

VSHOP-CMS SISTEMA DE GESTIÓN DE CONTENIDOS PARA TIENDAS  
VIRTUALES CON SOPORTE A DISPOSITIVOS MÓVILES

FRANCISCO JAVIER BRAVO VILLOTA  
CARLOS FIDEL TONGUINO ROSERO

UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
SAN JUAN DE PASTO  
2008

VSHOP-CMS SISTEMA DE GESTIÓN DE CONTENIDOS PARA TIENDAS  
VIRTUALES CON SOPORTE A DISPOSITIVOS MÓVILES

FRANCISCO JAVIER BRAVO VILLOTA  
CARLOS FIDEL TONGUINO ROSERO

Trabajo de Grado presentado como requisito parcial para optar el título de  
Ingeniero de Sistemas

Ing. Jesús Insuasty Portilla  
Director de proyecto

UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
SAN JUAN DE PASTO  
2008

"Las ideas y las conclusiones aportadas en el presente trabajo son responsabilidad exclusiva de sus autores"

Artículo 1, acuerdo No. 324 de octubre 11 de 1966, emanado por el Honorable Consejo Directivo de la Universidad de Nariño.

Nota de aceptación:

---

---

---

---

---

---

---

Firma del presidente del jurado

---

Firma del jurado

---

Firma del jurado

San Juan de Pasto, 03 de Noviembre de 2008.

## CONTENIDO

|   | pág. |
|---|------|
| INTRODUCCIÓN.....   | 17   |
| 1. CONTEXTUALIZACIÓN.....   | 22   |
| 1.1. COMERCIO ELECTRÓNICO.....  | 22   |
| 1.2. TIENDA VIRTUAL.....  | 24   |
| 1.2.1. Carro de compras.....  | 24   |
| 1.2.2. Mecanismo de ofertas y descuentos.....                           | 25   |
| 1.2.3. Cálculo automático de impuestos (IVA).....                       | 25   |
| 1.2.4. Informes de ventas.....  | 25   |
| 1.2.5. Métodos de pago.....   | 25   |
| 1.3. SEGURIDAD EN COMERCIO ELECTRÓNICO.....                             | 26   |
| 1.4. MARKETING.....   | 27   |
| 1.5. PRE VENTA.....   | 27   |
| 1.6. GESTIÓN DE CONTENIDOS.....   | 28   |
| 1.6.1. Sistema de gestión de contenidos.....                            | 28   |
| 1.6.2. Tipos de sistemas de gestión de contenidos.....                  | 28   |
| 1.6.3. Necesidad de un CMS.....   | 29   |
| 1.7. DESARROLLO DE SOFTWARE ORIENTADO A COMPONENTES.....                | 30   |
| 1.7.1. Beneficios del desarrollo de software basado en componentes..... | 31   |
| 1.7.2. Industrialización del desarrollo de software.....                | 32   |
| 1.8. XML.....   | 32   |
| 1.9. DESARROLLO DE SOFTWARE PARA DISPOSITIVOS MÓVILES.....              | 33   |
| 2. ANTECEDENTES.....  | 35   |

|        |  |    |
|--------|--|----|
| 2.1.   | AB-SHOP 3.....                                     | 35 |
| 2.2.   | JOOMLA.....  | 38 |
| 2.3.   | MAMBO.....   | 42 |
| 2.4.   | OSCOMMERCE: (OPEN SOURCE COMMERCE).....            | 45 |
| 3.     | HERRAMIENTAS UTILIZADAS.....                       | 51 |
| 3.1.   | WEB 2.0.....                                       | 51 |
| 3.2.   | PHP.....   | 52 |
| 3.3.   | JAVASCRIPT.....                                    | 55 |
| 3.4.   | AJAX.....  | 56 |
| 3.5.   | XAJAX.....   | 58 |
| 3.6.   | CSS: (CASCADING STYLE SHEETS).....                 | 59 |
| 3.7.   | JAVA ME.....                                       | 60 |
| 3.7.1. | CDC (Connected Device Configuration).....          | 61 |
| 3.7.2. | CLDC (Connected Limited Device Configuration)..... | 62 |
| 3.8.   | APACHE.....  | 63 |
| 3.9.   | MYSQL.....   | 64 |
| 3.10.  | XAMPP.....   | 64 |
| 4.     | ANÁLISIS Y DISEÑO VSHOP CMS.....                   | 65 |
| 4.1.   | LENGUAJE DE MODELADO UNIFICADO.....                | 65 |
| 4.1.1. | Diagramas de clases.....                           | 66 |
| 4.1.2. | Diagrama de paquetes.....                          | 66 |
| 4.1.3. | Diagrama de casos de uso.....                      | 67 |
| 4.2.   | EL PROCESO RACIONAL UNIFICADO O RUP.....           | 68 |
| 4.3.   | ANÁLISIS UML.....                                  | 70 |
| 4.3.1. | Funciones.....                                     | 70 |

|          |                                  |     |
|----------|----------------------------------|-----|
| 4.3.2.   | Casos de uso .....               | 72  |
| 4.3.2.1. | Aplicación web .....             | 72  |
| 4.3.2.2. | Aplicación móvil .....           | 90  |
| 4.3.3.   | Diagramas de casos de uso .....  | 94  |
| 4.3.4.   | Diagramas de clases .....        | 99  |
| 4.3.5.   | Diagramas de paquetes.....       | 122 |
| 4.3.5.1. | Aplicación web.....              | 122 |
| 4.3.5.2. | Aplicación móvil.....            | 124 |
| 4.3.6.   | Diagrama Entidad Relación .....  | 129 |
| 4.4.     | DISEÑO .....                     | 130 |
| 5.       | ARQUITECTURA DEL SISTEMA.....    | 153 |
| 5.1.     | Administrador de la tienda ..... | 153 |
| 5.1.1.   | Sistema operativo .....          | 160 |
| 5.1.2.   | Servidor web.....                | 160 |
| 5.1.3.   | Seguridad .....                  | 160 |
| 5.1.4.   | Entorno de trabajo .....         | 161 |
| 5.1.5.   | xWebComponentes .....            | 163 |
| 5.1.5.1. | XDataGridComponent.....          | 163 |
| 5.1.5.2. | xTabularDataAdmin .....          | 165 |
| 5.1.5.3. | xMasterDetail .....              | 167 |
| 5.1.5.4. | xCreateCSS.....                  | 169 |
| 5.1.6.   | CSS .....                        | 173 |
| 5.1.7.   | Capa de acceso a datos .....     | 173 |
| 5.2.     | TIENDA GENERADA.....             | 174 |
| 5.3.     | APLICACIÓN MOVIL.....            | 174 |

|   |     |
|---|-----|
| 5.3.1. Interfaz SVG..   | 175 |
| 5.3.1.1. Paquete SVG. El paquete SVG se compone de las siguientes clases: | 175 |
| 5.3.2. Persistente de datos.  | 176 |
| 5.3.2.1. Paquete conexión.  | 176 |
| 5.3.2.2. Paquete proceso  | 177 |
| 5.3.3. Sincronización.  | 177 |
| CONCLUSIONES  | 179 |
| Respecto a la aplicación web  | 179 |
| Respecto a la aplicación Móvil.....                                       | 180 |
| RECOMENDACIONES.....  | 181 |
| ANEXOS.....   | 183 |
| A1. GUIA DE INSTALACIÓN DE LA APLICACIÓN.....                             | 183 |
| Instalación.....  | 183 |
| Panel de Control de XAMPP.....  | 188 |
| A2. GUIA DE INSTALACIÓN LAMPP EN LINUX.....                               | 191 |
| REFERENCIAS BIBLIOGRÁFICAS.....   | 194 |

## ANEXOS

|  | pág. |
|--|------|
| A1. MANUAL DE INSTALACIÓN              | 169  |
| A2. GUIA DE INSTALACIÓN LAMPP EN LINUX | 177  |

## LISTA DE TABLAS

|   | pág. |
|---|------|
| Tabla 1 Versiones y cambios PHP                       | 54   |
| Tabla 2. Funciones del sistema                        | 70   |
| Tabla 3. Atributos principales de xDataGridComponents | 163  |
| Tabla 4. Atributos principales de xDataGridColumn     | 164  |

## LISTA DE FIGURAS

pág.

|   |     |
|---|-----|
| Figura 1 Participación de las exportaciones hacia estados unidos en las exportaciones de Colombia y de Nariño ..... | 1   |
| Figura 2 Sincronización entre dispositivo móvil y servidorMaximail.....   | 33  |
| Figura 3 Administrador AB-SHOP.....   | 36  |
| Figura 4 Modo diseñador AB-SHOP .....   | 37  |
| Figura 5 Administrador de Joomla .....  | 40  |
| Figura 6 Sitio Web realizado mediante Joomla.....   | 41  |
| Figura 7 Administrador de Mambo.....  | 44  |
| Figura 8 Sitio desarrollado en Mambo .....  | 45  |
| Figura 9 Módulo que ve el cliente y que permite realizar la interacción con este. .                                 | 47  |
| Figura 10 Módulo administrativo de osCommerce.....  | 48  |
| Figura 11 Comparación entre una aplicación clásica y una aplicación Ajax .....                                      | 58  |
| Figura 12 Plataforma Java.....  | 61  |
| Figura 13 CDC (Connected Device Configuration) .....  | 1   |
| Figura 14 CLDC (Connected Limited Device Configuration).....  | 1   |
| Figura 15 El Proceso Racional Unificado o RUP .....   | 69  |
| Figura 16 Diagrama de casos de uso 1 .....  | 1   |
| Figura 17 Diagrama de casos de uso 2 .....  | 95  |
| Figura 18 Diagrama de casos de uso 3 .....  | 1   |
| Figura 19 Diagrama de casos de uso 4 .....  | 1   |
| Figura 20 Diagrama de casos de uso 5 Aplicación Móvil.....  | 97  |
| Figura 21 Diagrama de Clases - Lógica Comercial.....  | 101 |
| Figura 22 Diagrama de clases detallado - Lógica Comercial 1 .....   | 102 |
| Figura 23 Diagrama de clases detallado - Lógica Comercial 2.....  | 103 |
| Figura 24 Diagrama de clases detallado - Lógica Comercial 3.....  | 104 |

