierre de gestión al finalizar el año, tomando en cuenta la actualización de los valores en base a la UVF de compra y el UFV de cierre de gestión.

**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.45:** Diagrama de Flujo de Datos Cierre de Gestión



Fuente: Elaboración propia

De esta manera se llegó a realizar el Modelo de objetos del sistema, aplicando los modelos OMT a detalle, aspecto que permitió determinar detalles que no estaban resueltos por completo como ser el Reporte de kardex de los artículos en general y otros.

### CAPITULO III

## DESARROLLO DEL PROYECTO

---

### 3.5. FASE DE IMPLEMENTACION Y PRUEBAS

Durante la fase de implementación según la metodología adoptada se llegó a desarrollar el código fuente del software, paralelamente se fue realizando las pruebas en el Sistema pruebas como la verificación de que si el registro funciona, los listados listan correctamente y otros.

La evaluación de la calidad del software se llegó a optar por utilizar la norma ISO/IEC 9126, para aplicar las métricas de calidad.

A continuación se muestran algunos segmentos de código fuente de las partes esenciales del Sistema de Almacén.

**FIGURA 3.46.:** Código Fuente Conectar a Base de Datos

```
<?php
class DBManager
{
    private $conexion;
    private $Servidor;
    private $BaseDatos;
    private $Usuario;
    private $Clave;
    public function __construct()
    {
        $this->conexion=null;
        $this->BaseDatos="almacen";
        $this->Servidor="localhost";
        $this->Usuario="almacen";
        $this->Clave="almacen";
    }
    public function con()
    {
        $this->conexion=mysql_connect($this->Servidor,$this->Usuario,$this-
>Clave);
        mysql_query("SET NAMES 'utf8'");
        mysql_select_db($this->BaseDatos);
        return $this->conexion;
    }
}
?>
```

Fuente: Elaboración propia

### CAPITULO III

## DESARROLLO DEL PROYECTO

FIGURA 3.47.: Código Fuente NotaIngresoUno

```
<?php
require("../clases/DBManager.class.php");
require("../clases/Almacen.class.php");
require("../clases/EntradasSalidas.class.php");
require("../clases/Funcionario.class.php");
require("../clases/Usuario.class.php");
$objusuario=new Usuario();
$objfuncionario=new Funcionario();
$objalmacen=new Almacen();
$objjeys=new EntradasSalidas();
$objdbmanager=new DBManager();
$conect=$objdbmanager->con();

$ultimo_ingreso=$objjeys->sacar_ultimo_ingreso($conect);
$datos=mysql_fetch_assoc($ultimo_ingreso);

$ingreso=$objjeys->sacar_tipo_ingreso($datos['id_tipo_ingreso'],$conect);
$ingre=mysql_fetch_assoc($ingreso);

$alm=$objalmacen->sacar_almacen($datos['id_almacen'],$conect);
$almc=mysql_fetch_assoc($alm);

?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>ingreso de articulo</title>
<link href="../css/styles.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="../js/jquery-1.2.1.pack.js"></script>
<script type="text/javascript" src="../js/sub_total_ingreso.js"></script>
<script type="text/javascript" src="../js/nuevo_ingreso.js"></script>
</head>
<body>
<br />
<div class="titulo1">AGREGANDO NUEVA NOTA DE ENTRADA</div>
<hr />
<div align="center">
<form action="" method="post" name="ingreso" id="ingreso"
onSubmit="ingreso_articulo_nuevo(); return false">
<table width="500" border="0" cellpadding="2" cellspacing="2" class="formulario">
<tr>
<td><span class="etiqueta">Tipo Ingreso</span></td>
<td><span class="etiqueta">:::</span></td>
```

Fuente: Elaboración propia

### CAPITULO III

## DESARROLLO DEL PROYECTO

---

**FIGURA 3.48.:** Código Fuente VerificarIngreso

```
<?php
require("../clases/DBManager.class.php");
require("../clases/EntradasSalidas.class.php");

$objdbmanager=new DBManager();
$objjeys=new EntradasSalidas();
$conect=$objdbmanager->con();

$tipo_documento=$_POST['tipo_documento'];
$tipo_ingreso=$_POST['tipo_ingreso'];
$proveedor=$_POST['proveedor'];
$almacen=$_POST['almacen'];
$nro_documento=trim(strtoupper($_POST['nro_documento']));
$nro_comprobante=trim(strtoupper($_POST['nro_comprobante']));
$detalle=trim(strtoupper($_POST['detalle']));
$ufv=trim($_POST['ufv']);
$fecha_ingreso=trim($_POST['fecha_ingreso']);
$year=trim(substr($fecha_ingreso, 0,4));
$id_usuario=$_POST['id_usuario'];
$nro_doc=$_POST['nro_doc'];

if ($objjeys-
>reg_nuevo_ingreso(array($nro_doc,$tipo_documento,$tipo_ingreso,$proveedor,$almacen,$id
_usuario,$nro_documento,$nro_comprobante,$detalle,$ufv,$fecha_ingreso,$year),$conect)==tr
ue)
    {
        header("Location: " . "../html/nota_ingreso_dos.php" );
        //echo "<h1>Grabacion Correcta</h1>";
    }
    else
    {
        echo "<h1>Error de grabacion</h1>";
    }

?>
```

Fuente: Elaboración propia

### CAPITULO III DESARROLLO DEL PROYECTO

**FIGURA 3.49.:** Código Fuente ValidarIngresoDetalles

```
<?php
require("../clases/DBManager.class.php");
require("../clases/EntradasSalidas.class.php");
$objdbmanager=new DBManager();
$objjeys=new EntradasSalidas();
$conect=$objdbmanager->con();
$id_egreso= $_POST['id_egreso'];
if($objjeys->actualizar_tabla_egresos($id_egreso,$conect)==true)
{
    if($objjeys->actualizar_tabla_egreso_detalle($id_egreso,$conect)==true)
    {
        //mostrar lista de notas de egreso
        header("Location: ". "../listados/listar_all_egresos.php" );
    }
    else
    {
        echo "<h1>No se puede actualizar la tabla egreso detalles</h1>";
    }
}
else
{
    echo "<h1>No se puede actualizar la tabla egresos</h1>";
}
?>
```

Fuente: Elaboración propia

La organización de todo el código fuente del sistema se muestra en la siguiente Figura.

**FIGURA 3.50.:** Organización del código fuente en el servidor

Archivos locales	Tam...	Tipo	Modificado
[-] Sitio - Sistema Almac...		Carpeta	07/06/2011 7:13
[-] clases		Carpeta	20/06/2011 21:08
[-] Connections		Carpeta	22/10/2010 11:52
[-] css		Carpeta	04/05/2011 11:48
[-] db		Carpeta	07/06/2011 11:23
[-] funciones		Carpeta	07/06/2011 22:45
[-] html		Carpeta	07/06/2011 11:12
[-] img		Carpeta	05/05/2011 9:33
[-] img psd		Carpeta	05/05/2011 9:33
[-] js		Carpeta	22/12/2010 12:00
[-] listados		Carpeta	07/06/2011 11:04
[-] reportes		Carpeta	07/06/2011 11:08
[-] SpryAssets		Carpeta	04/04/2011 9:56
[-] error.php	1KB	PHP Script	13/12/2010 17:22
[-] index.php	7KB	PHP Script	04/11/2010 12:20

Fuente: Elaboración propia

### CAPITULO III DESARROLLO DEL PROYECTO

En los siguientes gráficos se muestra muchas interfaces con las que el Sistema de Almacén interactúan los usuarios.

**FIGURA 3.51.:** Pantalla Ingreso al Sistema



Fuente: Elaboración propia

**FIGURA 3.52.:** Menú principal del Sistema



Fuente: Elaboración propia

**CAPITULO III  
DESARROLLO DEL PROYECTO**

**FIGURA 3.53.: Creación de la Nota de Entrada a Almacén**

Uuario :: INES ROSMERY CORTEZ APAZA      Rol :: ADMINISTRADOR DE ALMACEN      Fecha :: 22-06-2011      Cerrar Sesión

**Menu Sistema**

- Inicio
- Admin Parametros Sistema
- Admin Personal
- Administracion de Almacen
- Reportes

**AGREGANDO NUEVA NOTA DE ENTRADA**

**INTRODUZCA LOS DATOS**

Tipo Ingreso (\*) :: --Selecione--  
 Tipo de Documento (\*) :: --Selecione--  
 Proveedor (\*) :: --Selecione--  
 Almacen (\*) :: --Selecione--  
 Fecha Ingreso (\*) :: 2011-06-21 (yy-mm-dd)  
 Nro. Factura del Proveedor (\*) ::  
 Nro. Comprobante de Pago Contabilidad (\*) ::  
 Detalle (\*) ::  
 U.F.V. (\*) ::

Siguiente

Potenciado por :: Apache MySQL PHP Joomla

Fuente: Elaboración propia

**FIGURA 3.54.: Reporte de la Nota de Ingreso a Almacén.**

**NOTA DE INGRESO A ALMACENES**  
**SISTEMA DE GESTION DE ALMACEN**  
**GOBIERNO AUTÓNOMO MUNICIPAL DE COBIJA**

**Nro. 3/2011**

TIPO DOCUMENTO	::: HOJA DE RUTA	NRO. FACTURA PROVEEDOR	::: 12345
TIPO INGRESO	::: NORMAL (POR ADQUISICION)	NRO. COMPROBANTE DE PAGO	::: 123/2010
PROVEEDOR	::: IMPRENTA DANIELITA	FECHA INGRESO	::: 2011-01-07
NIT PROVEEDOR	::: 14473902	GESTION	::: 2011
ALMACEN	::: ALMACEN CENTRAL	USUARIO	::: admin
DETALLE	::: USDFL		

NRO.	RUBRO	COD. ARTI.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL
1	UTILES DE OFICINA	U.O-036 A	LAPICERO FABER CASTELL	500	1	500
2	PAPEL DE ESCRITORIO	P.E - 024	PAPEL BOND TAMAÑO CARTA	10	50	500
<b>SUMA TOTAL</b>						<b>1000</b>

SON ::: MIL 00/100  
BOLIVIANOS

ENCARGADO DE ALMACENES      JEFE DE UNIDAD ADM BIENES Y SERVICIOS

Fuente: Elaboración propia

### CAPITULO III DESARROLLO DEL PROYECTO

FIGURA 3.55.: Reporte Kardex del un Articulo en específico.

The screenshot shows the 'Sistema de Gestión de Almacén' interface for the 'Gobierno Autónomo Municipal de Cobija'. The user is 'INES ROSMERY CORTEZ APAZA' with the role of 'ADMINISTRADOR DE ALMACEN' on '22-06-2011'. The main menu on the left includes options like 'Inicio', 'Admin Parametros Sistema', 'Admin Personal', 'Administracion de Almacen', and 'Reportes'. The 'Reportes' section is expanded to show 'Kardex por cada Articulo'. The main content area is titled 'REPORTAR KARDEX POR CADA ARTICULO' and contains a form for 'INGRESO NOMBRE DEL ARTICULO'. The form fields are: 'Almacen' (dropdown menu set to 'ALMACEN CENTRAL'), 'Articulo' (empty text box), and 'Gestion' (dropdown menu set to '2011' with 'Ej. 2010' next to it). A 'Reportar' button is at the bottom of the form. Below the form is a 'Vista de Impresion' button and a table of inventory movements.

NRO	FECHA	DETALLE	CONTROL FISICO			PRECIO	CONTROL VALORADO		
			ENTRADAS	SALIDAS	SALDO		ENTRADA	SALIDA	SALDO
1	01/012011	INVENTARIO INICIAL	---	---	396	1	---	---	396
2	2011-01-07	USDFL	500	---	896	1	500	---	896
3	2011-01-12	ENTREGA DE MATERIAL	---	396	500	1	---	396	500
4	2011-01-12	ENTREGA DE MATERIAL	---	4	496	1	---	4	496

Fuente: Elaboración propia

FIGURA 3.56.: Formulario Cierre de Gestión.

The screenshot shows the 'Sistema de Gestión de Almacén' interface for the 'Gobierno Autónomo Municipal de Cobija'. The user is 'INES ROSMERY CORTEZ APAZA' with the role of 'ADMINISTRADOR DE ALMACEN' on '22-06-2011'. The main menu on the left includes options like 'Inicio', 'Admin Parametros Sistema', 'Admin Personal', 'Administracion de Almacen', and 'Reportes'. The main content area is titled 'FORMULARIO PARA CIERRE DE GESTION' and contains a form for 'CIERRE DE GESTION AUTOMATICO'. The form includes a warning message: 'Estas, a punto de cerrar gestion, cuando realice la operacion perdera los datos de esta gestion, y se reanudaran para la siguiente gestion, si Usted esta de acuerdo de realizar esta operacion ingrese los datos necesarios y click en el boton aceptar.' The form fields are: 'Almacen' (dropdown menu set to 'ALMACEN CENTRAL' with an asterisk), 'Correspondiente a la Gestion' (dropdown menu set to '2011' with an asterisk), 'Fecha' (text box set to '2011-06-21' with '(dd-mm-yy)' and an asterisk), and 'U.F.V. Cierre Gestion' (empty text box). 'Aceptar' and 'Cancelar' buttons are at the bottom of the form. At the bottom of the page, there are logos for 'Potenciado por' including 'Apache', 'MySQL', 'PHP', and 'AJAX'.

Fuente: Elaboración propia

**FIGURA 3.57.:** Formulario Nuevo Inventario.

The screenshot displays a web browser window with the URL `localhost:8080/almacen/html/administrador.php`. The page header includes the system title "Sistema de Gestión de Almacén" and "Gobierno Autónomo Municipal de Cobija". The user is identified as "INES ROSMERY CORTEZ APAZA" with the role "ADMINISTRADOR DE ALAMCEN" and the date "22-06-2011". A "Cerrar Sesión" button is visible in the top right.

The main content area is titled "FORMULARIO DE NUEVO INVENTARIO" and includes a "Listar Inventarios Activos" link. A sidebar menu on the left, titled "Menu Sistema", contains the following items: Inicio, Admin Parametros Sistema, Admin Personal, Administracion de Almacen, and Reportes.

The "NUEVO INVENTARIO" form contains the following fields:

- Almacen: --Seleccione-- (dropdown)
- Fecha Inventario: 2011-06-21 (calendar icon) (yy-mm-dd)
- Trimestre: PRIMER TRIMESTRE (dropdown)
- Detalle: (text input)
- Gestion: (text input) Ej: 2010

A "Guardar" button is located at the bottom of the form. At the bottom of the page, there are logos for "Potenciado por" and "Powered by" for Apache, MySQL, PHP, and Joomla.

Fuente: Elaboración propia

Las figuras mostradas anteriormente certifican la prueba del funcionamiento del sistema, de esta manera se llega a concluir el desarrollo del sistema de almacén.

### **3.5.1. Valoración de Calidad del Software**

La evaluación se refiere a la comprobación de la calidad del sistema elaborado en base a la Metodología OMT, para lo cual se aplica las métricas de calidad de la Norma ISO/IEC 9126.

#### **3.5.1.1. Calidad del Sistema de Almacén**

Basados en la norma de calidad ISO/IEC 9126 (**Ver Anexo B**) se ha elaborado un cuestionario de evaluación (**Ver Anexo E**), establece la forma de medir la calidad del Sistema, del cual se toma estos resultados el Software fue valorado con índices elevados en los criterios de: Utilización del usuario con acceso al Sistema.

Se ha asignado un peso (valor numérico) a cada atributo del Sistema de Almacén de acuerdo al grado de importancia o nivel de necesidad exigida por la responsable de la

### **CAPITULO III**

### **DESARROLLO DEL PROYECTO**

---

Unidad de Almacén del Gobierno Autónomo Municipal de Cobija, va desde 1 que significa muy poco importante, hasta el 10 que significa extremadamente importante, se ajusto cada pregunta de la evaluación a un atributo exigido, con estos resultados se aplicó una fórmula para obtener las métricas a cada atributo. El usuario califica un puntaje desde 1 hasta 5, de acuerdo a la siguiente escala:

**TABLA 3.17** Escala de calificación para el Sistema de Almacén

<b>Calificación</b>	<b>Puntaje</b>
Muy mal	1
Pésimo	2
Regular	3
Bueno	4
Muy Bueno	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

De acuerdo a la norma ISO/IEC 9126, esta se categoriza en seis elementos para realizar la evaluación del sistema. A continuación se detalla estos elementos:

#### **a. Funcionabilidad**

**TABLA 3.18** Resultado de la medida Funcionabilidad

<b>Atributo</b>	<b>Peso</b>	<b>Resultado</b>
Adecuación	10	4
Exactitud	10	5
Seguridad	10	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Funcionabilidad} = \frac{10*4 + 10*5 + 10*5}{10+10+10}$$

$$\text{Medida de Funcionabilidad} = 4,6$$

De acuerdo al resultado obtenido, se califica como valor aceptable, por lo que el Sistema de Almacén si cumple con la métrica de funcionalidad de acuerdo a los requerimientos exigidos por el responsable de almacén.

### CAPITULO III DESARROLLO DEL PROYECTO

---

#### b. Fiabilidad

**TABLA 3.19** Resultado de la medida Fiabilidad

Atributo	Peso	Resultado
Madurez	10	5
Tolerancia a Fallas	10	5
Recuperabilidad	10	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Fiabilidad} = \frac{10*5 + 10*5 + 10*5}{10+10+10}$$

$$\text{Medida de Fiabilidad} = 5$$

De acuerdo al resultado obtenido, se califica como valor aceptable de la evaluación del Sistema de Almacén en su métrica de fiabilidad (tiene validaciones sobre los campos, se recupera la información almacenada, etc.) lo cual significa que el sistema es seguro y recupera los datos almacenados fácilmente.