|  |     |
|--|-----|
| Figura 25 Diagramas de Clases detallado - Lógica comercial 4 .....         | 105 |
| Figura 26 Diagrama de clases - Manejo de apariencia de la tienda .....     | 107 |
| Figura 27 Diagrama de clases - Gestión de contenidos .....                 | 108 |
| Figura 28 Diagrama de clases detallado - Gestión de contenidos 1 .....     | 110 |
| Figura 29 Diagrama de clases detallado - Gestión de contenidos 2 .....     | 112 |
| Figura 30 Diagrama de clases detallado – XwebComponents .....              | 113 |
| Figura 31 Diagrama de clases detallado - Aplicación Móvil 1 .....          | 114 |
| Figura 32 Diagrama de clases detallado - Aplicación Móvil 2 .....          | 115 |
| Figura 33 Diagrama de clases detallado - Aplicación Móvil 3 .....          | 116 |
| Figura 34 Diagrama de clases detallado - Aplicación Móvil 4 .....          | 1   |
| Figura 35 Diagrama de clases detallado - Aplicación Móvil 5 .....          | 1   |
| Figura 36 Diagrama de clases detallado - Aplicación Móvil 6 .....          | 119 |
| Figura 37 Diagrama de clases detallado - Aplicación Móvil 7 .....          | 1   |
| Figura 38 Diagrama de paquetes - Gestión Comercial .....                   | 122 |
| Figura 39 Diagrama de Paquetes - FaceStore .....                           | 123 |
| Figura 40 Diagrama de paquetes – Content .....                             | 123 |
| Figura 41 Diagrama de paquetes – XwebComponents .....                      | 124 |
| Figura 42 Diagrama de paquetes - Aplicación Móvil - Proceso .....          | 125 |
| Figura 43 Diagrama de paquetes - Aplicación Móvil - Paquete SVG .....      | 126 |
| Figura 44 Diagrama de paquetes - Aplicación Móvil - Paquete Conexión ..... | 1   |
| Figura 45 Diagrama entidad relación .....                                  | 1   |
| Figura 46 Diseño de pantalla - Administración de tiendas .....             | 130 |
| Figura 47. Diseño de pantalla - Diseño de la tienda .....                  | 131 |
| Figura 48 Diseño de pantalla - Diseño de catálogos.....                    | 131 |
| Figura 49 Diseño de pantalla - Diseño sección Detalle – producto.....      | 132 |
| Figura 50 Diseño de pantalla - Datos de conexión a servidor.....           | 132 |
| Figura 51 Diseño de pantalla - Crear plantilla de tienda .....             | 133 |
| Figura 52 Diseño de pantalla - Crear plantilla de catálogos.....           | 134 |
| Figura 53 Diseño de pantalla - Crear plantilla de detalle producto.....    | 135 |
| Figura 54 Diseño de pantalla - Editar Plantillas .....                     | 136 |
| Figura 55 Diseño pantalla - Importar plantilla .....                       | 137 |

|   |     |
|---|-----|
| Figura 56 Diseño de pantalla - Gestión de productos .....                       | 139 |
| Figura 57 Diseño de pantalla - Registrar compra.....                            | 140 |
| Figura 58 Diseño de pantalla - Registrar Venta .....                            | 142 |
| Figura 59 Diseño de pantalla - Registrar venta oferta.....                      | 143 |
| Figura 60 Diseño de pantalla - Costo de productos .....                         | 144 |
| Figura 61 Diseño de pantalla - Administración de noticias .....                 | 145 |
| Figura 62 Diseño de pantalla - Administración de ofertas.....                   | 146 |
| Figura 63 Diseño de pantalla - Administración de opciones adicionales .....     | 147 |
| Figura 64 Diseño de pantalla - Interfaz básica xTabularDataAdmin .....          | 148 |
| Figura 65 Diseño de pantalla - Interfaz básica de componente xMasterDetail .... | 149 |
| Figura 66 Diseño de pantalla – Menú Móviles .....                               | 149 |
| Figura 67 Diseño de pantalla – Ingreso Clientes .....                           | 150 |
| Figura 68 Diseño de pantalla – Pre venta - Seleccionar Cliente .....            | 151 |
| Figura 69 Diseño de pantalla – Pre venta - Ingreso Producto Pre venta .....     | 151 |
| Figura 70 Diseño de pantalla – Pre venta.....                                   | 152 |
| Figura 71 Diseño de pantalla – Pantalla de Carga.....                           | 152 |
| Figura 72 Arquitectura del sistema .....  | 155 |
| Figura 73 Arquitectura del sistema - Diagrama en bloques .....                  | 156 |
| Figura 74 Esquema de directorios VSHOP CMS.....                                 | 159 |
| Figura 75 Entorno de trabajo VSHOP CMS.....                                     | 161 |
| Figura 76 Cuadro modal para formularios .....                                   | 162 |
| Figura 77 Modal para alertas .....  | 162 |
| Figura 78 Modal de confirmación.....  | 162 |
| Figura 79 Ejemplo funcionamiento de xDataGridComponent .....                    | 165 |
| Figura 80 Ejemplo funcionamiento xTabularDataAdmin .....                        | 166 |
| Figura 81 Ejemplo Tablas relacionales xMasterDetail .....                       | 167 |
| Figura 82 Ejemplo funcionamiento xMasterDetail.....                             | 169 |
| Figura 83 Ejemplo funcionamiento de xCreateCSS.....                             | 170 |
| Figura 84 Pantalla inicial de instalación XAMPP.....                            | 183 |
| Figura 85 Inicio de instalación XAMPP .....                                     | 184 |
| Figura 86 Instalación XAMPP - Selección directorio destino .....                | 185 |

|  |     |
|--|-----|
| Figura 87 Instalación XAMPP - Servicios .....                            | 186 |
| Figura 88 Instalación XAMPP - Finalización .....                         | 187 |
| Figura 89 Instalación XAMPP - Confirmación inicio panel de control ..... | 188 |
| Figura 90 Panel de control XAMPP .....                                   | 189 |
| Figura 91 PHPMYADMIN - Importar archivo.....                             | 190 |
| Figura 92 Instalación Lampp en Linux .....                               | 193 |

## **RESUMEN**

En este proyecto se muestra las etapas de análisis, diseño e implementación de VSHOP CMS, herramienta que permite la creación y administración de tiendas virtuales con soporte a dispositivos móviles.

La herramienta se compone de módulos que permiten la creación y gestión de una tienda virtual, permitiendo administrar la parte comercial de la tienda, la apariencia del sitio generado y los contenidos que aparecen en el mismo, permitiendo así a los usuarios implementar estrategias de comercio electrónico en sus empresas.

La herramienta cuenta con soporte a dispositivos móviles, mediante el sistema el usuario podrá contar con la información comercial de la tienda en su dispositivo y realizar pre ventas cuyos datos se sincronizan luego a la aplicación web principal.

## **ABSTRACT**

This project shows the stages of analysis, design and implementation of VSHOP CMS, a tool that allows the creation and administration of virtual stores with support for mobile devices.

The tool consists of modules that allow the creation and management of a virtual store, allowing manage the commercial part of the store, the appearance of the generated site and content that appear in it, allowing users implement e-commerce strategies in their companies.

The tool has support for mobile devices; using the system the users can get the business information from the store on their devices and perform pre sellings which are then synchronized with the main web application.

## INTRODUCCIÓN

El “conocimiento”, en su más amplia definición y en sus más diversas formas, es un bien cuya universalización permite el desarrollo y el avance de la sociedad en todos sus ámbitos, tanto tecnológicos sociales, considerando lo anterior y teniendo en cuenta que el Comercio Electrónico es la aplicación de tecnologías de información como los sistemas distribuidos y la movilidad para incrementar la eficiencia y efectividad de las relaciones de negocios entre socios comerciales y clientes, planteamos el desarrollo de un sistema de administración de tiendas virtuales para entorno Web que de soporte a gestión de tareas mediante dispositivos móviles el cual permitirá la aplicación y aprovechamiento de Internet como un medio masivo de comercialización para empresarios de la región.

El proyecto desarrollado tiene como título “VSHOP-CMS Sistema de Gestión de contenidos para tiendas virtuales con soporte a dispositivos móviles”, que está ubicado dentro de la línea de investigación de software y manejo de información.

El sistema “VSHOP-CMS” es una herramienta que permite la generación y administración de tiendas virtuales para entornos Web con las siguientes características, se divide por funcionalidad en tres módulos de la siguiente forma:

**Módulo1:** Generador de tiendas virtuales. Que permite configurar y generar las tiendas virtuales con las siguientes características:

- Gestión de datos generales de la tienda.
- Selección de estilo y apariencia de la tienda.

**Módulo 2:** Administrador de tienda virtual. Para administrar las tiendas virtuales, con las siguientes características:

- Administración de usuarios.
- Gestión de productos, marcas, precios.
- Gestión de catálogos de productos.
- Gestión de clientes.
- Gestión de pedido.
- Gestión de devoluciones.

**Módulo 3:** Comercialización de la tienda virtual. Para realizar el proceso de comercialización con el usuario final (cliente de la tienda), con funcionalidades tales como:

- Catálogo de productos.
- Buscador de productos.
- Carro de mercados.
- Gestión de pedidos.

También se da soporte al proceso de comercialización para dispositivos móviles con las siguientes características:

- Catálogo de productos.
- Buscador de productos directo.
- Gestión de pedidos.
- Gestión de clientes.

Se aclara que existen diversos mecanismos para la adquisición de productos por Internet, el sistema se limitará a generar la factura de la venta correspondiente para que el cliente realice una consignación con el código de su factura, dejando abierta la posibilidad de ampliar funcionalidades de pago con tarjeta de crédito u otros mecanismos.

La necesidad de evolución de las empresas que exige la alta competencia en el mercado actual, obliga a las empresas a estar en la continua búsqueda de estrategias que les permitan mantener y aumentar el nivel de ganancias y el mercado objetivo a quien se dirigen sus productos o servicios.

**Figura 1 Participación de las exportaciones hacia estados unidos en las exportaciones de Colombia y de Nariño**

**Participación de las exportaciones hacia Estados Unidos en las exportaciones totales de Colombia y de Nariño  
(millones de dólares)**

|                                    | 1995   | 1999   | 2004   |
|------------------------------------|--------|--------|--------|
| <b>A Colombia</b>                  |        |        |        |
| Exportaciones totales              | 10,201 | 11,617 | 16,730 |
| Hacia EEUU                         | 3,527  | 5,616  | 6,598  |
| <b>B. Departamento de Nariño</b>   |        |        |        |
| Exportaciones totales*             | 56.2   | 29.5   | 40.6   |
| Hacia EEUU*                        | 43.3   | 10.5   | 3.6    |
| <b>C. Indicadores</b>              |        |        |        |
| Exp.a EU / Exp. Colombia (%)       | 34.57  | 48.34  | 39.44  |
| Exp. Nariño / Exp. Colombia (%)    | 0.55   | 0.25   | 0.24   |
| Exp. Nariño a EU / Exp. Nariño (%) | 77.10  | 35.61  | 8.98   |

**DANE**

Según las estadísticas Nariño siempre ha sido catalogado a nivel nacional como uno de los departamentos que menos exportaciones genera, teniendo en cuenta esta situación se nota en el entorno regional un claro desaprovechamiento de la tecnología para el comercio de artículos hacia el mercado nacional e internacional, esto debido a que los empresarios no cuentan con herramientas que ofrezcan dichos servicios y también al desconocimiento del potencial que han adquirido herramientas como el comercio electrónico a nivel mundial.