#### c. Usabilidad

**TABLA 3.20** Resultado de la medida Usabilidad

Atributo	Peso	Resultado
Entendimiento	10	5
Aprendizaje	10	5
Operabilidad	7	5
Atracción	5	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Usabilidad} = \frac{10*5 + 10*5 + 7*5 + 5*5}{10 + 10 + 7 + 5}$$

$$\text{Medida de Usabilidad} = 5$$

De acuerdo al resultado obtenido en su métrica de Usabilidad, se establece que el Sistema de Almacén tiene una gran aceptación por parte del usuario, en el manejo de

### CAPITULO III DESARROLLO DEL PROYECTO

---

las interfaces y la facilidad de aprendizaje. Por el cual se otorga una buena puntuación, en esta fase es importante estar entre la puntuación de cuatro a cinco.

#### d. Eficiencia

**TABLA 3.21** Resultado de la medida Eficiente

Atributo	Peso	Resultado
Comportamiento de Tiempos	10	5
Utilización de los recursos	10	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Eficiencia} = \frac{10*5 + 10*5}{10 + 10}$$

$$\text{Medida de Eficiencia} = 5$$

De acuerdo al resultado obtenido, se establece que el Sistema de Almacén, cumple con los requerimientos exigidos en un determinado tiempo (hace lo que tiene que hacer, no demora mucho tiempo su ejecución), por lo que el software obtiene la puntuación aceptable de acuerdo a la métrica de eficiencia.

#### e. Mantenimiento

**TABLA 3.22** Resultado de la medida Mantenibilidad

Atributo	Peso	Resultado
Capacidad de ser analizado	10	4
Cambiabilidad	7	4
Estabilidad	5	3
Facilidad de Prueba	5	4

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Mantenimiento} = \frac{10*4 + 7*4 + 5*3 + 5*4}{10 + 7 + 5 + 5}$$

$$\text{Medida de Mantenimiento} = 3,8$$

### CAPITULO III DESARROLLO DEL PROYECTO

---

De acuerdo al resultado obtenido, se establece que el sistema de almacén cumple con los requerimientos mínimos exigidos para su mantenimiento y aceptabilidad. Por lo que se obtuvo una puntuación de **3.8**, lo cual significa que es aceptable.

#### f. Portabilidad

**TABLA 3.23** Resultado de la medida Portabilidad

Atributo	Peso	Resultado
Adaptabilidad	10	5
Coexistencia	5	5
Facilidad de Instalación	10	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

$$\text{Medida de Portabilidad} = \frac{10*5 + 5*5 + 10*5}{10 + 5 + 10}$$

$$\text{Medida de Portabilidad} = 5$$

De acuerdo al resultado, se establece el sistema de almacén es multiplataforma (Windows, Linux) y puede coexistir con otros sistemas, también es muy fácil su instalación por lo cual obtiene la puntuación más alta.

*En conclusión se tiene el siguiente resultado que se visualiza en el Tabla:*

**TABLA 3.24** Resultados generales de la medida estándar ISO/IEC 9126

Atributos	Puntaje
Funcionalidad	4,6
Fiabilidad	5
Usabilidad	5
Eficiencia	5
Mantenimiento	3,8
Portabilidad	5

**Fuente:** Elaboración propia en base a (Hernaz&Minguet, 2006)

### **CAPITULO III**

### **DESARROLLO DEL PROYECTO**

---

La Tabla 3.24, indica que el Sistema de Almacén cumple con las Métricas de Calidad de la Norma **ISO/IEC 9126**, por lo que **SÍ** es pertinente utilizarlo en la unidad de Almacén dependiente del Gobierno Autónomo Municipal de Cobija.

## CAPITULO III DESARROLLO DEL PROYECTO

---

### CAPITULO III

3.1.	FASE DE CONCEPTUALIZACION.....	41
3.1.1.	Requerimientos.....	41
3.1.2.	Casos de uso .....	43
3.2.	FASE DE ANALISIS .....	60
3.2.1.	Modelo de Objetos .....	60
i.	Identificación de Clases .....	61
ii.	Diccionario de datos.....	61
iii.	Asociación de Clases.....	62
3.2.2.	Modelo Dinámico.....	62
3.2.3.	Modelo Funcional.....	64
3.3.	FASE DE DISEÑO DEL SISTEMA .....	65
3.3.1.	Módulos del Sistema .....	65
3.3.2.	Arquitectura básica del sistema .....	66
3.3.3.	Procesos o funcionalidades de los módulos .....	66
3.3.4.	Estructura y diseño de la base de datos .....	67
3.4.	FASE DE DISEÑO DE OBJETOS.....	69
3.4.1.	Modelo de Objetos Detallado.....	69
3.4.2.	Modelo Dinámico detallado .....	72
3.4.3.	Modelo Funcional Detallado .....	80
3.5.	FASE DE IMPLEMENTACION Y PRUEBAS .....	85
3.5.1.	Valoración de Calidad del Software.....	92
3.5.1.1.	Calidad del Sistema de Almacén .....	92
a.	Funcionabilidad .....	93
b.	Fiabilidad.....	94
c.	Usabilidad.....	94
d.	Eficiencia.....	95
e.	Mantenimiento.....	95
f.	Portabilidad.....	96

**CAPITULO III**  
**DESARROLLO DEL PROYECTO**

---

## **CAPITULO IV**

### **CONCLUSIONES Y RECOMENDACIONES**

---

#### **4.1. CONCLUSIONES**

El desarrollo tecnológico actual facilita en gran medida la administración de información en las instituciones, facilitando el almacenamiento de la información en grandes cantidades superando las limitaciones físicas que se imponen. Con la realización de este software que se elaboro en base a la *Normativa de Administración de Almacenes del Estado Boliviano*, se estableció que es una labor muy importante que deben cumplir todas instituciones gubernamentales y requiere la concientización a los funcionarios públicos.

De acuerdo a la culminación del proceso del proyecto de grado se elaboran las siguientes conclusiones:

1. Se diagnostico los procesos identificando los requerimientos de la Unidad de Almacén dependiente del Gobierno Autónomo Municipal de Cobija.
2. Se realizó el respectivo análisis del sistema construyendo el modelo de objetos, modelo dinámico y el modelo funcional.
3. Se realizó el diseño del sistema en base a la metodología OMT.
4. Se desarrollo los módulos del sistema con sus respectivas funcionalidades o procesos.
5. Se realizaron las pruebas del funcionamiento de los módulos del sistema.
6. Se implemento el sistema en la Unidad de Almacén para su respectivo uso.

Después de su implementación del Sistema en la Unidad de Almacén se logro registrar todos los artículos con sus respectivos códigos, las entradas y salidas de artículos están en un proceso de regularización del mes de Enero y Febrero, las existencias y el inventario se va generando automáticamente conforme se realizan las entras y salidas.

El Kardex de cada artículo se puede reportar en el menor tiempo, por medio del Sistema el cual muestra el balance de todas las entradas y salidas.

También se logro implementar en el Sistema, el método de valuación Primeros en Entrar Primeros en Salir (PEPS), exigidas por la Norma de Administración de Almacenes.

## **CAPITULO IV**

### **CONCLUSIONES Y RECOMENDACIONES**

---

#### **4.2. RECOMENDACIONES**

Al culminar el proyecto de grado se plantea las siguientes recomendaciones:

1. Se recomienda realizar la conexión de Intranet entre el Predio Central y la Unidad de Almacén, para que el Sistema funcione en todas las dependencias de la Alcaldía.
2. Se recomienda desarrollar e implementar otros Módulos al Sistema por ejemplo del Módulo de Pedidos.
3. Se recomienda que el Proyecto de Grado sirva como material de consulta para los estudiantes de la Universidad Amazónica de Pando

**CAPITULO IV**  
**CONCLUSIONES Y RECOMENDACIONES**

---

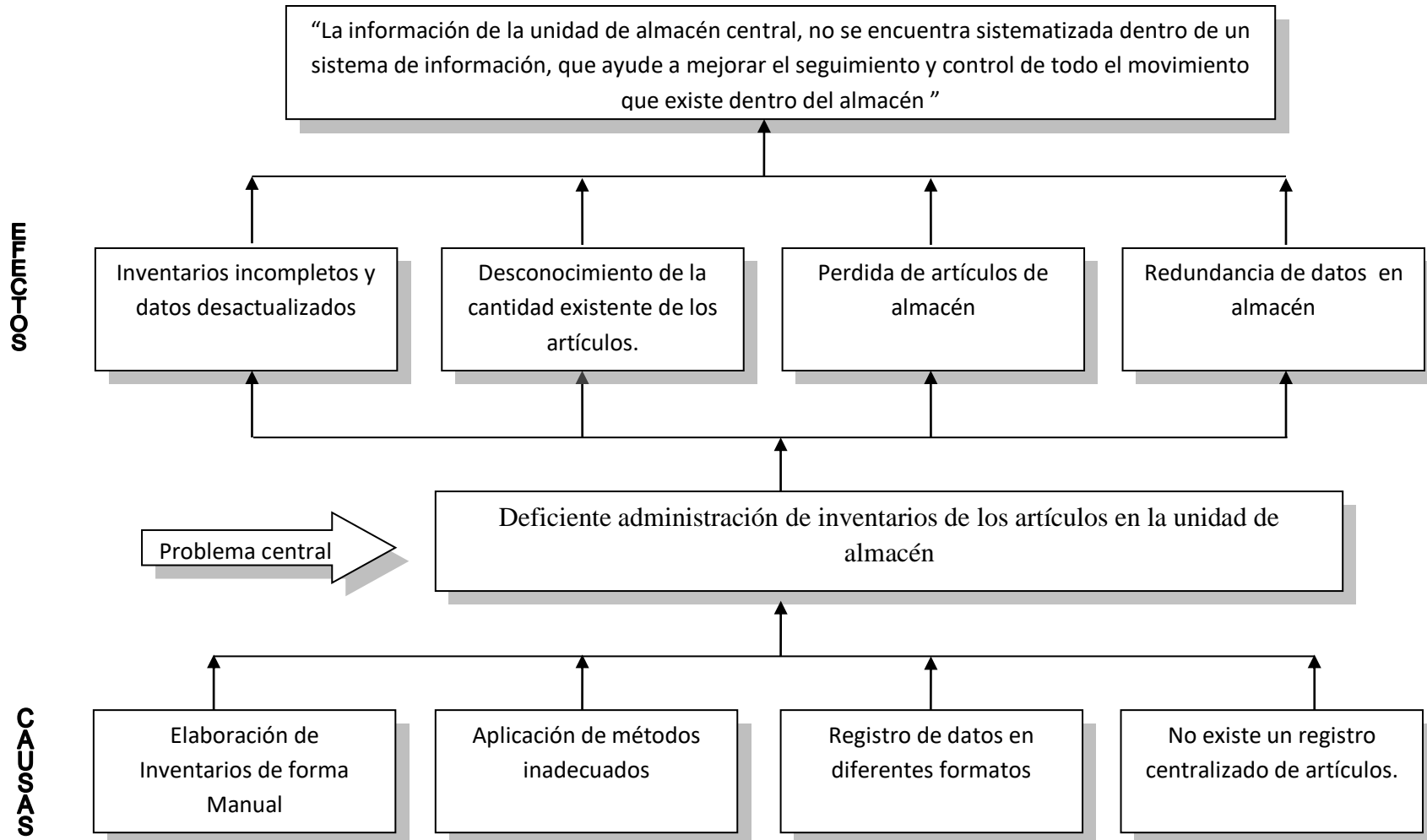
4.1. CONCLUSIONES..... 98

4.2. RECOMENDACIONES ..... 99

**ANEXO "A"**  
**ARBOL DE PROBLEMAS Y OBJETIVOS**

---

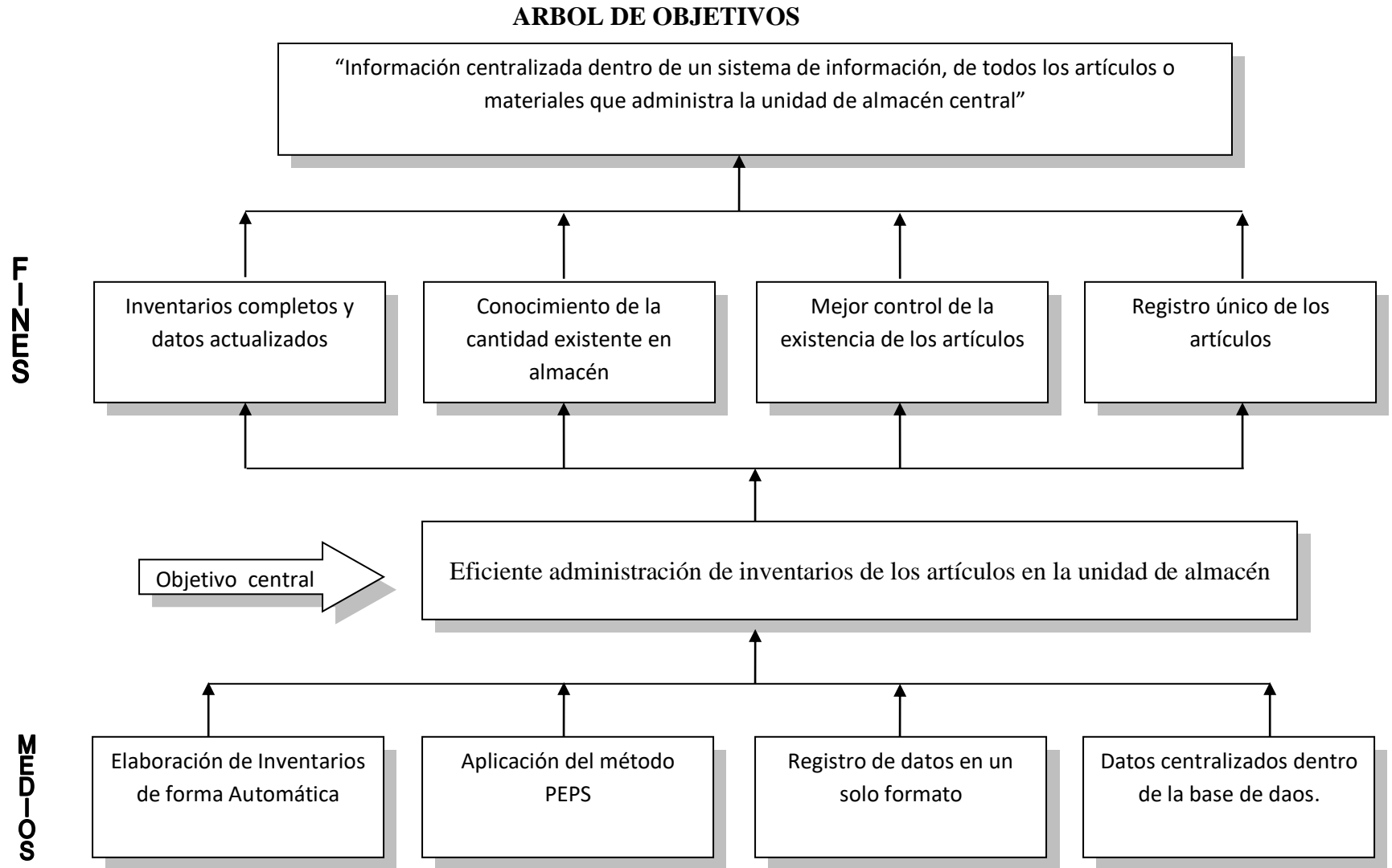
**ARBOL DE PROBLEMAS**



Fuente: Elaboración propia

**ANEXO "A"**  
**ARBOL DE PROBLEMAS Y OBJETIVOS**

---



Fuente: Elaboración propia

## ANEXO “B”

### CALIDAD DE SOFTWARE, NORMA ISO/IEC 9126

---

#### 1. Calidad del software

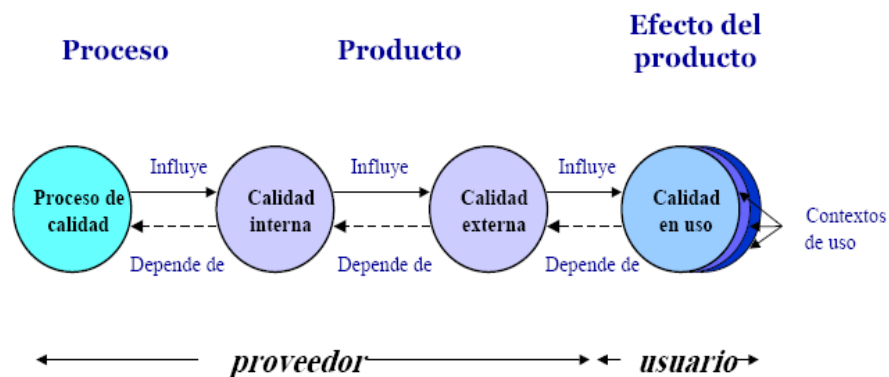
Actualmente existen diversos conceptos sobre la calidad de software, pero de acuerdo a la investigación de la tesis se mencionan las siguientes afirmaciones:

- Según (PRESSMAN, 2002), la calidad del software es la concordancia con los requerimientos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo (documentados) y con las características (implícitas) que se espera de todo el software.
- Según (IEEE, 1990). “La calidad del software es el grado con el que un Sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario”.