Actualmente para las empresas a nivel regional es muy difícil generar sitios Web que permitan el comercio electrónico, esto debido al hecho de no estar familiarizados con dicha tecnología y no contar con herramientas que permitan disminuir el tiempo de desarrollo y por consiguiente el costo del software.

La falta de contacto rápido y eficiente con los clientes, los costos adicionales que tienen que asumir por reuniones y envío de artículos, el desconocimiento de los alcances de la tecnología son hechos que hacen que el crecimiento económico de la región no se dé en gran escala, afectando así los ingresos y oportunidad de expansión de las empresas.

Para el desarrollo del sistema se plantea como objetivo general crear una herramienta que permita la generación y administración de tiendas virtuales que de soporte a dispositivos móviles, se divide este objetivo en tareas funcionales que

permiten que la aplicación brinde un correcto soporte a las necesidades que presenta tanto el usuario administrador para la generación de la tienda y como usuarios finales al consultar y seleccionar productos para agregar a su canasta de mercado, las tareas funciones en que se divide el módulo son:

- Desarrollar un módulo generador de tiendas virtuales que permita configurar los datos generales, el estilo y apariencia y la selección de módulos y opciones a incluir en la tienda virtual.
- Desarrollar un módulo de administración de tiendas virtuales de fácil uso para gestionar la información relacionada con clientes, productos, ventas y devoluciones, con la seguridad necesaria.
- Desarrollar un módulo de comercialización para usuarios finales que contiene todas las funcionalidades indispensables para el cliente además de contar con soporte para dispositivos móviles.

La búsqueda de nuevos mercados donde los productos puedan competir y ser exitosos es la tarea más tediosa y costosa para las empresas, el comercio tradicional busca a los clientes en un reducido campo, las nuevas tecnologías permiten la aparición del comercio electrónico, donde la empresa por medios electrónicos busca difundir sus productos en mercados que no puede explorar directamente o presencialmente, ya que los costos son muy altos. Internet es la principal herramienta que soporta el comercio electrónico, y la principal representación de la globalización de los mercados. Las tiendas virtuales son páginas Web que representan “locales virtuales” que permite a los clientes comprar y satisfacer su necesidad de productos con gran calidad y a bajos precios, logrando que las pymes puedan competir con las grandes multinacionales sin tener locales de ventas reales que representan un costo adicional para la venta de sus productos.

“VSHOP-CMS” es una herramienta que le permite a los usuarios no expertos y sin conocimientos previos de algún lenguaje de programación generar y administrar tiendas virtuales, de esta forma se abren nuevos mercados a sus empresas en relaciones como compañía-compañía y compañía-cliente, los productos podrán ser comprados sin necesidad de que existan intermediarios entre la empresa y el mercado, agilizando la entrega y la ganancia de un reconocimiento sin una inversión económica grande.

“VSHOP-CMS” se fundamenta en la tendencia de los CMS (Content Management System) que son herramientas software que le permiten al usuario crear y gestionar de forma uniforme, accesible y cómoda un sitio Web dinámico, con

actualizaciones periódicas y sobre el que puedan trabajar una o varias personas, cada una con una función determinada. Desde el punto de vista del cliente se trata de un sitio Web dinámico, con apariencia e interfaz uniforme, con diseño centrado en el usuario y que permite llevar a cabo fácilmente las tareas para las que ha sido diseñado.

Las nuevas tecnologías de conexión inalámbrica a Internet como son Wi-Fi (Wireless Fidelity) y WiMax ('Worldwide Interoperability for Microwave Access'), aumentaran la demanda de dispositivos móviles (Palms, Pocket PC, Smartphone), que se puedan conectar a Internet y que permitan realizar tareas de comercio electrónico.

Una ventaja fundamental de "VSHOP-CMS" es que ofrece los servicios de comercialización de productos a través de dispositivos móviles. Así la empresa podrá aumentar su productividad, reducir costos en comercializaron, promoción y mercadeo, y darle al cliente servicios adicionales de adquisición de productos en cualquier lugar del mundo, globalizando así su mercado y llegando a clientes que son aficionados a los dispositivos móviles y la navegación en Internet.

## **1. CONTEXTUALIZACIÓN**

En este punto se hace referencia a conceptos relativos al proyecto importantes para comprender el funcionamiento y características de VSHOP CMS.

### **1.1. COMERCIO ELECTRÓNICO**

Dentro de las tendencias comerciales actuales se presenta el comercio electrónico como una alternativa mediante la cual el empresario puede cubrir un extenso mercado sin acarrear un costo elevado, actualmente la información que se tiene acerca de este servicio o medio de comercialización es muy escaso dado que regionalmente muy pocas personas u empresas lo utilizan, se tiene la idea de comercio electrónico una persona frente a un computador personal, visitando sitios y realizando compras, pero aunque el E-commerce (Comercio Electrónico) es la compra y venta de bienes y servicios a través de internet y más específicamente en el World Wide Web esta imagen es muy vaga para tener en cuenta todo lo que lo compone, el comercio electrónico abarca muchas técnicas, ya sean de marketing, de software, de hardware y muchos otros componentes relacionados, los cuales permiten hacer de esta nueva tecnología, una de las más innovadoras y con un futuro prometedor [1].

E-commerce está estructurado por "Tiendas virtuales" en sitios web que ofrecen catálogos en línea. Incluso se han creado "Centros comerciales virtuales" con gran cantidad de tiendas con todo tipo de accesorios para la venta tales como Centrovirtual (<http://www.centrovirtual.com/>), TiendaNet, Escaparates, entre otros. Esta forma de comercio electrónico ha consolidado a grandes empresas que ya figuran en la bolsa de valores y son de los portales de internet más visitados. Por ejemplo, Amazon.com se ha consolidado como la librería más importante del mundo, recibiendo varios millones de visitantes al mes y miles de pedidos de libros; ha abierto inclusive nuevos servicios de venta de música y videos debido al éxito que ha obtenido y que se debe en parte a la interacción con los usuarios que pueden recomendar libros y escribir sus opiniones al respecto en la misma sección donde se vende. Otro destacado en este campo es CD-NOW, tienda donde se puede adquirir todo tipo de música y a la vez encontrar información sobre todos los artistas que allí figuran, muchos de las grandes empresas empiezan a

vincularse a internet como una forma de comercio alternativa, tales como Almacenes Éxito (<http://www.virtualexito.com/>) y la 14.

Y así, cientos de negocios han nacido en internet y otras compañías que ya se habían consolidado en el mercado, están haciendo la transición, ofreciendo cada vez mejores servicios por la competencia generada.

El comercio electrónico brinda a los empresarios oportunidades como:

- Presencia global. Internet es un medio de difusión masiva que garantiza que en cualquier parte del mundo puedan acceder a los servicios de una tienda virtual, además permite que el cliente tenga múltiples alternativas para seleccionar el producto que cumpla sus expectativas.
- Aumento de competitividad del servicio: el comercio electrónico permite a los proveedores llegar a muchos más clientes sin necesidad de efectuar grandes inversiones para ello, por ejemplo muchos proveedores usan el comercio electrónico para ofrecer un mejor servicio tanto en la pre y pos venta de sus productos y servicios, la mejora en la prestación de estos servicios permite que logre una gran mejora en la calidad del servicio.
- Adecuación personalizada de productos y servicios: el comercio electrónico permite un cercano contacto con el cliente pudiendo personalizar los productos y servicios adecuándolos a las necesidades de los compradores.
- Proceso de entrega: el comercio electrónico muchas veces acorta la cadena de entrega, haciendo no necesaria la intervención de intermediarios para que los productos lleguen al cliente final.
- Reducción de costos y precios: el hecho de que el local de una tienda virtual sea un sitio web alojado en un servidor hace que se reduzca en gran medida los costos de mantenimiento de la empresa, además el comercio electrónico permite estar en contacto directo con los clientes haciendo que los costos de transporte y logística se reduzcan.

Se debe anotar que es de especial importancia la asesoría de un ingeniero de sistemas, licenciado en informática, o profesional en áreas afines, el cual genere conceptos acerca de los procesos ejecutados en la actividad comercial, él será quien asegure que los procesos cumplen con los parámetros de seguridad y funcionalidad necesarios.

## 1.2. TIENDA VIRTUAL

Se entiende como tienda virtual a una analogía de una tienda física que comercializa productos o servicios, montada en un sitio web. El equivalente del local físico es el conjunto de páginas web mostradas al visitante desde el servidor web. Lógicamente, el local de la tienda debe estar en las mejores condiciones, se debe elegir muy bien los elementos que compongan la tienda y eso supone elegir bien dónde vamos a ubicar el servidor web y la apariencia que ha de tener.

Parte importante de la tienda al igual que en una tienda física es el mostrador o vitrina en donde se van a alojar los productos y catálogos que va a mirar el cliente, estos deben ser acordes con la imagen que se pretenda dar al negocio. El uso de los colores y los elementos gráficos pueden convertir la experiencia de compra del cliente en algo agradable que desee realmente repetir y recomendar a otras personas. Sin embargo, debe prestarse total atención al balance entre una vistosa presentación y la facilidad y velocidad de manejo y navegación de la Tienda Virtual, dado que el comercio virtual es en su mayoría una experiencia nueva para los clientes es de mucha importancia que se cubra o sobrepase las expectativas del mismo, con el fin de que repita su experiencia y difunda su agrado acerca del servicio.

En el momento en que un cliente ingresa al sitio su primera actividad será buscar los productos que le interesen, de igual manera ese es el interés del dueño de la tienda, de tal forma que es de gran importancia permitir al cliente conocer y casi palpar los productos que se ofrece. Existen varias formas de presentar esa información al cliente, en el contexto de administración del sitio se puede hablar de información estática, la cual se usa cuando la tienda ofrece muy pocos productos y la variación de sus características no es muy frecuente, ya que en tiempo de desarrollo es costoso cambiar la información; contrario a esto si la información se administra de manera dinámica, mediante una aplicación de apoyo se podrá fácilmente modificar información acerca de los productos que ofrece la tienda, esta forma de administración es más usada ya que permite un mejor control de la información del sitio y el manejo de una gran cantidad de productos.

**1.2.1. Carro de compras.** El carrito de compra es una herramienta auxiliar para el cliente, el cual permite que lleve control sobre los productos que va a adquirir y el valor de su compra, mientras el navega por la tienda en busca de otros productos, su importancia radica en que el cliente podrá adicionar, modificar y eliminar productos que le interesen; el buen o mal manejo que se le dé al carrito

de compras facilita o impide la consecución de una venta, mientras mayor sea la flexibilidad con la que el cliente interactúe con los productos y el carrito de compra, mayor será la satisfacción del mismo.

**1.2.2. Mecanismo de ofertas y descuentos.** Los comercios electrónicos y por lo tanto el software de apoyo a las tiendas virtuales deben permitir realizar ofertas puntuales acerca de los productos que se ofrecen, Un descuento debe poder ser aplicado como tanto por ciento del precio del producto o como una cantidad fija sobre éste y se debe poder establecer el mínimo y máximo de cantidad de productos para aplicar a la oferta.

**1.2.3. Cálculo automático de impuestos (IVA).** El sistema debe establecer automáticamente y mostrar al cliente el valor de pago de impuestos que se va a deducir por la compra a realizar, si se trata de una venta internacional (exportación) el sistema debe omitir este cálculo, en caso de que la legislación impositiva lo permita (generalmente, no se "exportan" impuestos).

**1.2.4. Informes de ventas.** La tienda virtual debe ofrecer a la parte administrativa informes que muestren el rendimiento de las diferentes transacciones que se presentan en un periodo determinado, se destacan informes tales como:

- Análisis de las ventas entre un rango de fechas específico.
- Listado detallado de los nuevos visitantes que han accedido a la tienda virtual
- Ranking de los productos más vendidos
- Ranking de los productos más solicitados

**1.2.5. Métodos de pago.** Parte fundamental en el proceso de comercio electrónico es el pago de la mercancía que el cliente adquiere, existen métodos que permiten realizar dichos pagos de manera electrónica o manual, entre ellos están:

- Contra reembolso: el cliente paga los productos adquiridos en el momento en que son entregados a él, el proveedor debe verificar la dirección y teléfono del cliente con el fin de garantizar la entrega, este sistema es bastante conveniente para negocios que comercializan productos

localmente ya que se facilita el proceso de verificación de información del cliente y la entrega de los productos.

- Giro postal: el cliente gira el dinero de pago por medio de servicios de correo, este sistema se usa en sitios que están empezando.
- Depósito en cuenta corriente: el cliente deposita la suma de pago en una cuenta corriente indicada por el proveedor y envía copia de su consignación por fax u otro método electrónico, este sistema es muy usado en Colombia por sitios dedicados al comercio de productos como Mercado libre ([www.mercadolibre.com.co](http://www.mercadolibre.com.co)) y deremate ([www.deremate.com.co](http://www.deremate.com.co)).
- Pay Pal: El cliente envía dinero deduciéndolo de su tarjeta de crédito, la transacción se hace desde la cuenta del cliente a una cuenta del proveedor afiliado a paypal, la empresa cobra una tarifa por transacción realizada. Pay pal ganó en los SIIA Codie Awards 2002 el premio a la "Mejor solución de comercio electrónico" y está reconocido por PC Magazine como uno de "Los 100 mejores sitios Web", Pay pal se vinculo a Colombia a su red de países recientemente.
- Tarjeta de crédito: es el sistema más utilizado por tiendas virtuales, por el excesivo costo de montaje del sistema existen empresas que proporcionan plataformas seguras de pago (Gateway de pago) las cuales se enlazan con el sitio del cliente. En Colombia existen empresas tales como recaudo.net ([www.recaudo.net](http://www.recaudo.net)) o pagos online ([www.pagosonline.net](http://www.pagosonline.net)) que ofrecen estos servicios. El inconveniente de estas es que a diferencia de Pay pal cobran un porcentaje de la venta en lugar de tarifas fijas por transacción.

### **1.3. SEGURIDAD EN COMERCIO ELECTRÓNICO.**

[2] En el comercio electrónico es imprescindible garantizar la integridad y confidencialidad de la información, la incorporación de mecanismos, técnicas y algoritmos adecuados para realizar transacciones electrónicas se hace necesario para evitar los riesgos a los que se expone una tienda virtual. Se hace mención de aspectos a tener en cuenta tales como:

- Autenticación: garantizar la identidad de todos los que participan en las transacciones, se incluye personas, entidades, sistemas, etc. Para ello se usa mecanismos como la verificación de claves, verificación de direcciones, la criptografía, biometría, entre otros.

- Confidencialidad: se debe garantizar que la información usada en las transacciones solo sea de acceso para las personas autorizadas, para evitar el acceso no autorizado se usan técnicas de encriptación o codificación de datos.
- Integridad: la integridad de la información hace referencia a garantizar que la información de intercambio entre las entidades que participan en los procesos no sea manipulada por terceros, para ellos se usa mecanismos como la firma digital.

#### **1.4. MARKETING**

El éxito de una tienda virtual depende de diversos factores tales como:

- Atraer al cliente: la principal forma de atraer a clientes es la publicidad, en internet se puede recurrir a banners publicitarios, posicionamiento en buscadores, relaciones públicas, notas de prensa y colaboraciones cruzadas.
- Retener al cliente: cuando el usuario a ingresado al sitio de la tienda ahora se debe lograr que se fije en los productos y al salir adquiera uno de los productos, para ello el sitio debe garantizar rapidez en la carga, diseño innovador, fácil manejo y navegación, completa información sobre productos, servicios y forma de adquisición.
- Vender: la consecución de una venta es lo más complicado no solo en tiendas virtuales sino en el comercio en general, se debe brindar la confianza suficiente que asegure al cliente la seguridad de la transacción, es importante la claridad sobre los datos del vendedor, de los productos, de la logística de entrega y la garantía para que un cliente adquiera los productos de la tienda.
- Fidelizar al cliente: se logra primero cumpliendo las expectativas del cliente, tanto sobre las características de los productos, los plazos de entrega, etc. Además se debe mantener informado al cliente sobre las novedades de los productos, es muy importante el trato personalizado al cliente.

#### **1.5. PRE VENTA**

Pre venta es un sistema mediante el cual una persona (pre-ventistas) toma un pedido que luego es entregado y cobrado por el repartidor. Regionalmente este sistema lo aplican empresas tan conocidas como Postobon, Coca cola y Alpina.

Este sistema de comercialización permite al proveedor ofrecer sus productos sin necesidad de llevarlos consigo, actualmente se hace uso de dispositivos móviles mediante los cuales los pre-ventistas toman el pedido de mercancía, que luego se despachara por el proveedor.

## **1.6. GESTIÓN DE CONTENIDOS**

Esta sección contiene la definición de Gestor de contenidos, los tipos de sistemas de gestión que existen y la necesidad de utilizar uno.

**1.6.1. Sistema de gestión de contenidos.** [3] Un Sistema de gestión de contenidos (Content Management System, en inglés, abreviado CMS) permite la creación y administración de contenidos por parte de los participantes principalmente en páginas web.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo el público.

### **1.6.2. Tipos de sistemas de gestión de contenidos.**

- Genéricos: Ofrecen la plataforma necesaria para desarrollar e implementar aplicaciones que den solución a necesidades específicas. Pueden servir para construir soluciones de gestión de contenidos, para soluciones de comercio electrónico, blogs, portales, etc. Tales como: Zope, MODx, OpenCMS, TYPO3, Apache lenya, Joomla, Drupal.
- Foros: sitio que permite la discusión en línea donde los usuarios pueden reunirse y discutir temas en los que están interesados. Ejemplos: phpBB, SMF, MyBB.
- Blogs: Publicación de noticias o artículos en orden cronológico con espacio para comentarios y discusión. Ejemplos:
- WordPress

- Movable Type
- Custom CMS
- Blogsmith/Weblogs
- Gawker CMS
- TypePad
- Blogger/Blogspot
- Plone
- Scoop
- Wikis: Sitio web dónde todos los usuarios pueden colaborar en los artículos, aportando información o reescribiendola. También permite espacio para discusiones. Indicado para material que irá evolucionando con el tiempo. Ejemplos: MediaWiki, TikiWiki.
- eCommerce: Son Sitios web para comercio electrónico. Ejemplo: osCommerce (licencia GPL) , Dynamicweb eCommerce (comercial).
- Portal: Sitio web con contenido y funcionalidad diversa que sirve como fuente de información o como soporte a una comunidad. Ejemplos: PHP-Nuke, Postnuke, Joomla, Drupal, e-107, Plone.
- Galería: Permite administrar y generar automáticamente un portal o sitio web que muestra contenido audiovisual, normalmente imágenes. Ejemplo: Gallery.
- e-Learning: Sirve para la enseñanza de conocimientos. Los usuarios son los profesores y estudiantes, tenemos aulas virtuales donde se ponen a disposición el material del curso,... La publicación de un contenido por un profesor es la puesta a disposición de los estudiantes, en una aula virtual, de ese contenido. Ejemplo: Moodle.
- Publicaciones digitales: son plataformas especialmente diseñadas teniendo en cuenta las necesidades de las publicaciones digitales, tales como periódicos, revistas, etc. Ejemplo: ePrints.

**1.6.3. Necesidad de un CMS.** [4] El gran flujo de información que existe en una tienda y los continuos cambios que se generan en todos los componentes de la misma, hacen necesario una aplicación que permita gestionar la información en el menor tiempo y con gran facilidad.

- Inclusión de nuevas funcionalidades en el web. El añadir funcionalidades a una web implica un costo operacional relativamente alto, lo más frecuente es que se vea la necesidad de incluir nuevos componentes a destiempo, lo cual implica que se deba generar soluciones de forma

inmediata con un CMS eso puede ser tan simple como incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la web. El sistema puede crecer y adaptarse a las necesidades futuras.

- Mantenimiento de gran cantidad de páginas. Es frecuente que en una tienda virtual se dé un crecimiento desmesurado de la información, en ese momento es necesario contar con un sistema de apoyo que permita dar gestión a dicha información de manera rápida y eficiente.
- Reutilización de objetos o componentes. Un sistema gestor de contenidos permite que se usen componentes para multipropósito o que varios componentes se unan para generar nuevas funcionalidades.
- Páginas interactivas. Las páginas dinámicas permiten que el usuario pueda interactuar con los componentes del sitio, el sitio deja de ser informativo para convertirse en un portal en donde el usuario puede explorar los distintos productos de una manera amigable y personalizada.
- Cambios del aspecto de la web. Un factor muy importante a la hora de asegurar el éxito de una tienda virtual es la novedad en el diseño, muchas veces se debe cambiar el diseño de una página para renovarla o atraer más clientes, el uso de CMS y CSS (Cascading Style Sheets) permite que se modifique fácilmente el aspecto de un sitio web.
- Consistencia de la web. La consistencia en un web no quiere decir que todas las páginas sean iguales, sino que hay un orden (visual) en vez de caos. El uso de componentes web hace que el usuario se familiarice con el funcionamiento de los elementos del sitio y sea muy fácil para el navegar en el sitio en búsqueda de los productos que necesite.
- Control de acceso. Dado que la seguridad es parte fundamental a la hora de implementar un sitio web, el uso de CMS es de gran importancia dado que garantiza que se tenga control sobre las personas que acceden al sitio y las actividades que realizan.