Para elaborar la calidad del software, es necesario comprender las necesidades reales de los usuarios con tanto detalle como sea posible (requisitos), a continuación se muestra algunos conceptos importantes:

- **Calidad:** Conjunto de propiedades y características de un producto que le confieren su aptitud para satisfacer unas necesidades explícitas o implícitas”
- **Control de calidad:** Conjunto de técnicas y actividades de carácter operativo, utilizadas para verificar los requerimientos relativos a la calidad del producto.
- **Garantía de calidad:** Conjunto de acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisfará los requerimientos dados sobre calidad”.

FIGURA B.1: Calidad del producto



Fuente: Basado en el de (PRESSMAN, 2002)

## ANEXO “B”

### CALIDAD DE SOFTWARE, NORMA ISO/IEC 9126

#### 1.1. Diferentes aspectos de la calidad

**Calidad Interna:** Medible a partir de las características intrínsecas, (código fuente)

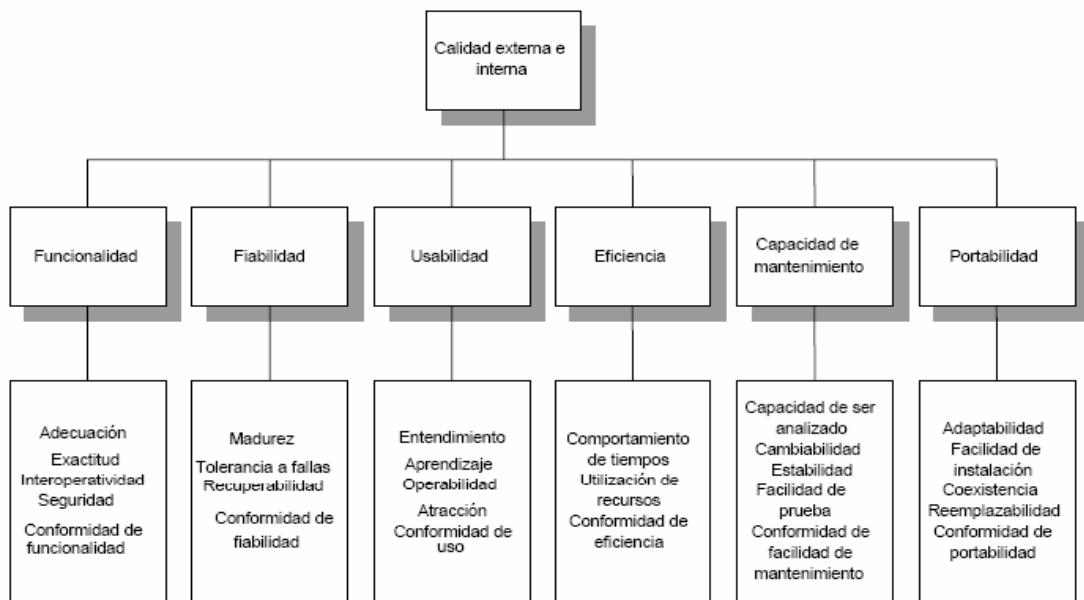
**Calidad Externa:** Medible en el comportamiento del producto, (prueba, piloto)

**Calidad de uso:** Durante la utilización efectiva por parte del usuario (evolución)

#### 2. Norma de calidad Internacional ISO/IEC 9126

Establece los lineamientos generales para la evaluación del software a partir de seis categorías de calidad: *Funcionalidad, fiabilidad, eficiencia, usabilidad, portabilidad y mantenibilidad.*

Figura B.2: Norma ISO/IEC 9126



Fuente: (Hernanz&Minguet, 2006)

Según la figura H2, cada una de las seis categorías está determinada por sub categorías que la concretan. A continuación se detalla estas categorizaciones:

##### 2.1. Funcionalidad

Es la capacidad del software para proveer las funciones que satisfacen las necesidades explícitas (lo que hace para satisfacer) e implícitas (cuando y como satisface), cuando el software se utiliza bajo condiciones específicas.

- **Adecuación:** Capacidad del software para proveer un adecuado conjunto de funciones para las tareas y objetivos especificados por el usuario.

- **Exactitud:** Capacidad del software para proveer los resultados o efectos correctos con un grado de precisión.
- **Interoperabilidad:** Capacidad del software para interactuar con uno o más sistemas especificados.
- **Seguridad de acceso:** Capacidad del software para proteger información y datos de forma que los usuarios no autorizados no puedan acceder.
- **Cumplimiento funcional:** Capacidad del software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con funcionalidad.

## **2.2. Fiabilidad**

Es la capacidad del software para mantener un nivel específico de funcionamiento cuando se está utilizando bajo condiciones especificadas. Según [ISO/IEC, 1997], es la habilidad de la unidad funcional de realizar una función requerida.

- **Madurez:** Capacidad del software para evitar fallas
- **Tolerancia a fallos:** Capacidad del software para mantener un nivel especificado de funcionamiento en caso de errores del software o de incumplimiento de su interfaz especificado.
- **Recuperabilidad:** Capacidad del software para mantener un nivel de funcionamiento y de recuperar los datos afectados en caso de fallos.
- **Cumplimiento de la fiabilidad:** Capacidad del software para adherirse a normas, convenciones o regulaciones relacionadas con la fiabilidad.

## **2.3. Usabilidad**

Capacidad del software de ser entendido, aprendido, usado y atractivo al usuario cuando es utilizado bajo las condiciones especificadas, se debe regir a todos los diversos ambientes del usuario.

- **Entendimiento:** Capacidad del software que permite al usuario entender el funcionamiento o como puede ser usado para tareas de uso particular.
- **Aprendizaje:** Capacidad del software que permite al usuario aprender sobre su aplicación.

- **Operabilidad:** Capacidad del software que permite al usuario operarlo y controlarlo.
- **Atracción:** Capacidad del software para ser atractivo al usuario.
- **Cumplimiento de la usabilidad:** Capacidad del software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.

#### **2.4. Eficiencia**

Es la capacidad del software para proveer un desempeño adecuado, de acuerdo a la cantidad de recursos utilizados y bajo las condiciones planeas.

- **Comportamiento temporal:** Capacidad del software para proporcionar tiempos de respuesta adecuados, tiempos de proceso y potencia apropiados bajo condiciones establecidas.
- **Utilización de recursos:** Capacidad del software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.
- **Cumplimiento de la eficiencia:** Capacidad del software para adherirse a normas o convenciones relacionadas con la eficiencia.

#### **2.5. Mantenibilidad**

Es la capacidad del software para ser modificado, las modificaciones pueden incluir correcciones, mejoras o adaptaciones del software a cambios en el entorno.

- **Capacidad para ser analizado:** Es la capacidad del software para atenerse a diagnósticos de deficiencia a causas de los fallos en el software.
- **Cambiabilidad:** Capacidad del software para permitir que una determinada modificación sea implementada.
- **Estabilidad:** Capacidad del software para evitar efectos inesperados debidos a modificaciones del software.
- **Facilidad de pruebas:** Capacidad del software para permitir que el software modificado sea validado.

## ANEXO “B”

### CALIDAD DE SOFTWARE, NORMA ISO/IEC 9126

---

- **Cumplimiento de la mantenibilidad:** Capacidad del software para adherirse a normas o convenciones relacionadas con la mantenibilidad.

#### 2.6. Categoría Portabilidad:

Capacidad del software para ser trasladado de un entorno a otro (Hardware, Software).

- **Adaptabilidad:** Capacidad del software para ser adaptado a diferentes entornos especificados.
- **Facilidad de Instalación:** Capacidad del software para ser instalado en un ambiente especificado.
- **Coexistencia:** Capacidad del software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.
- **Reemplazabilidad:** Capacidad del software para ser usado en lugar de otro software, para el mismo propósito, en el mismo entorno.
- **Cumplimiento de la portabilidad:** Capacidad del software para adherirse a normas o convenciones relacionadas con la portabilidad.

### 3. Métricas de Calidad

Consistes en utilizar las métricas más importantes aplicables a determinados atributos para obtener un único valor que permita medir el producto, para obtener el resultado final, se tienen que calcular los resultados de cada métrica aplicable y posteriormente juntar esos resultados utilizando la siguiente fórmula:

$$\text{Medida de SubCategorías} = \frac{r_1 m_1 + r_2 m_2 + \dots + r_n m_n}{r_1 + r_2 + \dots + r_n}$$

Donde  $r_1$  identifica el peso que se asigna a la métrica y  $m_1$  representa las métricas asociadas a las características. Esta fórmula se aplica a cualquier atributo de la norma ISO/IEC 9126, los pesos asociados a cada métrica está entre 1 a 10.

**MANUAL DE INSTALACION DEL SISTEMA DE GESTION DE ALMACEN**

Después de haber instalado el sistema operativo **Linux**, en la distribución de Ubuntu 9.10 se debe seguir el siguiente procedimiento

**1.- INSTALANDO PROGRAMAS NECESARIOS PARA QUE FUNCIONE EL SISTEMA.-**

Primeramente se debe crear clave para el usuario root, para eso abrimos la consola y escribimos lo siguiente:

```
user $: sudo passwd root
```

Ingresamos los datos que piden para luego ingresar como usuario root al sistema operativo, para eso introducimos lo siguiente.

```
user $: su root
```

Para actualizar la lista de programas ejecutamos el siguiente comando en consola

```
user #: aptitude update  
user#: aptitude upgrade
```

Una vez ingresado como usuario root se actualiza la lista de programas disponibles, luego comenzáramos a instalar los programas necesarios para que funcione el Sistema.

Primero Instalaremos Mysql el manejador de base de datos

```
root #: apt-get install mysql-server-5.1
```

Ingresamos la clave que preferimos utilizar esto para tener seguridad sobre la base de datos.

Segundo instalamos apache2 el servidor web.

```
root #: apt-get install apache2
```

El servidor apache2 nos permite trabajar en red con otros equipos que son clientes y que pueden visualizar el sistema por medio de un navegador web (firefox mozilla recomendado)

## ANEXO "C"

### MANUAL DE INSTALACIÓN

---

Después instalamos el lenguaje de programación php5

```
root #: apt-get install php5
```

Luego instalamos los archivos necesarios para que MySQL soporte php5 y apache2

```
root #: apt-get install libapache2-mod-auth-mysql  
root #: apt-get install php5-mysql
```

Y luego reiniciamos el servicio de apache

```
root #: /etc/init.d/apache2 restart
```

Luego probar en el navegador web si apache está funcionando

```
http://localhost
```

Si todo funciona bien el sistema operativo ya está preparado para instalar el sistema de gestión de almacén.

**ANEXO “C”  
MANUAL DE INSTALACIÓN**

---

**INSTALANDO CODIGO FUENTE DEL SISTEMA Y BASE DE DATOS**

El sistema está en el CD-ROM en un archivo .rar para descomprimir se debe instalar el siguiente programa por consola.

```
root #: apt-get install rar unrar
```

Para descomprimir el archivo.rar simplemente se presiona la tecla enter, después de descomprimir se debe copiar toda la carpeta al siguiente directorio.

```
root #: /var/www/
```

Con eso ya está, ahora para crear la base de datos se debe instalar el siguiente programa que permite administrar la base de datos de manera visual.

```
root #: apt-get install phpmyadmin
```

Una vez instalado ingresamos al programa por el navegador web.

```
http://localhost/phpmyadmin
```

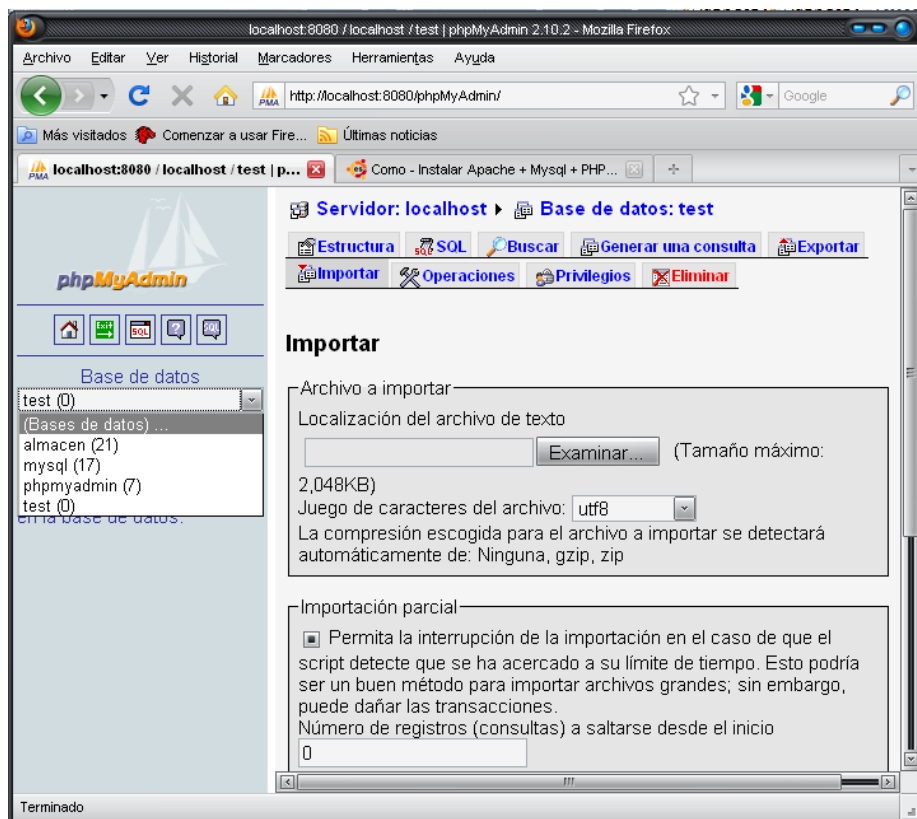
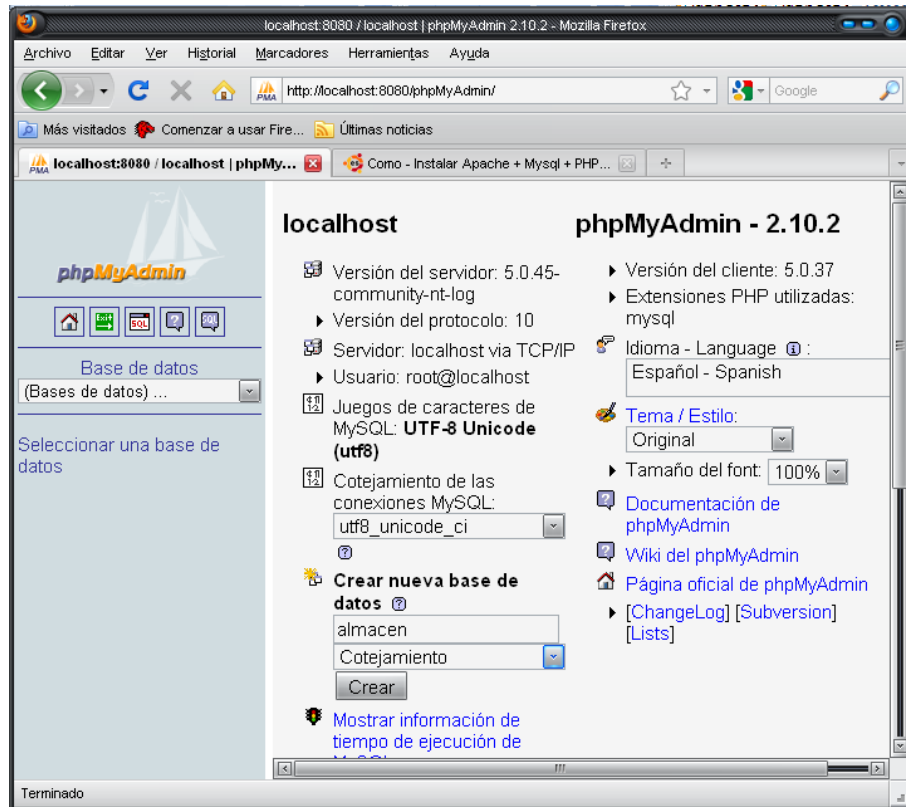
Luego nos pide autenticarse `para eso ingresamos los siguientes datos

**Usuario = root**

**Contraseña = (es la contraseña que ingresaste al momento de instalar mysql-serve-5.1)**

Luego procedemos a crear la base de datos con el nombre almacén luego click en el botón crear (así como se observa en la imagen).

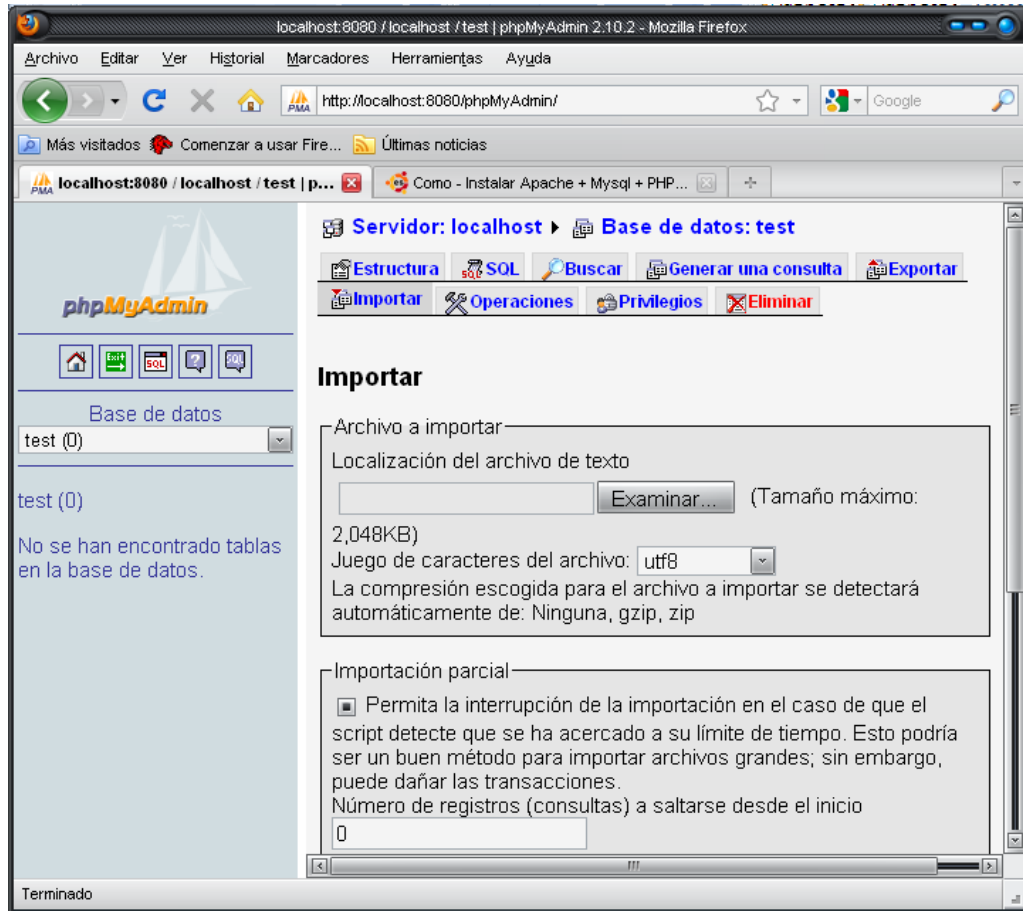
# ANEXO “C” MANUAL DE INSTALACIÓN



## ANEXO “C” MANUAL DE INSTALACIÓN

---

Importamos el script de la base de datos, buscamos el archivo con el botón examinar



El script se encuentra en el siguiente directorio

```
/var/www/almacen/db/almacen_estru_datos_oficial.sql
```

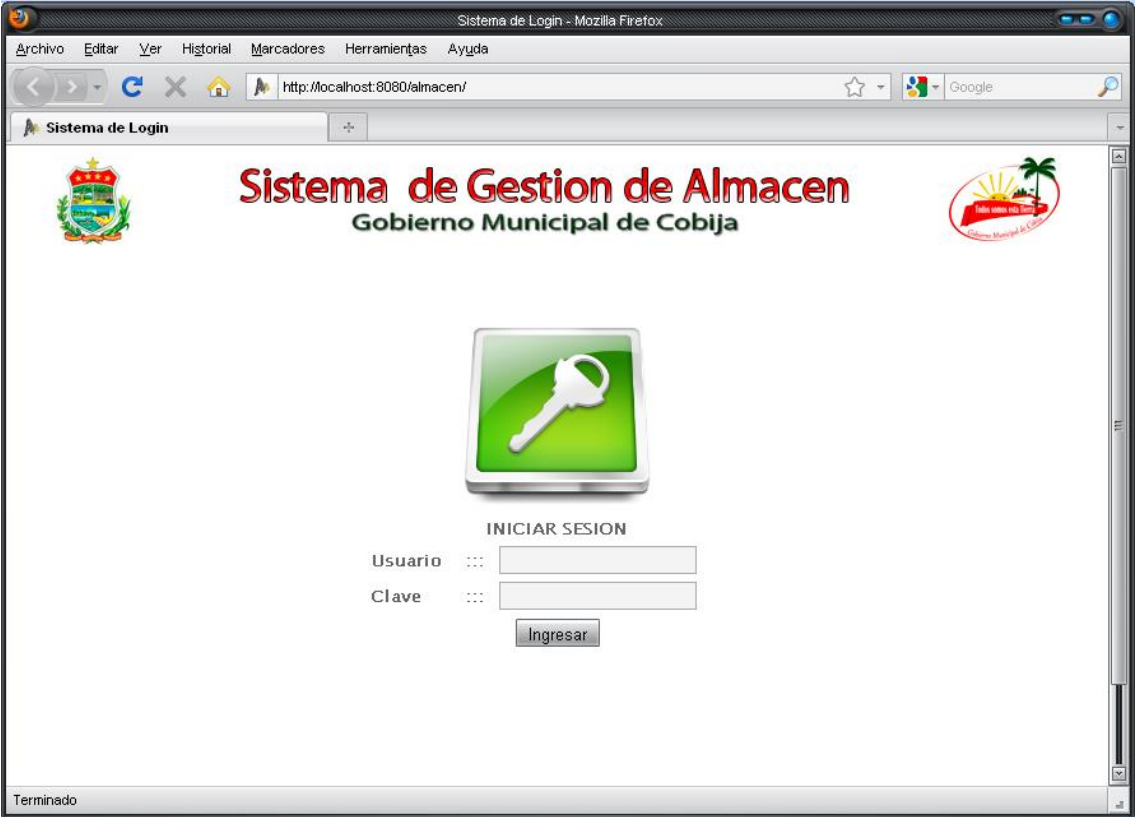
Una vez importado la base de datos se debe ejecutar el navegador web firefox mozilla e ingresar la siguiente dirección web.

```
http://localhost/almacen
```

# ANEXO “C” MANUAL DE INSTALACIÓN

---

Pantalla inicio donde pide nombre del usuario y la clave de autenticación.



## ANEXO “D” MANUAL DE USUARIO

### 1. INGRESO AL SISTEMA

Para ingresar al sistema se debe utilizar un navegador web (recomendado firefox mozilla 3.5), en la barra de dirección se debe escribir la siguiente URL.

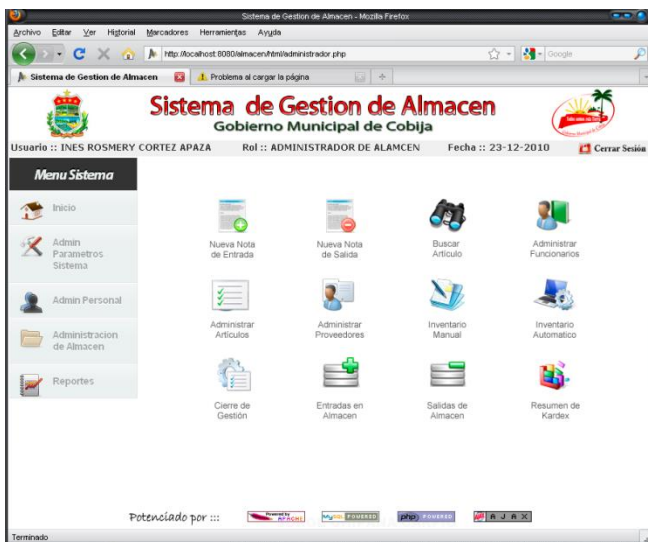
<http://localhost/almacen>

<http://192.168.1.xx/almacen>

Y obtendremos el siguiente resultado:



En el formulario Iniciar Sesión ingresamos datos del usuario más la clave de acceso y hacemos click sobre el botón Ingresar.



Después de ingresar se observa el menú general del sistema con las opciones para administrar las entradas y salidas de almacén.

De este modo ingresamos al Sistema de Gestión de Almacén, en el siguiente apartado veremos cómo administrar usuarios.

## 2. ADMINISTRANDO USUARIOS

La opción de administrar usuarios, nos permite crear, editar y eliminar usuarios que administran el sistema.

Lista de Usuarios que Administran el Sistema

NRO.	USUARIO	ROL	NOMBRE COMPLETO	FECHA REGISTRO	EDITAR	ELIMINAR
1	jorge	SUPER ADMINISTRADOR	JORGE CHIPANA CARPIO	2010-12-09 17:35:53		
2	donpoma	ENCARGADO DE ENTRADAS Y SALIDA	MARCO ANTONIO POMA CHOQUEHUANCA	2010-12-09 17:40:17		
3	edgar	RESPONSABLE DE KARDEX	EDGAR NINA BARRETO	2010-12-09 17:42:14		
4	manuel	ENCARGADO DE ENTRADAS Y SALIDA	MANUEL LÓPEZ	2010-12-09 17:42:49		
5	auditoria	RESPONSABLE DE AUDITORIA	ENCARGADO DE AUDITORIA	2010-12-09 17:43:13		
6	admin	ADMINISTRADOR DE ALMACEN	INES ROSMERY CORTEZ APAZA	2010-12-13 11:03:38		
7	prueba	ENCARGADO DE ENTRADAS Y SALIDA	PRUEBA	2010-12-22 09:16:32		

En el menú Admin Personal y la opción Usuarios del Sistema, obtenemos la lista completa de usuarios que estas activos con sus respectivos roles.

FORMULARIO DE EDICION DE USUARIO

INGRESE NUEVOS DATOS

Nombre ::: MARCO ANTONIO POM (\*)

Usuario ::: donpoma (\*)

Nueva Clave ::: (\*)

Rol ::: Administrador (\*)

Aceptar Cancelar

Dentro de la lista general de usuarios activos, cuenta con la opción de editar información, dentro de este formulario se puede cambiar el nombre de la persona, nombre del usuario y la clave de acceso.

## ANEXO “D” MANUAL DE USUARIO



Dentro de la lista general de usuarios activos, cuenta con la opción de eliminar información, este formulario permite eliminar el usuario seleccionado o cancelar la acción a realizar.



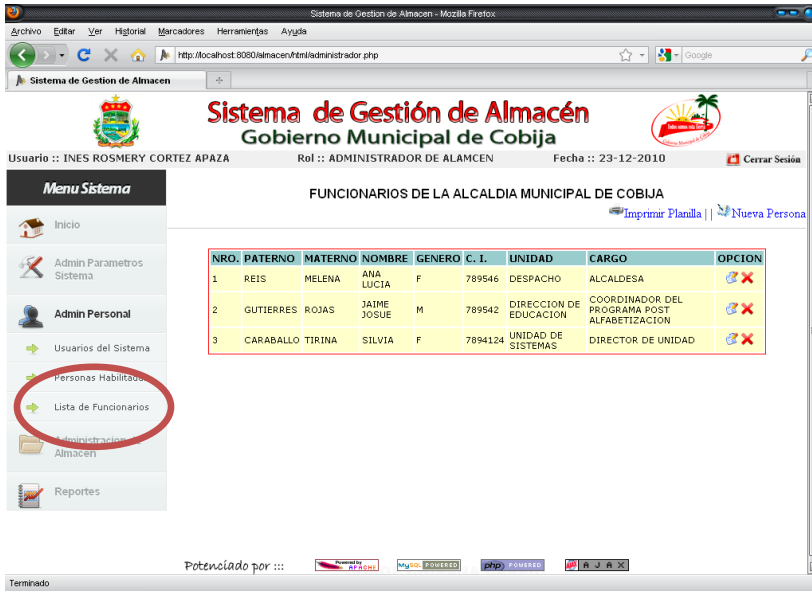
La opción Nuevo Usuario, permite crear un nuevo usuario del sistema, definiendo el nombre completo, el nombre del usuario, la clave de acceso y el tipo de usuario o el rol que va tener dicho usuario.

De este modo administramos los usuarios del sistema, en el siguiente apartado veremos cómo Administrar funcionarios de la Alcaldía.

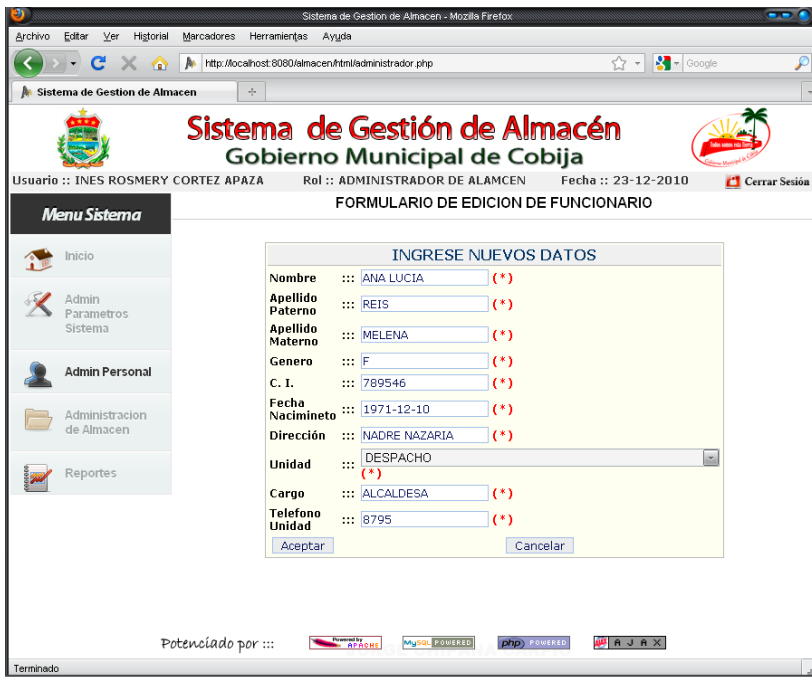
# ANEXO “D” MANUAL DE USUARIO

## 3. ADMINISTRAR FUNCIONARIOS DE LA ALCALDIA

Este modulo está con finalidad de tener el control de los usuarios que realizan pedidos a la unidad de almacén, cuando se realiza la entrega de algún artículo se debe registrar a que funcionario y a que unidad se está entregado dicho artículo, entonces este modulo permite administrar los funcionarios que están activos en el G.M.C.



Haciendo click sobre la opción Lista de Funcionarios, se obtiene la lista completa de funcionarios que están activos en la alcaldía también muestra a la unidad o dirección que pertenecen más el cargo que tienen.

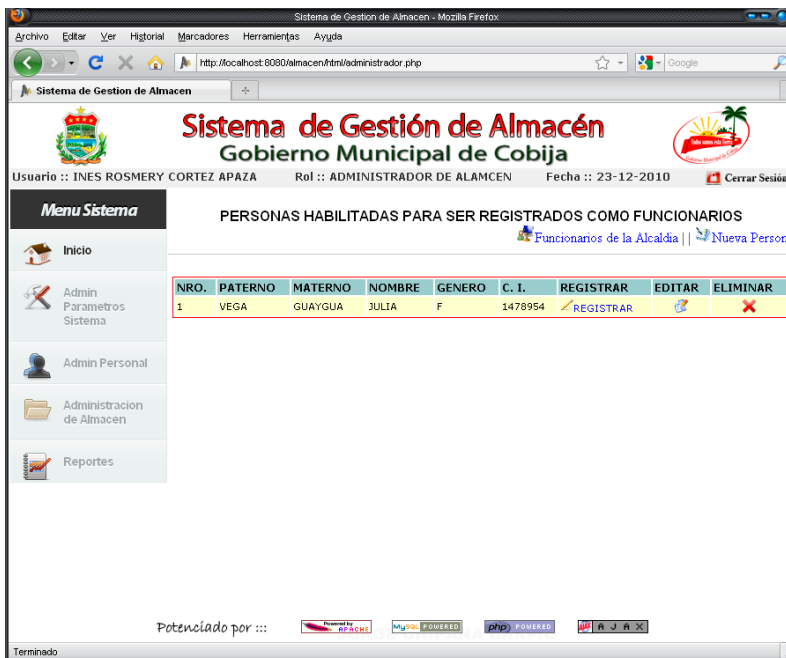


También se puede editar y eliminar los funcionarios del G.M.C.

# ANEXO “D” MANUAL DE USUARIO



En el lado superior derecho de la lista de funcionarios existe una opción de nueva persona, esta opción se debe utilizar si el funcionario que está realizando el pedido no existe en la base de datos como nueva persona.



En la parte superior derecha del formulario de registrar nueva persona existe una opción Ver lista de Personas, esta opción muestra las personas que han sido registrados recientemente, para que se pueda registrar como funcionario click en la opción registrar que está en la lista personas.

## ANEXO “D” MANUAL DE USUARIO



En este formulario ingresamos datos adicionales para la persona por ejemplo, la unidad a la que pertenece, también el cargo que ocupa más el teléfono de la unidad.



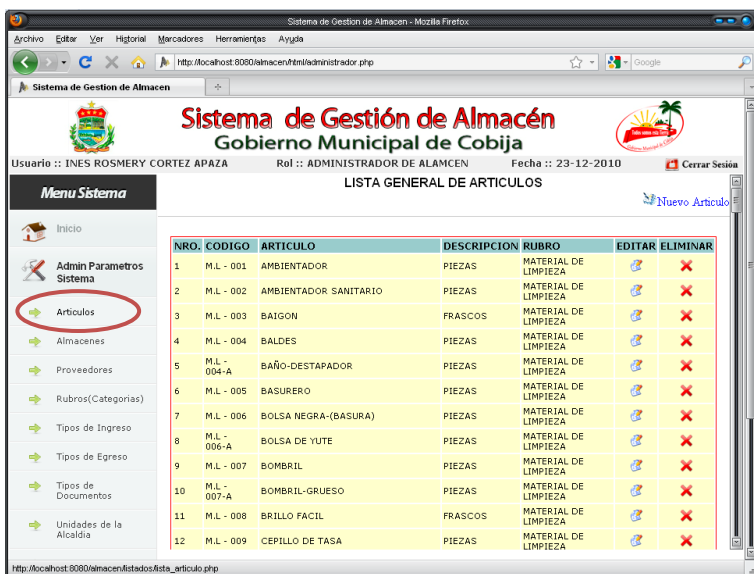
Como se puede observar ya se encuentra registrado como funcionario con los datos adicionales que se le agrego.

De este modo administramos funcionarios del gobierno municipal de cobija, en el siguiente apartado veremos cómo administrar los parámetros del sistema.