## **1.7. DESARROLLO DE SOFTWARE ORIENTADO A COMPONENTES.**

[5] El continuo avance tecnológico ha hecho que cada vez más la gente intente ponerse a la par y dar buen uso a la información, los métodos anteriores de desarrollo de software no dan abasto para cubrir las necesidades que presentan los usuarios dada la complejidad de los requerimientos y el corto tiempo para la entrega, por tal razón aparecen nuevos paradigmas para el desarrollo de software tales como el *Desarrollo orientado a componentes* persigue el mejoramiento y optimización de recursos en el proceso desarrollo de aplicaciones.

Para enfrentar los retos que presenta el entorno actual se debe apuntar hacia la industrialización del desarrollo de software, se busca optimizar los tiempos de desarrollo y el costo del mismo, teniendo en cuenta lo anterior el desarrollo de software basado en componentes permite reutilizar piezas de código pre elaborado que permiten realizar diversas tareas, conllevando a diversos beneficios como las mejoras a la calidad, la reducción del ciclo de desarrollo y el mayor retorno sobre la inversión.

Un componente se define como una unidad de composición de aplicaciones software, que posee un conjunto de interfaces y un conjunto de requisitos, y que ha de poder ser desarrollado, adquirido, incorporado al sistema y compuesto con otros componentes de forma independiente, en tiempo y espacio.

La unión e interacción de los componentes permite crear aplicaciones, el desarrollo de proyectos de software a nivel regional se orienta hacia el desarrollo de aplicaciones a la medida, por tal razón es muy complejo el desarrollo de nuevas herramientas para cada una de las aplicaciones, por tal razón es de gran relevancia el desarrollo de componentes que puedan ser reutilizados en varias aplicaciones.

**1.7.1. Beneficios del desarrollo de software basado en componentes.** El uso del paradigma de desarrollo de software basado en componentes tiene las ventajas tales como:

- Reutilización del software. Lleva a alcanzar un mayor nivel de reutilización de software, un componente puede ser usado en varias aplicaciones.
- Simplifica las pruebas. Al realizar componentes para posterior reutilización se planean y se realizan con mucho más empeño, y al usarse más de una vez se garantiza que se hayan probado en profundidad y que funcionen de manera óptima.
- Simplifica el mantenimiento del sistema. La interacción entre componentes permite mayor facilidad en procesos de detección de errores, y para afrontar problemas que impliquen modificaciones es mucho más fácil y rápido modificar un componente que generar cambios que impliquen la modificación de la aplicación completa.
- Mayor calidad. Dado que un componente puede ser construido y luego mejorado continuamente por un experto u organización, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo.

**1.7.2. Industrialización del desarrollo de software.** Se trata de pasar de la fabricación a mano de software hacia la fabricación industrial del mismo, de buscar la automatización en los procesos de producción de software, se avanza a pasos agigantados en este enfoque gracias al desarrollo de componentes que permiten acelerar el desarrollo de interfaz gráfica de las aplicaciones pero la meta es conseguir aplicar componentes no solo para automatizar el desarrollo de interfaz gráfica sino lograr aplicarlos al máximo para ayudar en el desarrollo de la lógica del negocio como tal.

En la actualidad se usa el modelado UML como soporte del análisis en el desarrollo de aplicaciones, para mejorar la productividad es necesario buscar mecanismos que permitan configurar, adaptar y ensamblar rápidamente componentes desarrollados independientemente, autodescriptivos e independientes de ubicación.

El desarrollo de software basado de componentes se convirtió en el pilar de la Revolución Industrial del Software y se proyecta hoy en día en diversas nuevas formas de hacer software de calidad con los costos más bajos del mercado y en tiempos que antes eran impensables.

## **1.8. XML.**

[6] Sigla en inglés de Extensible Markup Language (lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

XML es un lenguaje que permite jerarquizar y estructurar la información y describir los contenidos dentro del propio documento, así como la reutilización de partes del mismo. La información estructurada presenta varios contenidos (texto, imágenes, audio, etc.) y formas: hojas de cálculo, tablas de datos, libretas de direcciones, parámetros de configuración, dibujos técnicos, etc. La forma da alguna indicación de qué papel puede jugar el contenido (por ejemplo, el contenido de una sección encabezada con un significado difiere del contenido de una nota a pie de página, lo que significa algo diferente que el contenido de un pie de foto o el contenido de

una tabla de datos). Más o menos todos los documentos tienen la misma estructura.

Los principales usos de XML son los siguientes:

- XML aplicado a los sitios web: permite separar contenido y presentación, y que los mismos datos se puedan mostrar de varias formas distintas sin demasiado esfuerzo.
- XML para la comunicación entre aplicaciones: representación de los datos muy simple, fácil de transmitir por la red, estándar. En los últimos tiempos este uso se está haciendo muy popular con el surgimiento de los Servicios web.
- XML para la configuración de programas: representación de los datos simple y estándar, en contraposición con los crípticos formatos propietarios.

## **1.9. DESARROLLO DE SOFTWARE PARA DISPOSITIVOS MÓVILES**

[7] En la actualidad gracias al avance tecnológico y al abaratamiento de los dispositivos de procesamiento de información y comunicación se ha resaltado la relevancia de la ubicuidad, La ubicuidad es el concepto de estoy aquí y necesito procesarlo ya, además el concepto también se refiere a la movilidad.

En diversos campos de aplicación en el entorno es evidente la necesidad de llevar la información a todas partes, por ejemplo profesionales tales como los agrónomos o ingenieros ambientales, por nombrar algunos casos, realizan gran parte de su trabajo fuera de su oficina, los procesos que desarrollan se optimizarían en gran medida si contaran con sistemas de información que apoyen su gestión y que puedan ser instalados en su celular o en un lpaq.

Dentro del trabajo con aplicaciones móviles se puede tratar tres grandes procesos, el primero es el trabajo que realiza el cliente en el dispositivo móvil, el segundo la aplicación servidor que recibirá los datos de los dispositivos móviles, y por último el proceso de sincronización entre el dispositivo móvil del cliente y la aplicación servidor.

**Figura 2 Sincronización entre dispositivo móvil y servidor**



Existen varias técnicas para la sincronización entre la información guardada en los dispositivos móviles y la aplicación del servidor, entre ellas están:

**Slow sync:** En términos generales cuando se va a sincronizar con este protocolo la pda transfiere todos los datos hacia el PC. y compara uno a uno los registros para ver cual ha sufrido cambios, se ha borrado o ha sido adicionado.

**Fast sync:** Este protocolo de sincronización es mucho más rápido que el anterior y se basa en que sólo pasan al servidor los datos que han tenido modificaciones, en otras palabras, los datos que no han sufrido cambios, borrados o adiciones no se sincronizan.

- **Intellisync anywhere:** está dedicado para la sincronización en red de diferentes dispositivos, donde los dispositivos móviles sincronizan con un servidor central.
- **SyncMI (Synchronous Mark Up Language):** el objetivo de este protocolo es la sincronización "abierta" entre dispositivos de diferentes familias. Varias compañías unen esfuerzos para que sus equipos trabajen con este protocolo de sincronización. Se puede decir que se busca un protocolo de sincronización Universal.
- **Per-To-Per:** es otro protocolo de sincronización donde todos y cada uno de los dispositivos se sincronizan sin la necesidad de un computador central o servidor como ocurre en el caso del protocolo Intellisync anywhere.

## 2. ANTECEDENTES

En este punto se describen las características, ventajas y desventajas de algunas herramientas similares a VSHOP CMS.

### 2.1. AB-SHOP 3.

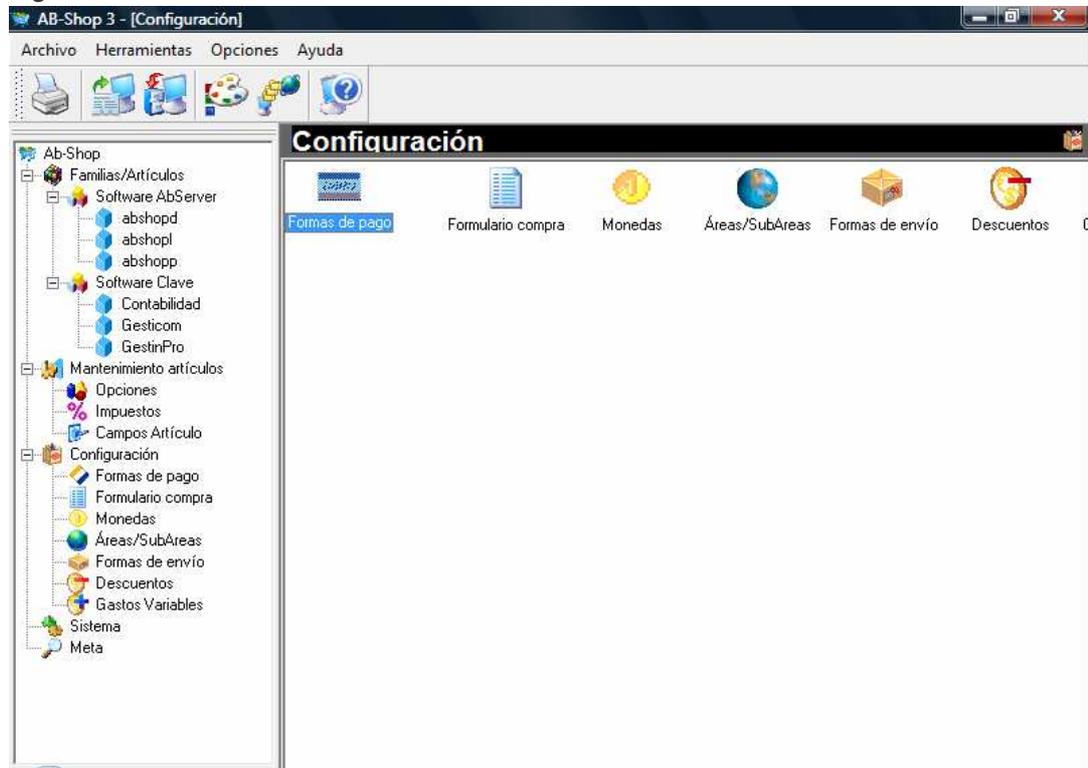
[8] Generador de Contenidos de licencia propietaria con un precio que varía de 750 euros a 2000 euros, es una herramienta desarrollada en java<sup>tm</sup>, AB-Shop dispone de dos módulos:

Módulo AB-Shop en este caso es el módulo administrador de tienda: donde se realiza la configuración de la tienda; opciones de pago, generación de familias (catálogos), importación y generación de artículos, administración de características de artículos.