## 4. ADMINISTRAR PARAMETROS DEL SISTEMA

### 4.1. ADMINISTRAR ARTICULOS.

Administrar Artículos permite crear, editar, eliminar los diferentes tipos de artículos que existen en la unidad de almacén para eso mostramos los procedimientos



Al hacer click sobre artículos obtenemos la lista completa los artículos que están registrados en la base de datos clasificados por sus respectivos rubros, también podemos editar y eliminar los diferentes artículos.

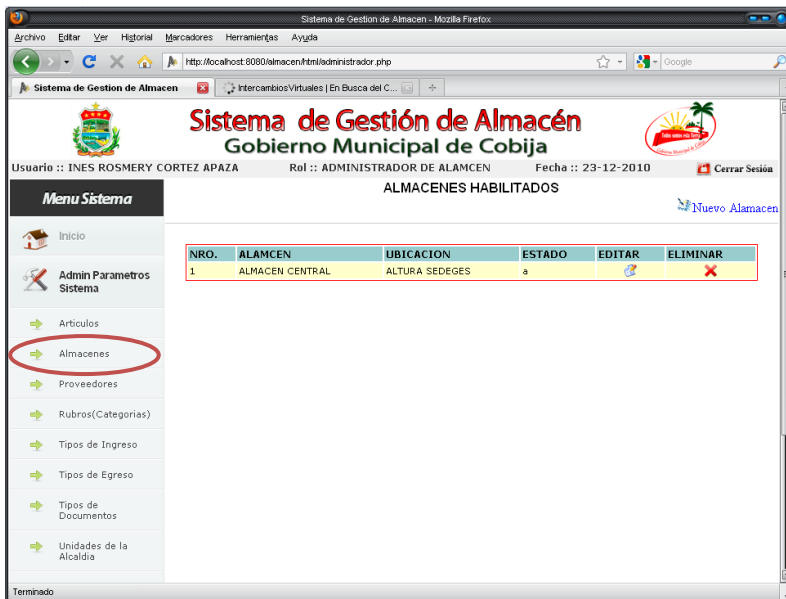


En la parte superior derecha de la anterior figura existe la opción de Nuevo Artículo, por medio de este formulario se puede crear nuevo artículo especificando al rubro que pertenece, el código del artículo, nombre del artículo y la unidad de medida.

# ANEXO “D” MANUAL DE USUARIO

## 4.2. ADMINSTRAR ALMACENES

Administrar almacenes es un sub-modulo que permite crear los sub almacenes que son parte de la Gobierno Municipal de Cobija.



Al hacer click sobre Almacenes, aparece la lista completa de almacenes que están activos y funcionando, al hacer click en la opción nuevo almacén se va a la siguiente pantalla.

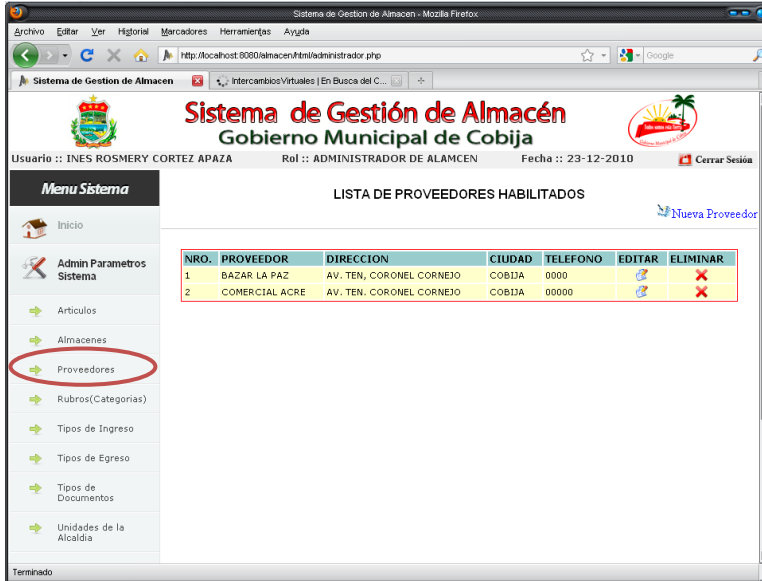


En este formulario se puede crear nuevo almacén con su respectiva ubicación geográfica.

# ANEXO “D” MANUAL DE USUARIO

## 4.3. ADMINSTRAR PROVEEDORES

Administrar proveedores, permite crear, editar, eliminar los proveedores del cual se adquiere diferentes artículos, explicaremos el procedimiento.



Muestra la lista completa de los proveedores registrados en el sistema, el cual también se puede editar su información y eliminar al proveedor.



Al hacer click en la parte superior derecha en la opción Nuevo Proveedor de la figura anterior muestra el formulario registrar nuevo proveedor que permite adicionar un nuevo proveedor que no se encuentra en la base de datos.

# ANEXO “D” MANUAL DE USUARIO

## 4.4.ADMINISTRAR RUBROS (CLASIFICACION)

El sub modulo administrar rubros crea los diferentes rubros las que clasifica a cada artículo que pertenece a dicho rubro, a continuación se muestra los procedimientos.



Al hacer click en Rubros se observa la lista completa de los rubros que están habilitados.

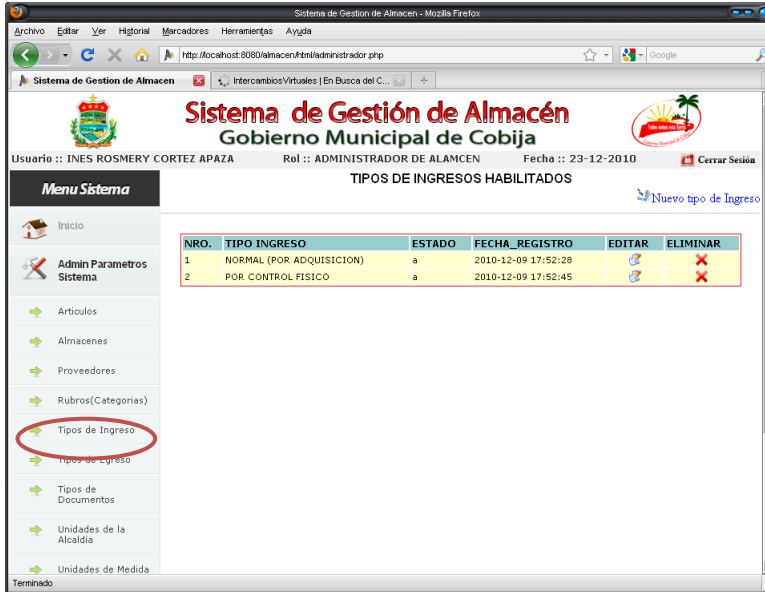


Al hacer click en la parte superior derecha en la opción que dice Nuevo Rubro, se puede crear o agregar nuevos rubros que todavía faltan.

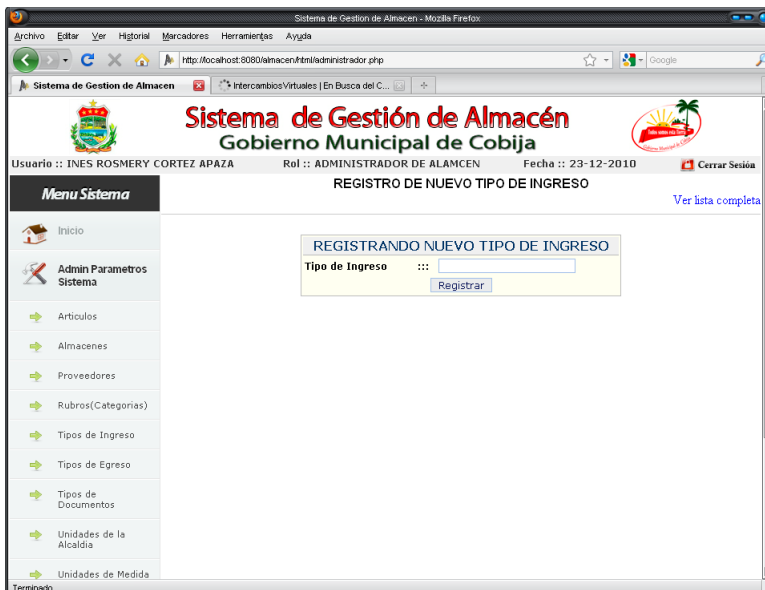
# ANEXO “D” MANUAL DE USUARIO

## 4.5.ADMINISTRAR TIPOS DE INGRESO

Administrar Tipos de Ingreso permite crear nuevos tipos de ingreso que esto se utiliza para el formulario de ingreso de artículos.



Lista completa de los tipos de ingreso que existen

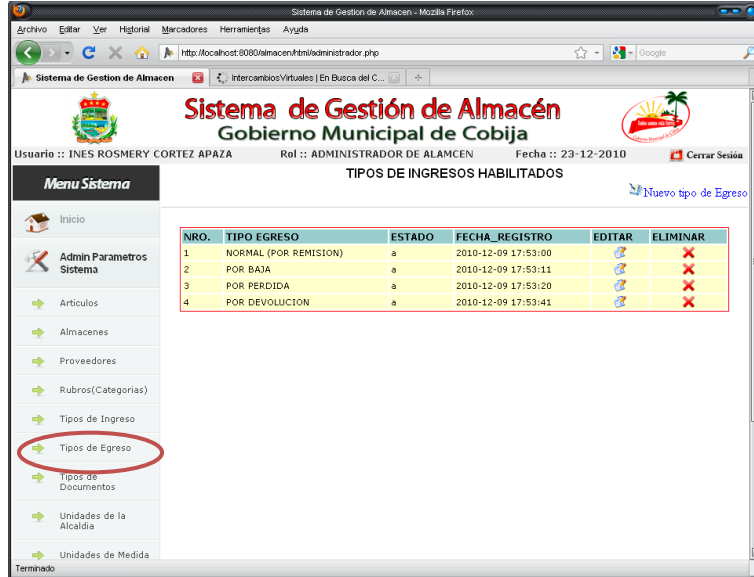


Formulario que permite crear un nuevo tipo de ingreso

# ANEXO “D” MANUAL DE USUARIO

## 4.6.ADMINISTRAR TIPOS DE EGRESO

Administrar Tipos de Egreso permite crear nuevos tipos de egreso que esto se utiliza para el formulario de egreso de artículos (salida).



Muestra la lista completa de los tipos de egreso que existen y están habilitados para la salida de artículos

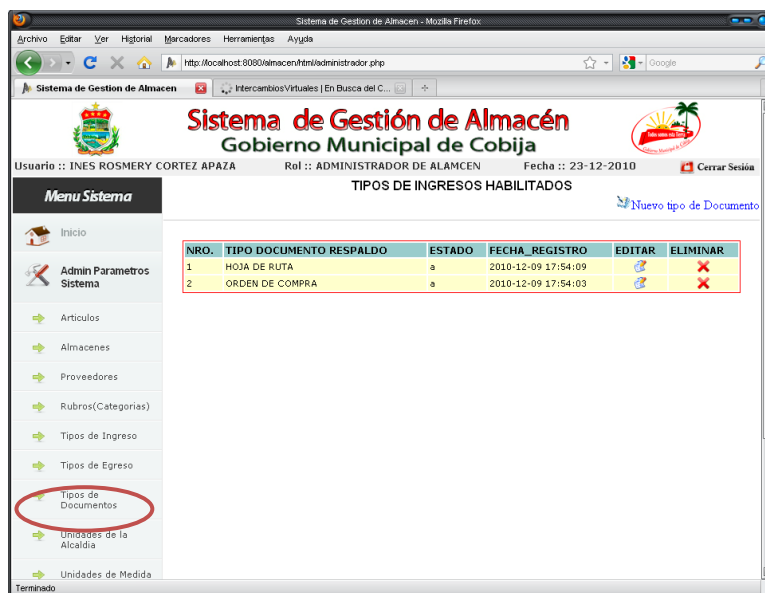


Este formulario permite crear nuevos Tipos de Egreso.

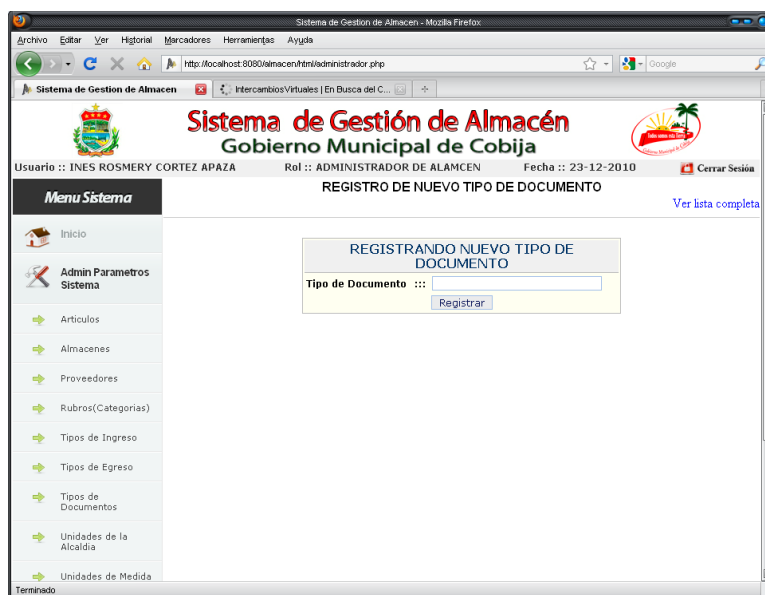
## ANEXO “D” MANUAL DE USUARIO

### 4.7.ADMINISTRAR TIPOS DE DOCUMENTO

Administrar tipos de documento también permite crear nuevos tipos de documento



Muestra la lista de tipos de documentos que existen y que están habilitando en el sistema



Con este formulario se puede crear nuevo tipo de documento.

## ANEXO “D” MANUAL DE USUARIO

### 4.8.ADMINISTRAR UNIDADES DE LA ALCALDIA

Administrar Unidades es un sub modulo que permite crear las dirección, las unidades y todas las oficinas que dependen del Gobierno Municipal de Cobija



Se puede observar la lista completa de las direcciones, unidades de la Alcaldía.

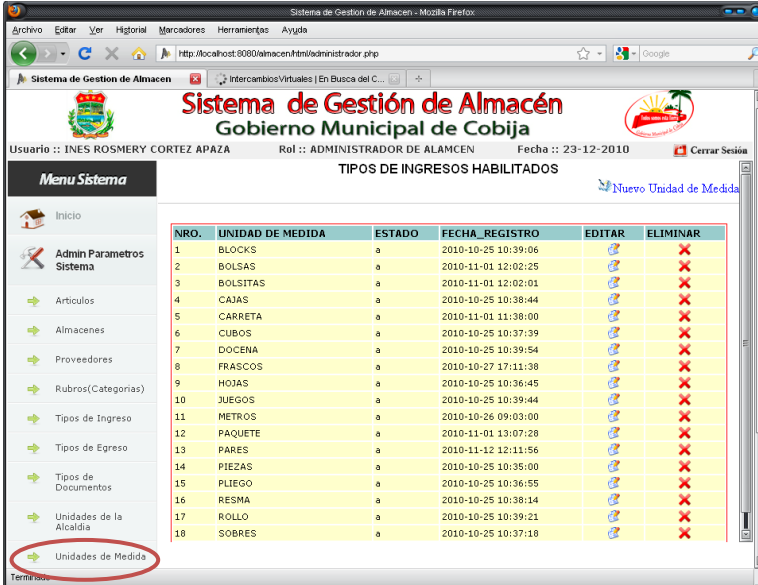


Con este formulario se puede registrar nuevas unidades que sean parte del Gobierno Municipal de Cobija.

## ANEXO “D” MANUAL DE USUARIO

### 4.9.ADMINISTRAR UNIDADES DE MEDIDA


Este sub modulo permite crear nuevas unidades de medida que también es utilizado por el sistema.



The screenshot shows the 'Sistema de Gestión de Almacén' web application. The page title is 'TIPOS DE INGRESOS HABILITADOS'. The user is 'INES ROSMERY CORTEZ APAZA' with the role 'ADMINISTRADOR DE ALAMCEN' on '23-12-2010'. The page displays a table of registered units of measure with the following data:

NRO.	UNIDAD DE MEDIDA	ESTADO	FECHA_REGISTRO	EDITAR	ELIMINAR
1	BLOCKS	a	2010-10-25 10:39:06		
2	BOLSAS	a	2010-11-01 12:02:25		
3	BOLSITAS	a	2010-11-01 12:02:01		
4	CAJAS	a	2010-10-25 10:38:44		
5	CARRETA	a	2010-11-01 11:38:00		
6	CUBOS	a	2010-10-25 10:37:39		
7	DOCENA	a	2010-10-25 10:39:54		
8	FRASCOS	a	2010-10-27 17:11:38		
9	HOJAS	a	2010-10-25 10:36:45		
10	JUEGOS	a	2010-10-25 10:39:44		
11	METROS	a	2010-10-26 09:03:00		
12	PAQUETE	a	2010-11-01 13:07:28		
13	PARES	a	2010-11-12 12:11:56		
14	PIEZAS	a	2010-10-25 10:35:00		
15	PLIEGO	a	2010-10-25 10:36:55		
16	RESMA	a	2010-10-25 10:38:14		
17	ROLLO	a	2010-10-25 10:39:21		
18	SOBRES	a	2010-10-25 10:37:18		

Se muestra la lista completa de las unidades de medida que están registradas en el sistema



The screenshot shows the 'Sistema de Gestión de Almacén' web application. The page title is 'REGISTRO NUEVA UNIDAD DE MEDIDA'. The user is 'INES ROSMERY CORTEZ APAZA' with the role 'ADMINISTRADOR DE ALAMCEN' on '23-12-2010'. The page displays a form for registering a new unit of measure:

REGISTRANDO NUEVA UNIDAD DE MEDIDA

Unidad de Medida :

Se puede crear nuevas unidades de medida.