El módulo administrador AB-SHOP se divide en un árbol que define las principales actividades que puede realizar el administrador.

- Familias y Artículos: Permite crear catálogos (familias) que administraran la parte de tipo de los artículos y la creación de artículos.
- Mantenimiento artículos: definición de los impuestos y las características de los artículos como ejemplo el color la talla o el tipo.
- Configuración: Definición y modificación de las formas de pago, la creación del formulario de pago, los tipo de modernas, las áreas de ventas, las formas de envío y los descuentos,
- Sistema: configuración de la herramienta FTP para la actualización de la tienda.

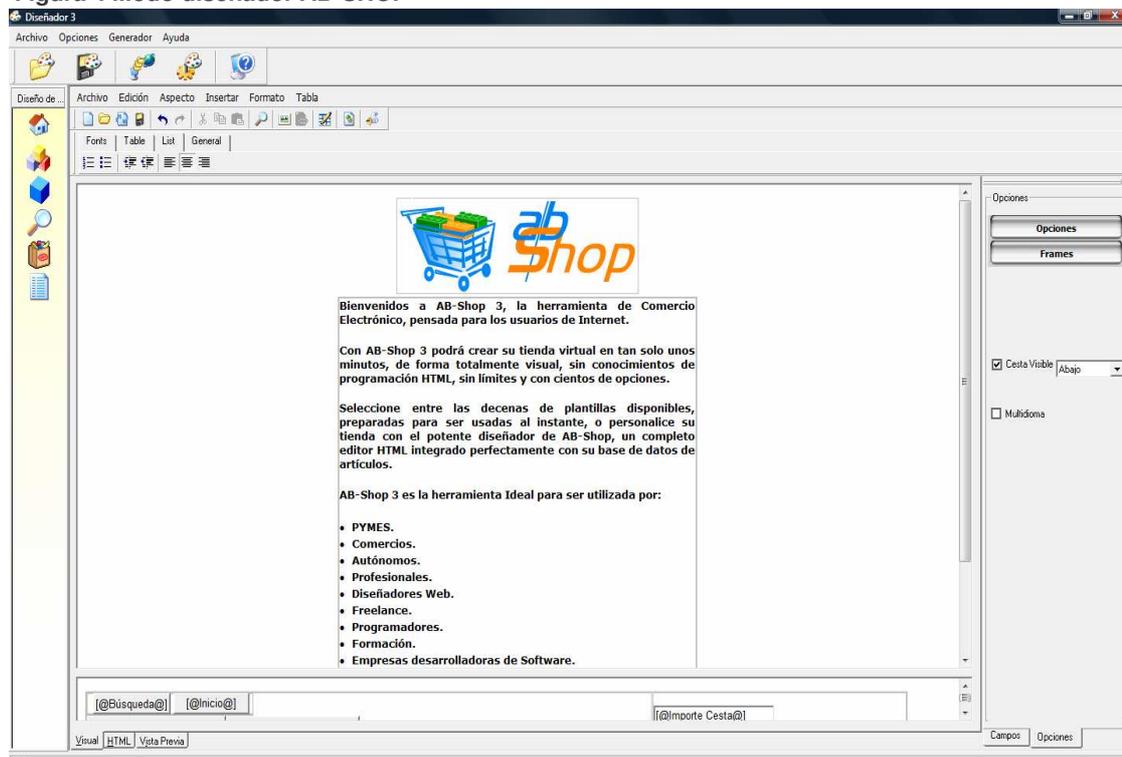
Figura 3 Administrador AB-SHOP



AB-SHOP

- Módulo Diseñador: Módulo donde se encuentra un editor avanzado de HTML que permite al usuario elegir plantillas predefinidas o diseñar sus propias plantillas para, elegir la apariencia de la tienda. AB-SHOP costa de un programa de comunicación FTP que permite actualizar los archivos de la tienda, solo los modificados o la tienda en su totalidad.

Figura 4 Modo diseñador AB-SHOP



## AB-SHOP

AB-SHOP es una herramienta costosa, con poca usabilidad, no consta de un administrador de clientes, si un cliente realiza una compra o pedido la tienda no guarda sus datos para próximos pedidos o creación de cuentas de usuario, no posee administración de productos en oferta o productos de novedad. AB-SHOP solo es capaz de administrar una única tienda, si es necesario mudar la tienda a otro servidor, es necesario comprar una nueva licencia de uso, AB-SHOP es una herramienta no robusta, aunque se concentran en que un usuario no experto realiza la tarea con facilidad, su costo para uso incrementa la imposibilidad de que la tienda se independice y necesita de los creadores de AB-SHOP para poder generar tiendas de calidad o pagar desarrollo a estos.

## 2.2. JOOMLA.

[9] Joomla es un sistema gestor de contenidos de código abierto que nació en el 2005 como una visión open source de Mambo, la última versión 1.5 está totalmente escrita en PHP 5, orientado a objetos, este administrador de contenidos trabaja sobre una base de datos MYSQL.

Joomla nace con una división el 17 de Agosto de 2005 del grupo de trabajo de Mambo. La corporación Miro formó una organización sin ánimo de lucro con el propósito inicial de fundar el proyecto y protegerlo de pleitos. El grupo de desarrollo no estuvo de acuerdo con la nuevas clausulas y afirmo que iban en contra de la comunidad y la filosofía open source, y así nace Joomla como una respuesta del grupo de desarrollo a las nuevas políticas de los directivos de la empresa Miro.

En Joomla se incluyen características como: hacer caché de páginas para mejorar el rendimiento, indexamiento web, versiones imprimibles de páginas, flash con noticias, blogs, foros, polls (encuestas), calendarios, búsqueda en el sitio web, e internacionalización del lenguaje. Su nombre es una pronunciación fonética para anglófonos de la palabra swahili jumla que significa "todos juntos" o "como un todo". Se escogió como una reflexión del compromiso del grupo de desarrolladores y la comunidad del proyecto.

[10] Joomla como todo Sistema Gestor de Contenidos costa de dos partes:

Un administrador que permite al usuario generar o modificar los contenidos de su página, realizar administración de usuarios, y definir los contenidos de página como blogs, menús, artículos y otros

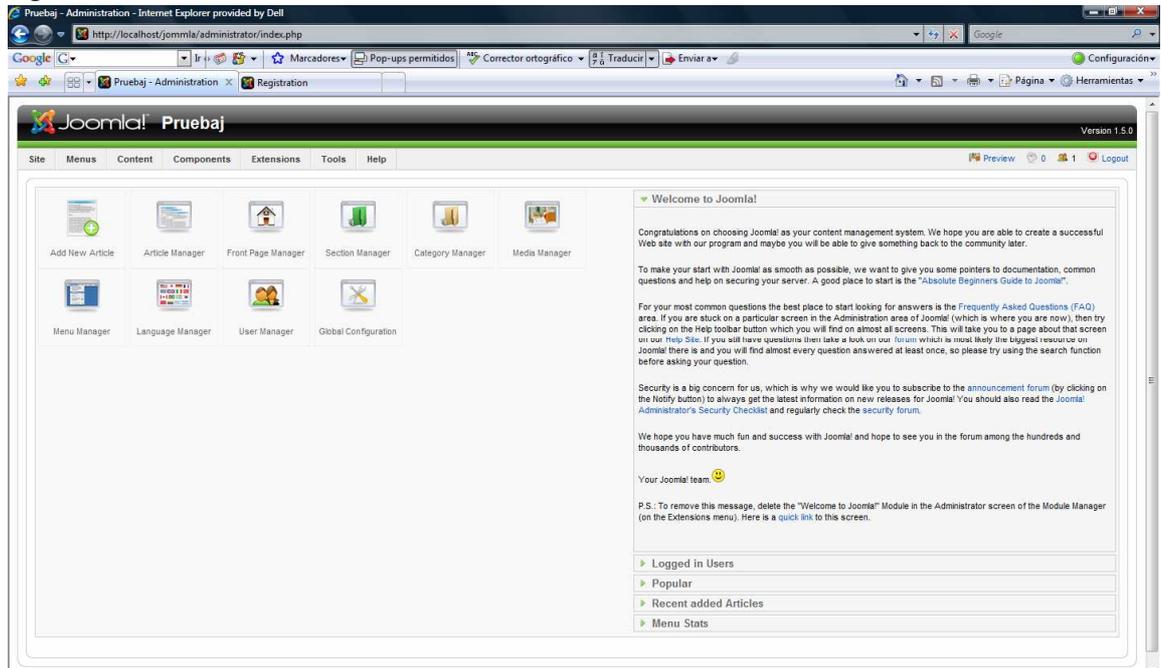
La administración puede realizarse por varios usuarios, según el tipo de usuario se definen los permisos de administración así:

- Los usuarios externos tienen un Menú del usuario cuando se registran. Se dividen en: **Registrados**, Autores, Editores y Publicadores, según puedan incluir enlaces o escribir, modificar y publicar los contenidos.
- Los usuarios internos administran la web desde la Administración y se dividen en: Managers, que tienen acceso a todos los controles de contenidos, Administradores, que son los que configuran la web y añaden componentes, y (3) Súper administradores, que son los que tienen los máximos privilegios.

Las partes básicas del Administrador son:

- Las secciones son los primeros elementos en que se dividen los contenidos textuales.
- Las secciones se dividen en categorías, que también son obligatorias para incluir un contenido textual.
- Una vez que se hayan creado las secciones y las categorías ya se pueden escribir los artículos. Hay que distinguir entre los artículos normales, y los estáticos, que se escriben principalmente para ser incluidos en menús de forma aparte.
- Los componentes y los mambots permiten extender los contenidos, que ya no serán sólo artículos, sino encuestas, enlaces de interés, noticias externas, descarga de archivos, contactos, y otros elementos que se pueden instalar.
- Los módulos permiten organizar los contenidos en la parte externa de la web.
- Las plantillas de diseño (templates), con las que se puede modificar al gusto el diseño externo de la web, con la posibilidad de importar más desde otras webs.
- El gestor multimedia permite subir y ordenar las imágenes y otro tipo de archivos,
- Las herramientas de configuración.