Todas la opciones que se explico sirven para que el sistema funcione a un 100% si algún dato falta se debe acudir a estas opciones por ejemplo cuando se está realizando un nuevo entrega de material y no existe la unidad y el encargado a quien entregar la nota de salida se

## ANEXO “D” MANUAL DE USUARIO

debe crear la unidad y el funcionario asignando a la unidad que pertenece, entonces estos datos son muy importantes.

### 5. ADMINISTRACION DE ALMACEN

#### 5.1. NOTA DE ENTRADA

La nota de entrada o ingreso se debe realizar cuando se está haciendo una nueva adquisición de material o artículo, esto debe ser registrado debidamente ordenado como se muestra a continuación.

Sistema de Gestión de Almacén - Mozilla Firefox

http://localhost:8080/almacen/html/administrador.php

Sistema de Gestión de Almacén

Sistema de Gestión de Almacén  
Gobierno Municipal de Cobija

Usuario :: INES ROSMERY CORTEZ APAZA Rol :: ADMINISTRADOR DE ALMACEN Fecha :: 23-12-2010 Cerrar Sesión

Menu Sistema AGREGANDO NUEVA NOTA DE ENTRADA

Inicio

Admin Parametros Sistema

Admin Personal

Administracion de Almacen

Nota de Entrada

Nota de Salida

Inventario Manual

Inventario Automatico

Cierre de Gestion

Reportes

Terminado

INTRODUZCA LOS DATOS

Tipo Ingreso :: --Seleccione--

Tipo de Documento :: --Seleccione--

Proveedor :: --Seleccione--

Almacen :: --Seleccione--

Fecha Ingreso :: 2010-12-23 (yy-mm-dd)

Numero Documento ::

Nro. Comprobante de Pago ::

Detalle ::

U.F.V. ::

Siguiete

Para elaborar la nueva nota de entrada haga click en la opción Nota de Entrada, en el cual se debe llenar todos los campos que tiene el formulario un dato importante es llenar la U.F.V. del día.

Sistema de Gestión de Almacén - Mozilla Firefox

http://localhost:8080/almacen/html/administrador.php

Sistema de Gestión de Almacén

Sistema de Gestión de Almacén  
Gobierno Municipal de Cobija

Usuario :: INES ROSMERY CORTEZ APAZA Rol :: ADMINISTRADOR DE ALMACEN Fecha :: 23-12-2010 Cerrar Sesión

Menu Sistema AGREGANDO NUEVA NOTA DE ENTRADA

Inicio

Admin Parametros Sistema

Admin Personal

Administracion de Almacen

Nota de Entrada

Nota de Salida

Inventario Manual

Inventario Automatico

Cierre de Gestion

Reportes

Terminado

Tipo Ingreso :: NORMAL (POR ADQUISICION)

Fecha Ingreso :: 2010-12-23

Detalle :: COMPRA DE MATERIALES

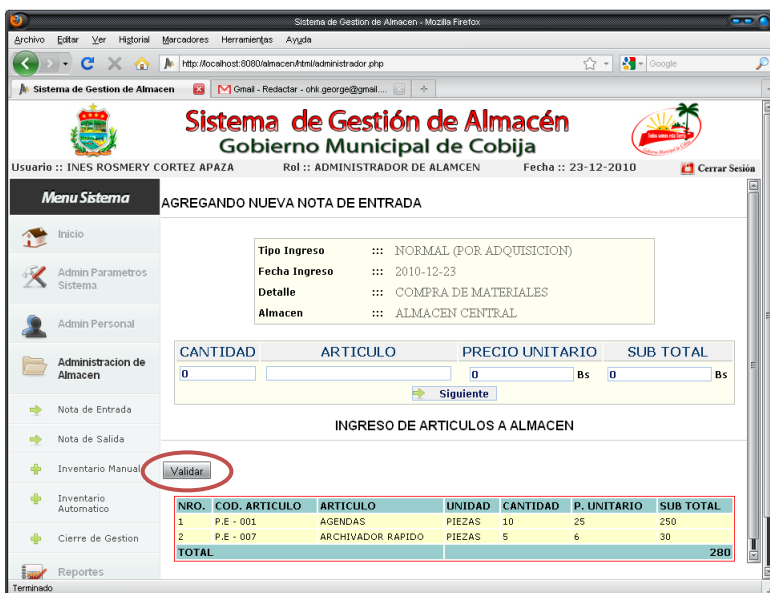
Almacen :: ALMACEN CENTRAL

CANTIDAD	ARTICULO	PRECIO UNITARIO	SUB TOTAL
0		0 Bs	Bs

Siguiete

Después de llenar todos los campos del anterior formulario, se pasa al siguiente formulario donde se ingresa la cantidad, artículo, precio unitario del artículo que está ingresando y click en el botón siguiente y se puede seguir agregando mas artículos.

# ANEXO “D” MANUAL DE USUARIO



Después de ingresar los datos click en el botón siguiente inmediatamente en la parte inferior aparece en una tabla los datos ingresados, después de haber finalizado se debe hacer click en el botón validar para que se valide los datos.



Luego de validar las entradas se puede observar la lista de entradas que se realizaron, en la lista tiene la opción de Ver hacemos click en ver.

## ANEXO “D” MANUAL DE USUARIO

Nota de Ingreso a Almacenes - Mozilla Firefox

http://localhost:8080/almacen/reportes/nota\_de\_ingreso.php?id\_ingreso=1

Sistema de Gestion de Almacenes

**NOTA DE INGRESO A ALMACEN**  
**SISTEMA DE GESTION DE ALMACEN**  
**GOBIERNO MUNICIPAL DE COBIJA**

TIPO DOCUMENTO	::: ORDEN DE COMPRA	NRO. ORDEN DE COMPRA	::: A-004
TIPO INGRESO	::: NORMAL (POR ADQUISICION)	NRO. COMPROBANTE DE PAGO	::: A-004
PROVEEDOR	::: COMERCIAL ACRE	FECHA INGRESO	::: 2010-12-23
NIT PROVEEDOR	::: --	GESTION	::: 2010
ALMACEN	::: ALMACEN CENTRAL	USUARIO	::: admin
DETALLE	::: COMPRA DE MATERIALES		

NRO.	RUBRO	COD. ARTI.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL
1	PAPEL DE ESCRITORIO	P.E - 001	AGENDAS	10	25	250
2	PAPEL DE ESCRITORIO	P.E - 007	ARCHIVADOR RAPIDO	5	6	30
<b>SUMA TOTAL</b>						<b>280</b>

RESPONSABLE DE ALMACEN  
RECIBI CONFORME

PROVEEDOR  
ENTREGO CONFORME

Terminado

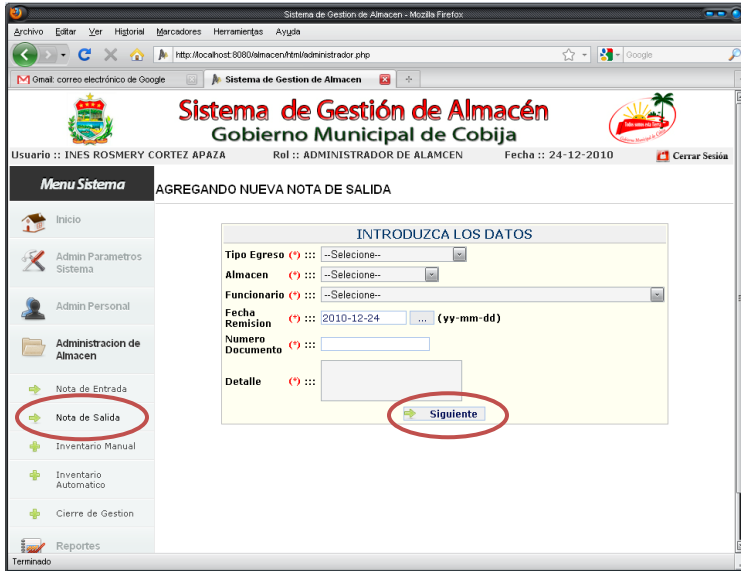
Obtenemos la vista de la nota de entrada listo para imprimir, para imprimir haga click en el logotipo del G.M.C. que está en la parte superior derecha, al momento de imprimir puede sacar dos copias.

Con esto terminamos la parte de realizar la nota de entrada a almacén.

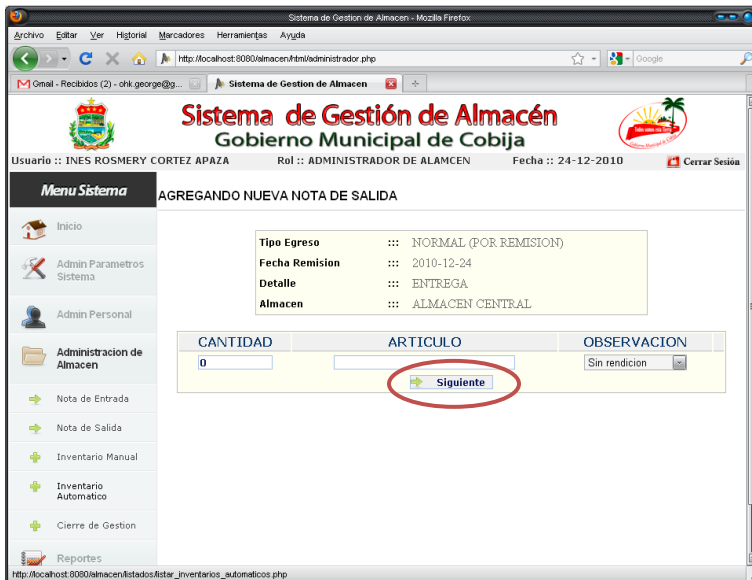
# ANEXO “D” MANUAL DE USUARIO

## 5.2. NOTA DE SALIDA

Las notas de salida o entrega se deben de realizar cuando existe una solicitud de productos o artículos, se debe seguir los siguientes procedimientos.

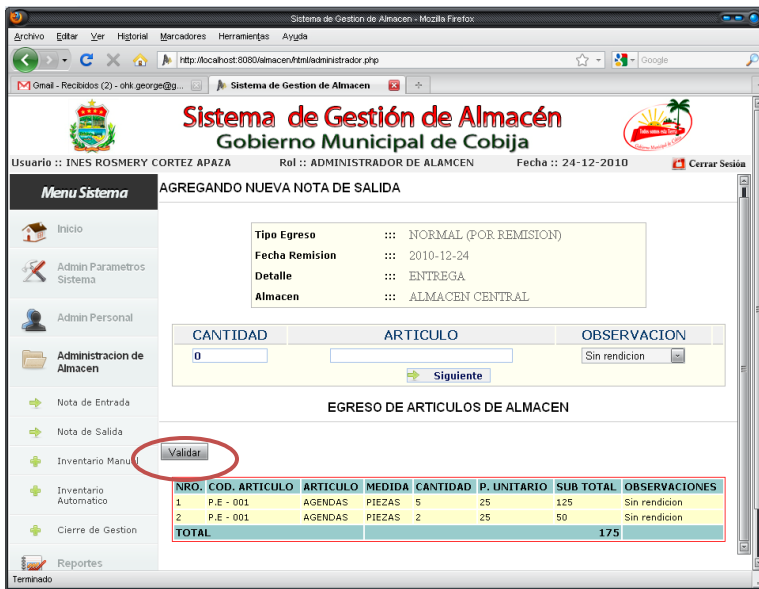


Al hacer click en la opción Nota de Salida, aparece el formulario de nota de salida, donde se debe llenar todos los datos requeridos y presionar click en el botón siguiente para pasar a la siguiente formulario.



En esta segunda parte ingresamos la cantidad y el artículo que se va entregar el campo observación se utiliza en casos especiales por ejemplo. En el caso de losetas que debe de estar sujeto a rendición y hacemos click en el botón siguiente para luego seguir despachando mas artículos.

# ANEXO “D” MANUAL DE USUARIO



Después de ingresar los datos en la parte inferior aparece la lista de artículos que estamos despachando una vez terminado hacemos click sobre el botón Validar.



Inmediatamente nos muestra la lista de salidas que se realizador el primero en la lista es el ultimo despacho que se realizó, para imprimir dicho nota de salida click en vista impresion que está en la misma lista.



## ANEXO “D” MANUAL DE USUARIO

Nota de Salida de Almacén - Mozilla Firefox

http://localhost:8080/almacen/reportes/nota\_de\_egreso.php?id\_egreso=1

Sistema de Gestion de Almacen

Nota de Salida de Almacenes

 **NOTA DE SALIDA DE ALMACEN**   
**SISTEMA DE GESTION DE ALMACEN**  
**GOBIERNO MUNICIPAL DE COBIJA**

<b>NRO DOCUMENTO</b>	::: A-004	<b>ALMACEN</b>	::: ALMACEN CENTRAL
<b>TIPO SALIDA</b>	::: NORMAL (POR REMISION)	<b>FECHA SALIDA</b>	::: 2010-12-24
<b>FUNCIONARIO</b>	::: JULIA VEGA GUAYGUA	<b>GESTION</b>	::: 2010
<b>UNIDAD G.M.C.</b>	::: DIRECCION DE EDUCACION	<b>USUARIO</b>	::: admin
<b>DETALLE</b>	::: ENTREGA		

NRO.	RUBRO	COD. ARTI.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL	OBSERVACION
1	PAPEL DE ESCRITORIO	P.E - 001	AGENDAS	5	25	125	Sin rendicion
2	PAPEL DE ESCRITORIO	P.E - 001	AGENDAS	2	25	50	Sin rendicion
<b>SUMA TOTAL</b>						<b>175</b>	

RESPONSABLE DE ALMACEN  
ENTREGA CONFORME

FUNCIONARIO G.M.C.  
RECIBI CONFORME

Terminado

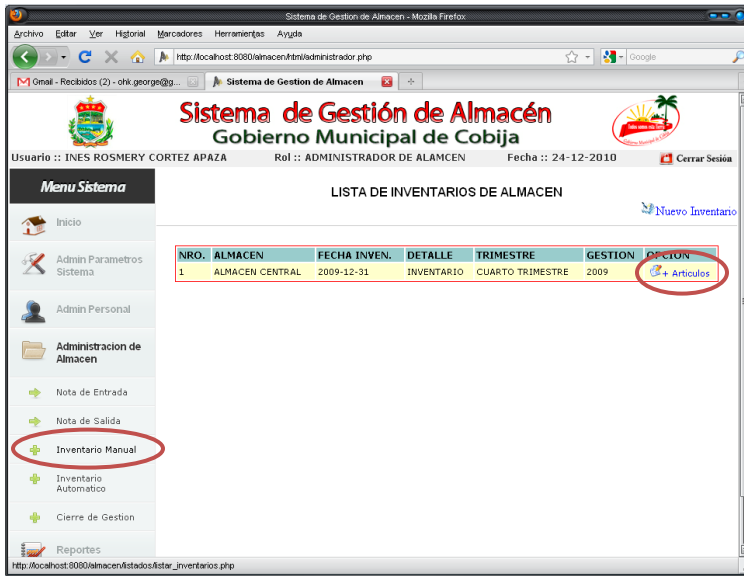
Obtenemos la nota de salida listo para imprimir con todos los datos necesarios, para imprimir haga click en el logotipo del G.M.C. que está en la parte superior derecha, al momento de imprimir puede sacar dos copias.

Con esto terminamos la parte de realizar la nota de salida de almacén.

# ANEXO “D” MANUAL DE USUARIO

## 5.3. INVENTARIO MANUAL

La opción de Inventario Manual se agregó con la finalidad ingresar datos del inventario que se tiene impreso de la gestión 2010, para poder empezar con nuevos datos en el sistema, porque si no existe un inventario de la gestión 2010 del cuarto trimestre específicamente y no se haya hecho un cierre de gestión de la gestión 2010 no se puede ingresar datos para la gestión 2011 por esta razón se implemento esta opción del inventario manual.

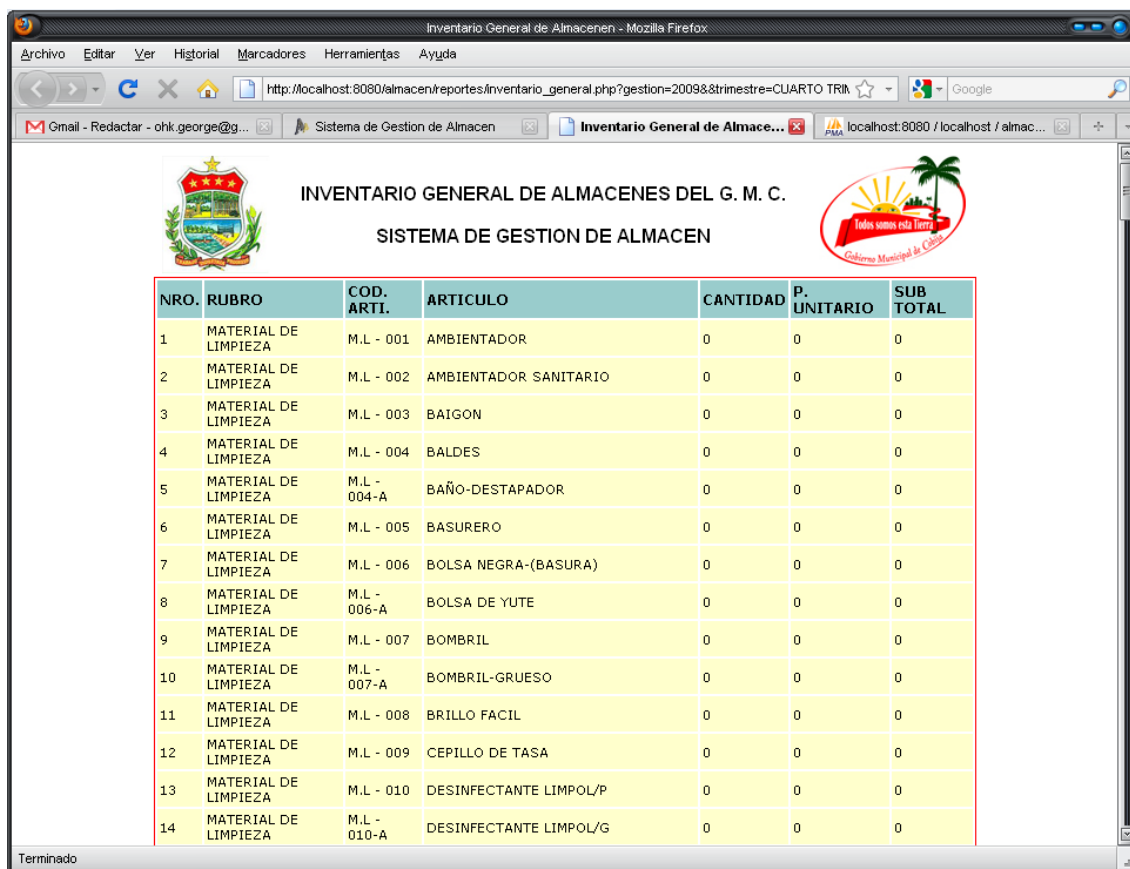


Al hacer click en la opción Inventario Manual obtenemos la lista de inventarios que se realizó, para agregar artículos manualmente click en la opción +artículos.



En este formulario ingresamos los datos que se realizó la anterior gestión 2010, los datos necesarios son, el artículo, la cantidad, precio del artículo y para guardar click en el botón Guardar, para ver la vista impresión click en Ver Resumen de Inventario en la parte superior derecha.

## ANEXO “D” MANUAL DE USUARIO



INVENTARIO GENERAL DE ALMACENES DEL G. M. C.  
SISTEMA DE GESTION DE ALMACEN

NRO.	RUBRO	COD. ARTL.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL
1	MATERIAL DE LIMPIEZA	M.L - 001	AMBIENTADOR	0	0	0
2	MATERIAL DE LIMPIEZA	M.L - 002	AMBIENTADOR SANITARIO	0	0	0
3	MATERIAL DE LIMPIEZA	M.L - 003	BAIGON	0	0	0
4	MATERIAL DE LIMPIEZA	M.L - 004	BALDES	0	0	0
5	MATERIAL DE LIMPIEZA	M.L - 004-A	BAÑO-DESTAPADOR	0	0	0
6	MATERIAL DE LIMPIEZA	M.L - 005	BASURERO	0	0	0
7	MATERIAL DE LIMPIEZA	M.L - 006	BOLSA NEGRA-(BASURA)	0	0	0
8	MATERIAL DE LIMPIEZA	M.L - 006-A	BOLSA DE YUTE	0	0	0
9	MATERIAL DE LIMPIEZA	M.L - 007	BOMBRIL	0	0	0
10	MATERIAL DE LIMPIEZA	M.L - 007-A	BOMBRIL-GRUESO	0	0	0
11	MATERIAL DE LIMPIEZA	M.L - 008	BRILLO FACIL	0	0	0
12	MATERIAL DE LIMPIEZA	M.L - 009	CEPILLO DE TASA	0	0	0
13	MATERIAL DE LIMPIEZA	M.L - 010	DESINFECTANTE LIMPOL/P	0	0	0
14	MATERIAL DE LIMPIEZA	M.L - 010-A	DESINFECTANTE LIMPOL/G	0	0	0

Terminado

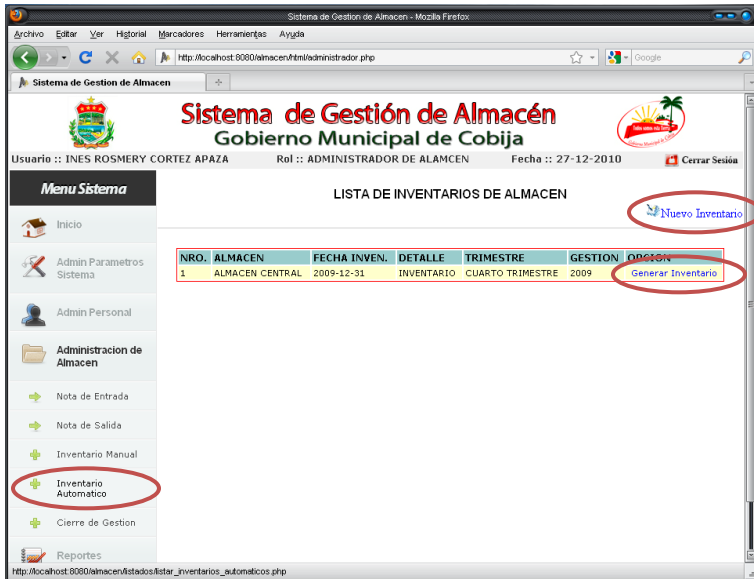
Con esto terminamos la parte de Inventario manual, esto solo se debe utilizar por primera vez y última vez cuando el sistema se implemente por primera vez.

Con esto terminamos la parte de Inventario Manual.

# ANEXO “D” MANUAL DE USUARIO

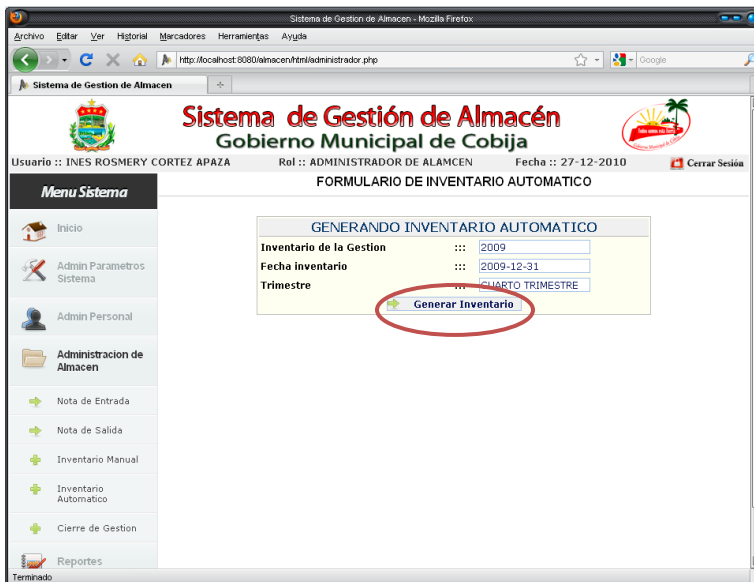
## 5.4. INVENTARIO AUTOMATICO

El inventario automático está habilitado, para generar inventario trimestral automático siempre y cuando se haya ingresado todos los ingresos y salidas al sistema, porque una vez realizado el inventario ya no se puede ingresar las notas de ingresos y salidas con fecha antes del inventario, por esta razón es necesario tiene que completarse todas la notas.



Al hacer click sobre la opción Inventario Automático, en la parte derecha muestra la lista completa de inventarios realizados según el año y el trimestre, en el campo donde dice OPCION se puede observar dos opciones uno generar inventario y vista impresión y en la parte superior derecha dice Nuevo Inventario que permite crear un nuevo inventario.

Generar Inventario se observa cuando se crea una un nuevo inventario el cual no tiene datos actuales registrados y la Opción Vista Impresión se muestra cuando ya el inventario fue generado.



Este formulario se muestra cuando se hace click sobre la opción generar inventario, en este formulario muestra los datos generales con que se va generar el inventario para generar has click en el botón Generar Inventario.

## ANEXO “D” MANUAL DE USUARIO

Después muestra el inventario trimestral, en la parte inferior revisamos los datos y se debe validar los datos haciendo click en el botón validar inventario, tal como se muestra en la imagen.

**Sistema de Gestión de Almacén**  
Gobierno Municipal de Cobija

Usuario :: INES ROSMERY CORTEZ APAZA    Rol :: ADMINISTRADOR DE ALAMCEN    Fecha :: 27-12-2010    Cerrar Sesión

**Menu Sistema**

- Inicio
- Admin Parametros Sistema
- Admin Personal
- Administracion de Almacen
- Reportes

**FORMULARIO DE INVENTARIO AUTOMÁTICO**

**GENERANDO INVENTARIO AUTOMÁTICO**

Inventario de la Gestión ::: 2009  
Fecha inventario ::: 2009-12-31  
Trimestre ::: CUARTO TRIMESTRE

[Generar Inventario](#)

**INVENTARIO GENERAL DE ALMACENES DEL G.M.C.**

NRO.	RUBRO	COD. ARTI.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL
1	MATERIAL DE LIMPIEZA	M.L - 001	AMBIENTADOR	0	0	0
2	MATERIAL DE LIMPIEZA	M.L - 002	AMBIENTADOR SANITARIO	0	0	0
3	MATERIAL DE LIMPIEZA	M.L - 003	BAIGON	0	0	0
4	MATERIAL DE LIMPIEZA	M.L - 004	BALDES	0	0	0
5	MATERIAL DE LIMPIEZA	M.L - 005	BALDES	0	0	0

Validar Inventario

## ANEXO “D” MANUAL DE USUARIO

Para imprimir el inventario se debe hacer click sobre el logotipo del G-A-M-C.

NRO.	RUBRO	COD. ARTIL.	ARTICULO	CANTIDAD	P. UNITARIO	SUB TOTAL
1	MATERIAL DE LIMPIEZA	M.L - 001	AMBIENTADOR	0	0	0
2	MATERIAL DE LIMPIEZA	M.L - 002	AMBIENTADOR SANITARIO	0	0	0
3	MATERIAL DE LIMPIEZA	M.L - 003	BAIGON	0	0	0
4	MATERIAL DE LIMPIEZA	M.L - 004	BALDES	0	0	0
5	MATERIAL DE LIMPIEZA	M.L - 004-A	BAÑO-DESTAPADOR	0	0	0
6	MATERIAL DE LIMPIEZA	M.L - 005	BASURERO	0	0	0
7	MATERIAL DE LIMPIEZA	M.L - 006	BOLSA NEGRA-(BASURA)	0	0	0
8	MATERIAL DE LIMPIEZA	M.L - 006-A	BOLSA DE YUTE	0	0	0
9	MATERIAL DE LIMPIEZA	M.L - 007	BOMBRIL	0	0	0
10	MATERIAL DE LIMPIEZA	M.L - 007-A	BOMBRIL-GRUESO	0	0	0
11	MATERIAL DE LIMPIEZA	M.L - 008	BRILLO FACIL	0	0	0
12	MATERIAL DE LIMPIEZA	M.L - 009	CEPILLO DE TASA	0	0	0
13	MATERIAL DE LIMPIEZA	M.L - 010	DESINFECTANTE LIMPOL/P	0	0	0

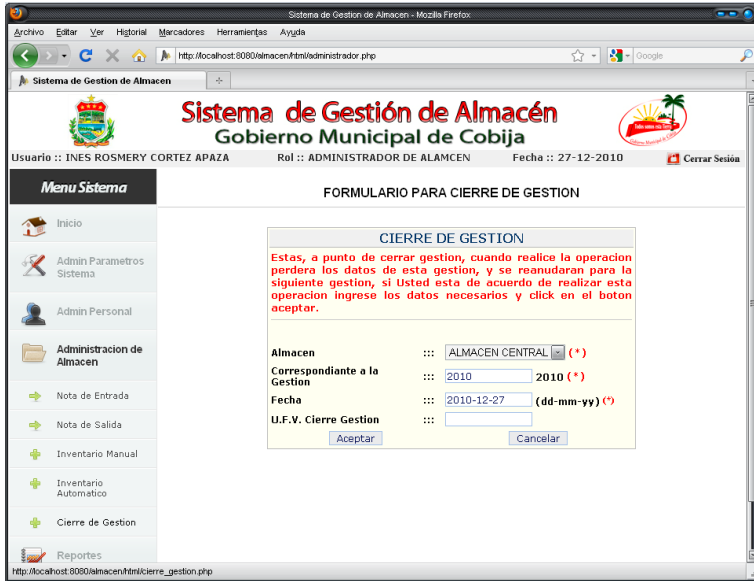
Con esto terminamos la parte del inventario automático que es muy importante pero hay que tener mucho cuidado en aplicarla que un error puede hacer variar los resultados a final de la gestión.

En el siguiente apartado veremos el cierre de gestión.

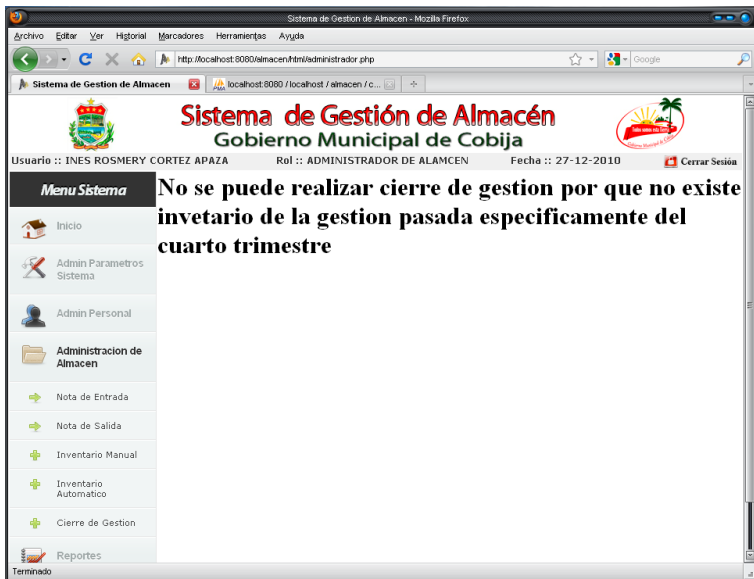
## ANEXO “D” MANUAL DE USUARIO

### 5.5. CIERRE DE GESTION

Esta opción está habilitada específicamente para cerrar gestión con datos existentes y precios actualizados para la siguiente gestión a continuación explicamos paso a paso las opciones.



Al hacer click en la opción Cierre Gestión nos muestra el formulario con los datos generales, el año es el actual la fecha actual y un campo que pide la UVF del día estos valores se pueden cambiar el dato más importante es el UFV del día.



Este mensaje sale cuando no existe un inventario de la gestión que se quiere hacer cierre de gestión específicamente del cuarto trimestre porque ahí está la información actualizada para hacer una duplicidad para la siguiente gestión.

Entonces se realiza el inventario de la gestión que se quiere hacer el cierre de gestión se puede elegir el inventario manual o automático dependiendo a las circunstancias.

## ANEXO “D” MANUAL DE USUARIO

Después de generar el inventario ya se puede cerrar gestión.

**Sistema de Gestión de Almacén**  
Gobierno Municipal de Cobija

Usuario :: INES ROSMERY CORTEZ APAZA    Rol :: ADMINISTRADOR DE ALMACEN    Fecha :: 27-12-2010    Cerrar Sesión

**CIERRE DE GESTION**

A continuación le mostramos una lista detallada de los artículos que existen en : ALMACEN CENTRAL.....de la GESTION : 2010 ...si Usted esta de acuerdo con estos datos y que estos mismos se actualicen para la siguiente gestión a click en Aceptar.....: Para imprimir este reporte haga click en el boton Imprimir, despues de este paso ya no podra ver el mismo reporte aproveche a sacar dos copias de impresion.

Imprimir    Cancelar

CODIGO RUBRO ::: M.L			RUBRO (CLASIFICACION) ::: MATERIAL DE LIMPIEZA					
NRO.	COD. ARTICULO	ARTICULO	MEDIDA	CANTIDAD	PRECIO	SUB TOTAL	PRECIO ACTUALIZADO	SALDO ACTUALIZADO
1	M.L - 001	AMBIENTADOR	PIEZAS	0	0	0	0	0
2	M.L - 002	AMBIENTADOR SANITARIO	PIEZAS	0	0	0	0	0
3	M.L - 003	BAIGÓN	FRASCOS	0	0	0	0	0
4	M.L - 004	BALDES	PIEZAS	0	0	0	0	0
5	M.L - 004-A	BAÑO-DESTAPADOR	PIEZAS	0	0	0	0	0
6	M.L - 005	BASURERO	PIEZAS	0	0	0	0	0
7	M.L - 006	BOLSA NEGRA-(BASURA)	PIEZAS	0	0	0	0	0
8	M.L - 006-A	BOLSA DE YUTE	PIEZAS	0	0	0	0	0
9	M.L - 007	BOMBRIL	PIEZAS	0	0	0	0	0

En el lado derecho se puede observar la lista detallada de los artículos con la cantidad, precio actual y saldo actual más el precio actualizado y saldos actualizados.

Este precio actualizado se replica para la siguiente gestión, si se hace click en el botón

**Sistema de Gestión de Almacén**  
Gobierno Municipal de Cobija

Usuario :: INES ROSMERY CORTEZ APAZA    Rol :: ADMINISTRADOR DE ALMACEN    Fecha :: 27-12-2010    Cerrar Sesión

**Se termino de cerrar gestion ahora cuenta con nuevos datos para la Gestion : 2011**

aceptar, también se puede observar la vista impresión.