Figura 5 Administrador de Joomla

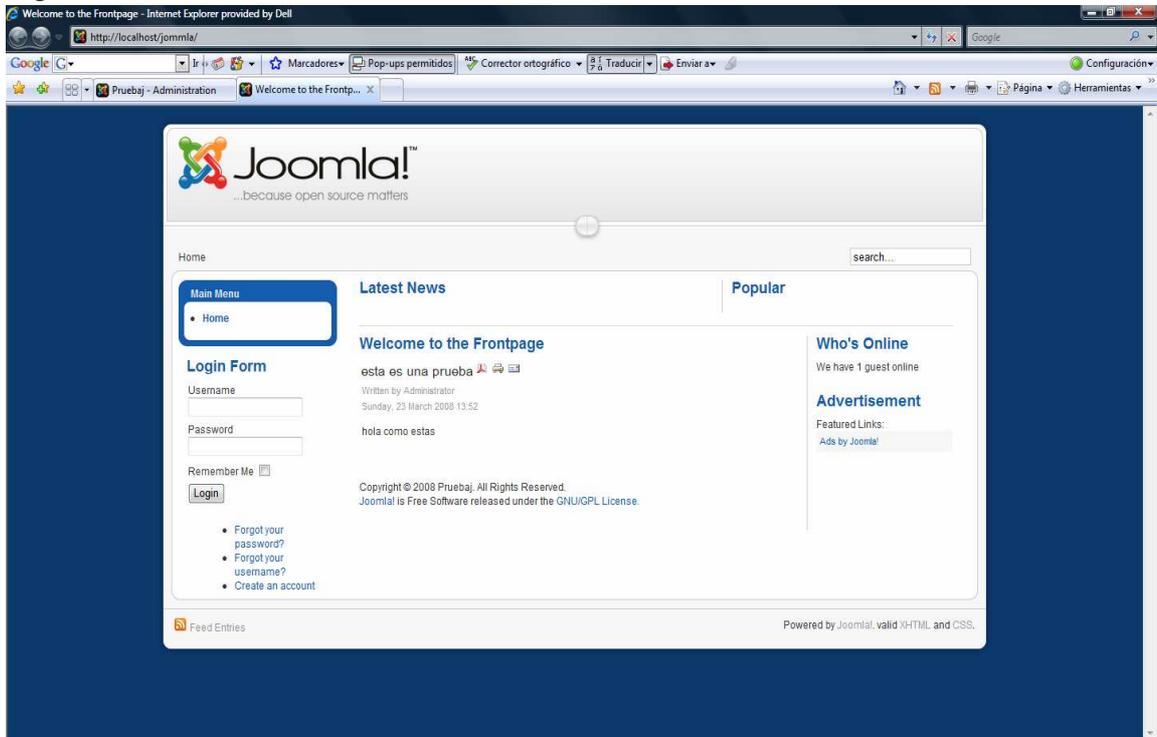


Wikipedia

Y la parte que mira el cliente que es el sitio Web como tal, dividida en las siguientes partes:

- Una columna central donde se incluyen los contenidos.
- Una o dos columnas laterales donde aparecen los menús y los módulos.
- La cabecera, donde aparece el nombre del portal y, a veces, otros menús.
- El pie, donde se pueden leer los créditos y otros textos de copyright.

Figura 6 Sitio Web realizado mediante Joomla



Wikipedia

Joomla es la herramienta más popular en el mercado ya sean de open source o propietarias, es la mejor herramienta que existe como sistema gestor de contenidos para sitios web, es un sistema muy amigable para los usuarios y la experiencia de sus desarrolladores sobre el ambiente anterior Mambo, logra que sea la herramienta más aventajada en la creación de sitios web con sistemas gestores de contenidos. Joomla como Mambo son sistemas que generar sitios Web, pero no tienen componentes especializados para generar tiendas virtuales, que manejen, productos, servicios, ventas, carritos de compras o administraciones de clientes, existen paquetes adicionales desarrollados por la comunidad, pero sus funcionalidades son mínimas. Joomla no maneja ambientes que le permitan al usuario ver un sistema y no una página web, no hace uso de tecnologías como *Ajax*, utilizando aun recarga de páginas para realizar cambios, dejando a un lado la ideología de la Web 2.0.

### 2.3. MAMBO.

[11] Mambo Open Source es un sistema gestor de contenidos antecesor de Joomla, generado por la empresa australiana Miro, para la creación de sitios web, desarrollada en PHP y como gestor de Base de Datos MySQL [1].

Mambo permite la creación de sitios web dinámicos de una manera fácil y rápida, sin tener conocimientos previos de herramientas o plataformas orientadas a la web. Se basa en el desarrollo de templates y CSS para la apariencia y administración de la tienda.

Las características de Mambo son:

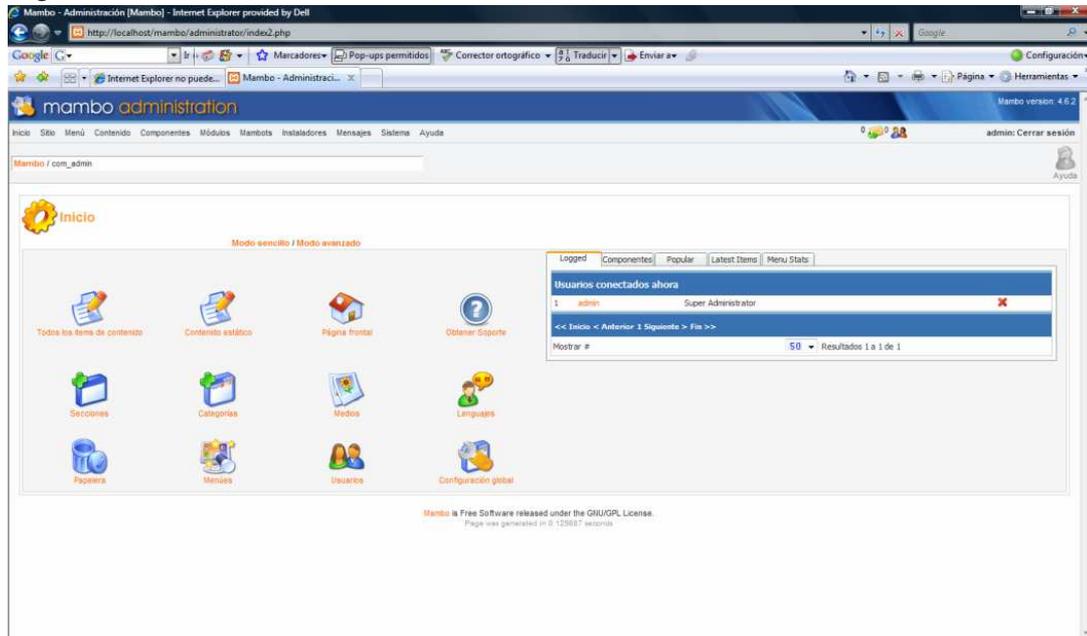
- Base de datos movida por los estándares PHP/MySQL.
- Módulo de seguridad multinivel para usuarios/administradores.
- Noticias, productos o secciones totalmente editables y configurables.
- Sección de temas que pueden ser enviados por los usuarios registrados.
- Plantillas o Templates totalmente configurables incluyendo menú central y bloques a izquierda y derecha, según se quiera.
- Soporte de subida de imágenes para incorporar a la librería y para usar en cualquier parte del sitio web. Foros dinámicos y encuestas con vista de resultados.
- Administración: Generador automático de noticias en titulares.
- Envío de noticias, artículos, FAQ's y enlaces por parte de los usuarios registrados.
- Jerarquía de Objetos: cuantas secciones, departamentos, divisiones y páginas se quiera.
- Librería de Imágenes - Almacena todos los archivos PNG's, PDF's, DOC's, XLS's, GIF's y JPEG's online para un fácil uso.
- Búsqueda Automática de Directorios.
- Manejador y Sindicación de Noticias. Más de 360 grupos de noticias de todo el mundo donde escoger y publicar en tu web. Manager de Archivos. Almacena tus artículos antiguos y publícalos de nuevo cuando quieras.
- Posibilidad de impresión, convertidor a PDF o envío por email de cualquier noticia o artículo publicado.
- Editor de texto.
- Editor de Usuarios.

- **Manejador Encuestas.** Posibilidad de publicar encuestas distintas en cada página.
- **Módulos configurables.** Descargas de nuevos módulos.
- **Manager de Plantillas.** Descarga de nuevas plantillas.
- **Pre visualización de Plantillas.** Puedes ver el aspecto de la página al cambiar de plantilla antes de actualizar los cambios.
- **Administrador de banners.**

**Administrador:** De aspecto muy similar al de Joomla, posee opciones de manejo como:

- **Página Principal:** Define los contenidos de la pagina inicial del sitio.
- **Secciones:** La división de las páginas del sitio, noticias, preguntas frecuentes, encuestas entre otras.
- **Categorías:** Partes de las secciones, noticias más frecuentes, novedades etc.
- **Menús:** administrador de todos los menús del sitio.
- **Usuarios:** administración de los usuarios que pueden realizar cambios en la página: **Manejadores** que tienen acceso a todos los controles de contenidos, **Administradores**, que son los que configuran la web y añaden componentes, y **Súper administradores**, que son los que tienen los máximos privilegios y la administración de todos los demás usuarios.
- **Administración de medios:** realiza la gestión de todos los archivos adicionales como imágenes, sonidos, animaciones etc., que serán utilizados después en la construcción del sitio.
- **Contenido estático:** gestión de los artículos que aparecerán en el sitio.
- **Componentes:** la administración de **Banners**, **polls** (encuestas), **comentarios**, **usuarios**, **Noticias**.