## ANEXO “D” MANUAL DE USUARIO

Esto se muestra cuando se hace el cierre de gestión y se cuenta con  
Datos actualizados para la siguiente gestión

**REPORTE DEL CIERRE DE GESTION 2010**  
**SISTEMA DE GESTION DE ALMACEN**  
**GOBIERNO MUNICIPAL DE COBIJA**

CODIGO RUBRO ::: M.L			RUBRO (CLASIFICACION) ::: MATERIAL DE LIMPIEZA			
NRO.	COD.	ARTICULO	MEDIDA	CANTIDAD	PRECIO ACTUALIZADO	SALDO ACTUALIZADO
1	M.L - 001	AMBIENTADOR	PIEZAS	0	0	0
2	M.L - 002	AMBIENTADOR SANITARIO	PIEZAS	0	0	0
3	M.L - 003	BAIGON	FRASCOS	0	0	0
4	M.L - 004	BALDES	PIEZAS	0	0	0
5	M.L - 004-A	BAÑO-DESTAPADOR	PIEZAS	0	0	0
6	M.L - 005	BASURERO	PIEZAS	0	0	0
7	M.L - 006	BOLSA NEGRA-(BASURA)	PIEZAS	0	0	0
8	M.L - 006-A	BOLSA DE YUTE	PIEZAS	0	0	0
9	M.L - 007	BOMBRIL	PIEZAS	0	0	0
10	M.L - 007-A	BOMBRIL-GRUESO	PIEZAS	0	0	0
11	M.L - 008	BRILLO FACIL	FRASCOS	0	0	0
12	M.L - 009	CEPILLO DE TASA	PIEZAS	0	0	0
13	M.L - 010	DESINFECTANTE LIMPOL/P	FRASCOS	0	0	0
14	M.L - 010-A	DESINFECTANTE LIMPOL/G	FRASCOS	0	0	0
15	M.L - 011	ESCOBAS	PIEZAS	0	0	0
16	M.L - 011-A	ESCOBILLA DE MANO	PIEZAS	0	0	0
17	M.L - 012	ESCOBONES	PIEZAS	0	0	0
18	M.L - 013	ESPONJA	PIEZAS	0	0	0
19	M.L - 013-A	ENCENDEDOR DE COCINA	PIEZAS	0	0	0
20	M.L - 013-B	FOSFORO	CAJAS	0	0	0
21	M.L - 014	GOMA ESCURRIDORA DE AGUA	PIEZAS	0	0	0
<b> ::: SUMA TOTAL :::</b>					<b>0 Bs.</b>	<b>0 Bs.</b>
CODIGO RUBRO ::: P.E			RUBRO (CLASIFICACION) ::: PAPEL DE ESCRITORIO			
NRO.	COD.	ARTICULO	MEDIDA	CANTIDAD	PRECIO ACTUALIZADO	SALDO ACTUALIZADO
1	P.E - 001	AGENDAS	PIEZAS	3	27.36	82.08

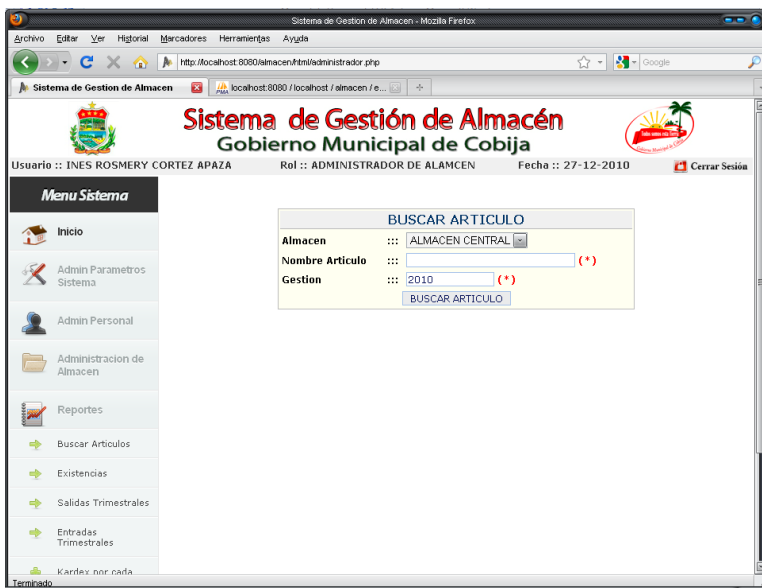
Vista impresión del cierre de gestión.

Con esto terminamos la parte de cierre de gestión, en el siguiente apartado veremos otro modulo que es parte del sistema gestión de almacén.

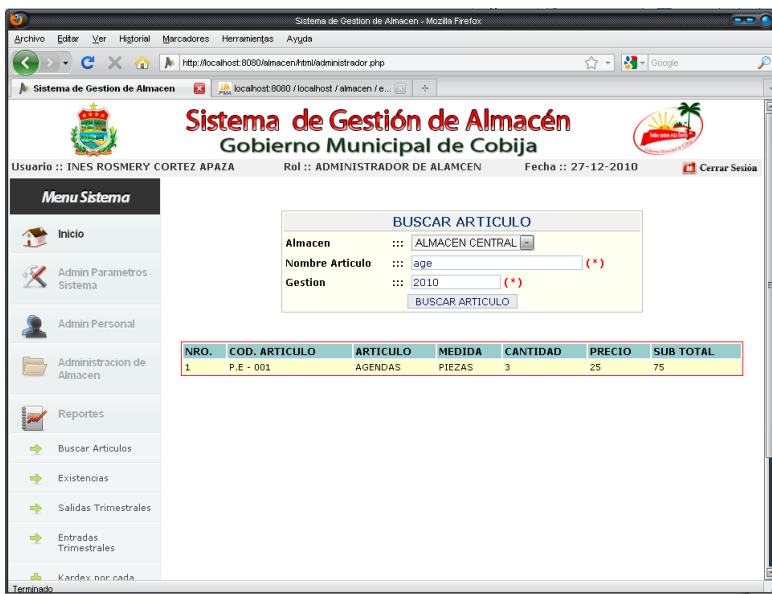
## 6. REPORTES

### 6.1. BUSCAR ARTÍCULO

Buscar artículo permite el artículo determinando el almacén y la gestión y el nombre del artículo.



En el formulario buscar artículo es necesario definir el almacén en nombre del artículo pero no es necesario que sea completo el nombre puede ser las iniciales además es necesario especificar el año.



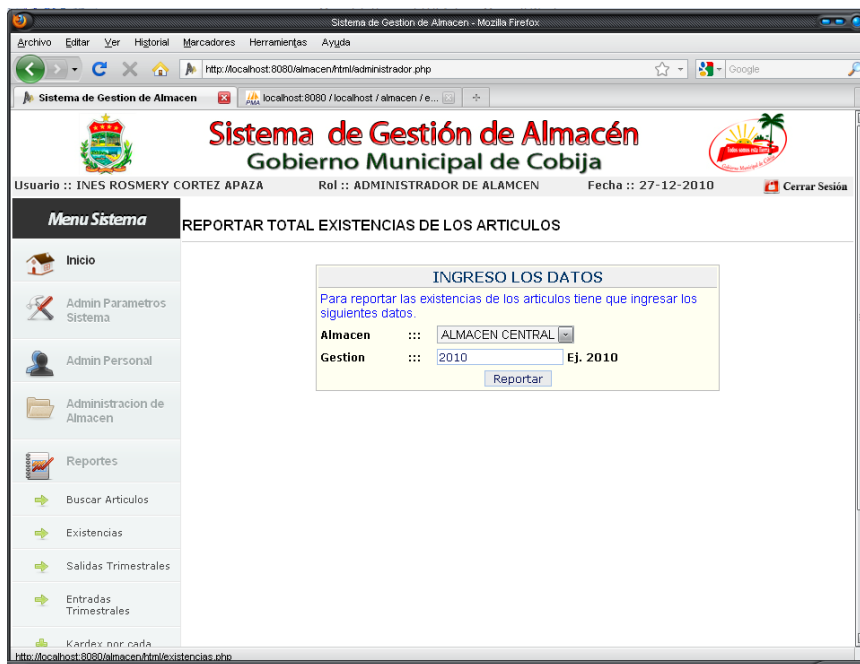
Ingresamos las iniciales del artículo y hacemos click en el botón buscar en la parte inferior muestra el artículo con la información necesario y útil para tener conocimiento, también este sub modulo se puede utilizar para averiguar cuánto se tiene en existencia de tal artículo.

## ANEXO “D” MANUAL DE USUARIO

---

Con esto terminamos la parte del buscar articulo y resaltando la importancia de este sub modulo.

### 6.2. EXISTENCIAS



Este sub modulo de reportes muestra las existencias actuales de los artículos en almacén organizados por rubros.

Así como se observa en la siguiente imagen.

## ANEXO “D” MANUAL DE USUARIO



En la parte inferior del formulario se puede observar el resumen de los artículos existentes, si se muestra en blanco quiere decir que el artículo su existencia esta en cero, también se puede observar la vista de impresión.

### 6.3. SALIDAS TRIMESTRALES

Las salidas trimestrales están específicamente diseñadas para listar las notas de salida.



Hacemos click sobre Salidas Trimestrales y el parte derecha centro veremos el formulario donde se debe ingresar la gestión y elegir el trimestre del cual se quiere realizar el reporte.

## ANEXO “D” MANUAL DE USUARIO



Después se puede ver el reporte en la parte inferior del formulario donde podemos elegir la opción vista impresión.

Con esto terminados la sección salidas trimestrales y la sección entradas trimestrales es similar solo cambia el titulo, entonces no lo explicaremos.

# ANEXO “D” MANUAL DE USUARIO

## 6.4. KARDEX POR CADA ARTÍCULO

El kardex por cada artículo ayuda a sacar kardex de un artículo, se debe especificar el almacén el artículo y la gestión y así reportara el artículo especificado.

Así como se observa en la imagen.

The screenshot shows a web browser window with the URL `http://localhost:8080/almacen/vhtml/administrador.php`. The page title is 'Sistema de Gestión de Almacén Gobierno Municipal de Cobija'. The user is logged in as 'INES ROSMERY CORTEZ APAZA' with the role 'ADMINISTRADOR DE ALAMCEN' on '27-12-2010'. The main menu on the left includes 'Inicio', 'Admin Parametros Sistema', 'Admin Personal', 'Administracion de Almacen', 'Reportes', 'Buscar Articulos', 'Existencias', 'Salidas Trimestrales', 'Entradas Trimestrales', and 'Kardex por cada Articulo'. The main content area is titled 'REPORTAR KARDEX POR CADA ARTICULO' and contains a form for reporting the kardex. The form has the following fields: 'Almacen' (dropdown menu with 'ALMACEN CENTRAL' selected), 'Articulo' (text input field), and 'Gestion' (dropdown menu with '2010' selected and 'Ej. 2010' as a label). A 'Reportar' button is located below the form. Below the form is a 'Vista de Impresion' button and a table with the following data:

NRO	FECHA	DETALLE	CONTROL FISICO			PRECIO	CONTROL VALORADO		
			ENTRADAS	SALIDAS	SALDO		ENTRADA	SALIDA	SALDO
1	01/01/2010	INVENTARIO INICIAL	---	---	0	0	---	---	0
2	2010-12-23	COMPRA DE MATERIALES	10	---	10	25	250	---	250
3	2010-12-24	ENTREGA	---	5	5	25	---	125	125
4	2010-12-24	ENTREGA	---	2	3	25	---	50	75

Este kardex individual se puede imprimir para luego adjuntar al informe.

# ANEXO “D” MANUAL DE USUARIO

## 6.5. KERDEX GENERAL DE ARTICULOS

El kardex general hace un reporte de todos los artículos en conjunto ordenadamente por rubros.

Para este reporte se debe ingresar datos como: el almacén y la gestión. Así como se observa en la imagen.



También se puede ver la vista impresión haciendo click en el botón Vista de Impresión.

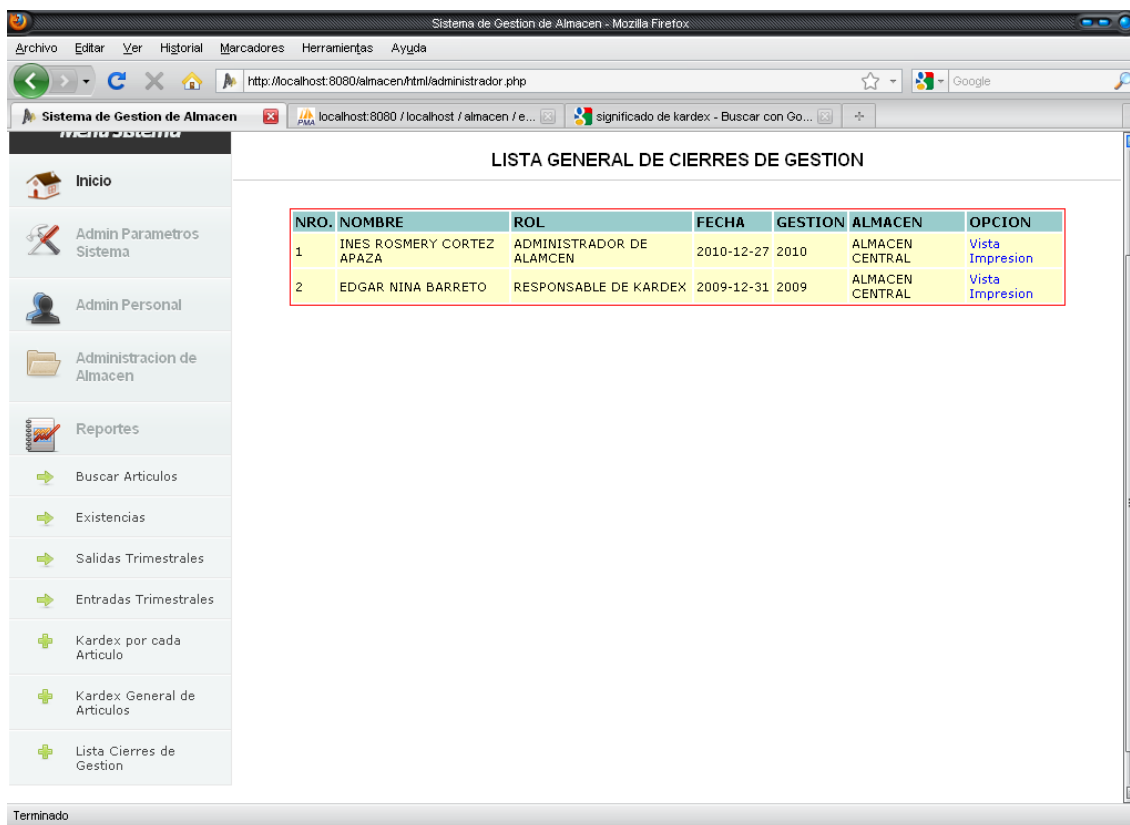


Modo vista impresión

# ANEXO “D” MANUAL DE USUARIO

## 6.6. LISTA CIERRES DE GESTION

Muestra una lista de los cierres de gestión que se realizaron.



The screenshot shows a web browser window with the title "Sistema de Gestion de Almacen - Mozilla Firefox". The address bar shows "http://localhost:8080/almacen/html/administrador.php". The page content is titled "LISTA GENERAL DE CIERRES DE GESTION". On the left, there is a sidebar menu with options like "Inicio", "Admin Parametros Sistema", "Admin Personal", "Administracion de Almacen", "Reportes", "Buscar Articulos", "Existencias", "Salidas Trimestrales", "Entradas Trimestrales", "Kardex por cada Articulo", "Kardex General de Articulos", and "Lista Cierres de Gestion". The main content area displays a table with the following data:

NRO.	NOMBRE	ROL	FECHA	GESTION	ALMACEN	OPCION
1	INES ROSMERY CORTEZ APAZA	ADMINISTRADOR DE ALAMCEN	2010-12-27	2010	ALMACEN CENTRAL	Vista Impresion
2	EDGAR NINA BARRETO	RESPONSABLE DE KARDEX	2009-12-31	2009	ALMACEN CENTRAL	Vista Impresion

También se puede observar la vista impresión de cada cierre de gestión que se realizó.

Con esto terminamos el manual de usuario del sistema de gestión de almacén, estas pautas que se dio en este manual, será de gran ayuda en algunas circunstancias en la cual necesite consultar algún proceso.

Para mas consultas a detalles, directamente a la Unidad de Sistemas del Gobierno Municipal de Cobija.

EVALUACIÓN DEL SOFTWARE POR PARTE DEL USUARIO CON ACCESO A  
SISTEMA DE GESTION DE ALMACEN

Calificaciones	Valor
Muy mal	1
Pésimo	2
Regular	3
Bueno	4
Muy Bueno	5

Nro.	Atributos	Puntaje
<b>FUNCIONALIDAD</b>		
1	¿Cumple los requerimientos básicos funcionales?	
2	¿El sistema cuenta con algún tipo de seguridad en cuanto al acceso al sistema ?	
3	¿Posee niveles de acceso al sistema de almacén?	
<b>FIABILIDAD</b>		
4	¿Las entradas al sistema son seguras?	
5	¿El sistema muestra fallas al realizar entradas y salidas?	
6	¿El sistema le permite recuperar datos en caso de fallos?	
<b>USABILIDAD</b>		
8	¿El manejo del sistema es de fácil entender?	
9	¿Las funcionalidades del sistema son fáciles de aprender?	
10	¿El funciona correctamente con la ausencia del Mouse o Ratón?	
11	¿El sistema es atractivo y tiene un buen interfaz de usuario?	
<b>EFICIENCIA</b>		
12	¿El sistema hace respuesta a las peticiones en el menor tiempo posible menor a 10 segundos?	
13	¿El uso de recursos no es elevado si no es razonable?	