Figura 7 Administrador de Mambo

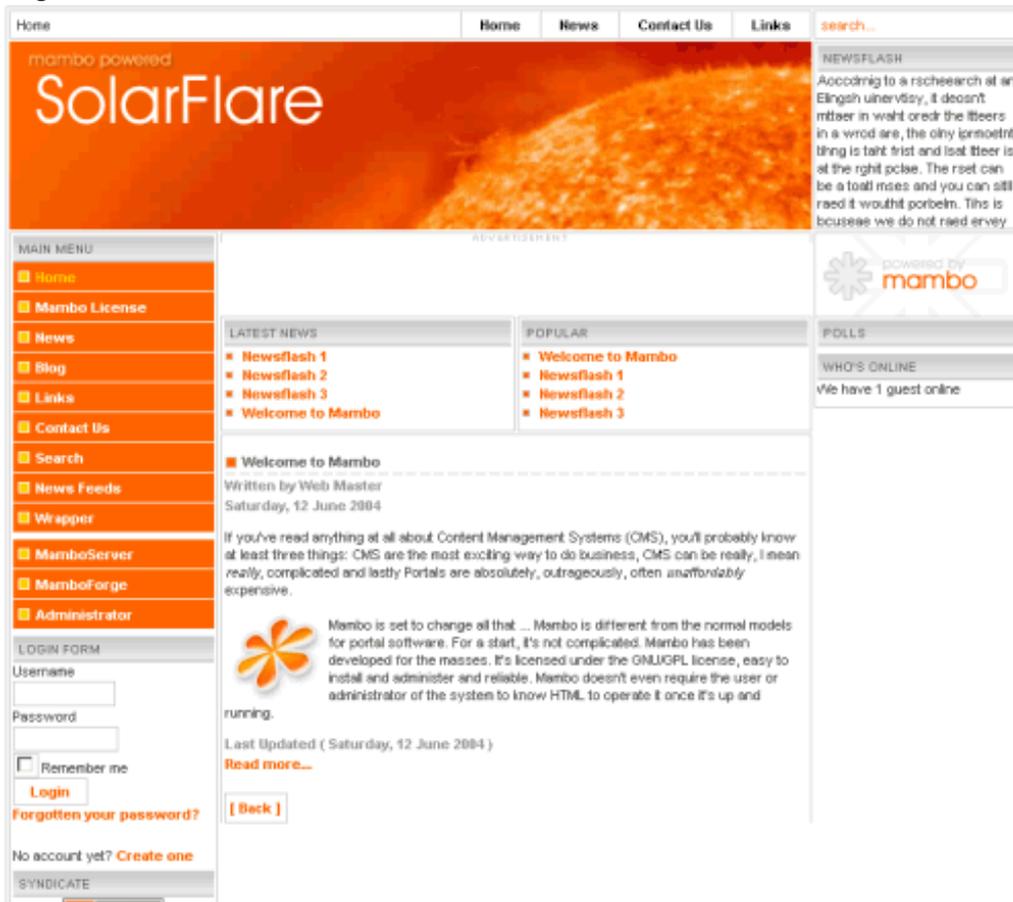


Wikipedia

**El sitio web:** Donde aparece el contenido desarrollado por los usuarios del administrador, dependiente totalmente del template elegido y con divisiones de secciones como noticias o los más populares y el registro de usuarios del sitio (Figura 8).

Mambo es una herramienta muy similar a Joomla, ya que Joomla es derivada de mambo, es una herramienta igualmente robusta, y tiene muchas de las funciones de Joomla, pero Joomla es aun más amigable que mambo, la desventaja principal de esta herramienta es que no está hecha para la generación de tiendas virtuales, no posee los módulos ni mecanismos para relación la administración de artículos, catálogos, y carrito de compras, es también una herramienta que depende de el reenvío de los datos para poder mostrarle al usuario los resultados, no es una herramienta con ideología Web 2.0 ni se han utilizado herramientas como *Ajax* para su desarrollo.

Figura 8 Sitio desarrollado en Mambo



Wikipedia

## 2.4. OSCOMMERCE: (OPEN SOURCE COMMERCE).

[12] OsCommerce es un gestor de contenidos que nació en el año 2000, es Open Source y está licenciado sobre GPL, fue desarrollado sobre plataforma PHP en su versión 4.0, con Mysql como servidor de Base de Datos, y Apache como servidor HTTP.

Como todo gestor de contenidos OsCommerce posee el módulo Front-end o en este caso Catálogo de productos: Es la tienda virtual en sí, donde se puede revisar los catálogos, realizar las compras, hacer compras por la canasta de compras, registrar clientes, realizar pedidos. Es el módulo que ve el cliente y que permite realizar la interacción con este.



Figura 9 Módulo que ve el cliente y que permite realizar la interacción con este.



El módulo Back-end en este caso el módulo de administración permite:

Creación de catálogos y productos, insertar fabricantes, definir características de los productos, ofertas, tipo de pagos, idiomas; Oscommerce soporta como base tres idiomas: inglés, español y alemán, pero se pueden descargar más de su sitio web, definición de modernas e impuestos.

Figura 10 Módulo administrativo de osCommerce



OsCommerce soporta los siguientes tipos de Pago:

- Authorize.net
- Tarjeta de Crédito
- Contra Reembolso
- iPayment
- Cheque/Transferencia Bancaria
- NOCHEX
- Pay pal
- 2Checkout
- PSiGate
- SECPay

Características principales de OsCommerce son:

### **Generales**

- Los pedidos, clientes y productos se almacenan en una base de datos de fácil consulta vía administración-web.
- Los clientes podrán comprobar el histórico y el estado de sus pedidos una vez registrados
- Los clientes pueden cambiar sus datos de perfil de usuario desde su apartado cliente.
- Múltiples direcciones de envío por usuario, para regalos por ejemplo
- Búsqueda de productos.
- Posibilidad de permitir a los usuarios valorar los productos comprados, además de comentarlos.
- Posibilidad de implementar un servidor seguro (SSL).
- Puede mostrar el número de productos en cada una de las categorías.
- Lista global o por categoría de los productos más vendidos y más vistos.
- Fácil e intuitiva navegación por categorías.
- Plataforma multi-idomas, por defecto estarán disponibles el español, inglés y alemán.

### ***Producto***

- Relaciones dinámicas entre productos.
- Descripciones de productos basadas en HTML.
- Generación automática de productos especiales.
- Controla la posibilidad de mostrar o no en la tienda virtual los productos agotados.
- Posibilidad de ofrecer a los usuarios la suscripción a una noticias de novedades.

### ***Pagos***

- Medios de pago offline (transferencias, cheques, ingresos, etc.).
- Muchos medios de pago online (E-Pagado, PayPal, TPV virtual, etc.).
- Posibilidad de deshabilitar algunos medios de pago según la zona geográfica del usuario.
- Posibilidad de añadir el IVA por zonas geográficas y por producto

### ***Envíos***

- Precios de envío por peso, destino y precio.

- Precios reales disponibles en tiempo real para algunos operadores (UPS, FedEx).
- Envío gratuito según importe del pedido y destino
- Posibilidad de deshabilitar determinados servicios de envío en función de zonas geográficas.

Las falencias de OsCommerce, radican en su naturaleza Open Source, OsCommerce es una herramienta desarrollada para que crezca con los aportes de la comunidad, no es una herramienta confiable, sus principales problemas son se seguridad , para la instalación y ejecución de OsCommerce, solicita la habilitación en la ejecución del lenguaje PHP la habilitación de variables globales, por cuestiones de seguridad esta habilitación no es buena, ya que permite que se ejecuten por todos los scripts del sitio el registro e interpretación de variables como captura de cookies, envíos posts y gets, esto abre un gran hueco de seguridad permitiendo que usuarios mal intencionados logren ejecutar código sobre el sitio.

Con pocos recursos para un usuario inexperto, la herramienta fue desarrollada para que la comunidad la modifique, su documentación solo abarca explicaciones de cómo modificar el código fuente y sus presentaciones, esto obliga a los usuarios de OsCommerce ser conocedores de la plataforma PHP, y lenguajes como HTML y JavaScript, así que osCommerce es una herramienta para desarrolladores expertos, La no utilización de CSS para el cambio de aparecía de la tienda, hace que esta siempre se vea igual, para realizar cambios en apariencia es necesario conocer los archivos bases de OsCommerce, aunque entrega una herramienta de Administración de Archivos, es de muy difícil navegación no permite realizar búsquedas y su editor es muy básico, sin diferenciación de código entre los tags PHP y HTML, En OsCommerce la creación de catálogos y productos es eliminara pero solo soporta una tienda para administración. Por último OsCommerce, en su Back-end es una herramienta que todavía hace uso del reenvío de datos a páginas, su navegación hace que el usuario tenga que ver por cada una de sus actividades la recarga de la página que este observando, no hace uso de conceptos como la web 2.0 y herramientas como Ajax.

### 3. HERRAMIENTAS UTILIZADAS

En este punto se describe cada una de las herramientas utilizadas para la realización de VSHOP CMS.

#### 3.1. WEB 2.0.

[13] [14] La Web 2.0, aunque muchos la han catalogado como una palabra de moda o una nueva política de Marketing, es más que eso, la web 2.0, se puede definir como una ideología de desarrollo que toma a internet como una plataforma, que busque que toda la comunidad, que opera en internet sea quien controle el contenido de esta, para esto las aplicaciones web deben permitirle al usuario ser el visionario de la aplicación, cambiar el paradigma de que las paginas son solo estáticas, y que los usuarios de estas no pueden realizar actividades de cambio sobre ellas, los sitios web son ya aplicaciones como tal, prestan servicios, y buscan formas de que usuario final sea quien haga que el sitio crezca y se enriquezca, el conocimiento es el poder, en este caso el control de la información es el poder, pero la información en internet no pertenece a una sola persona, la información es pública, todos pueden trabajar sobre ella, encontrar respuestas a sus preguntas, y dar respuestas a preguntas de otros, la web 2.0 es una ideología de cambio, de busca que internet sea quien preste los servicios y quien le dé al usuario la capacidad de ser el dueño de su propio entorno, busca que los sitios web sean aplicaciones que superen las aplicaciones de escritorio. Y que el usuario pueda interactuar con estas, de forma muy amigable, sin necesidad de pensar que está viendo un sitio web si no que es una aplicación que corre sobre internet; Internet es la plataforma.

Las tecnologías que soportan la web 2.0:

- Transformar software de escritorio hacia la plataforma del web.
- Respeto a los estándares como el XHTML.
- Separación de contenido del diseño con uso de hojas de estilo.
- Sindicación de contenidos.
- Ajax (javascript asincrónico y XML).
- Uso de Flash, Flex o Lazlo.
- Uso de Ruby on Rails para programar páginas dinámicas.

- Utilización de redes sociales al manejar usuarios y comunidades.
- Dar control total a los usuarios en el manejo de su información.
- Proveer APIs o XML para que las aplicaciones puedan ser manipuladas por otros.
- Facilitar el posicionamiento con URL sencillos.

Al final la ideología web 2.0 es la forma de que la red se convierta en la herramienta de todos y para todos, los usuarios solo quieren ver lo que necesitan, este deseo y derecho hace que internet se publica y de propiedad total de los usuarios, y la información que este en ella es administrada por los usuarios; ese es la web 2.0 una forma que internet solo sea del usuario y de nadie más.

### **3.2. PHP.**

[15] Es un lenguaje de programación interpretado, que puede ser embebido en HTML, PHP es el acrónimo “Hypertext Pre-processor”, fue desarrollado para ser una herramienta de desarrollo de páginas web, pero debido al auge de la web 2.0 a evolucionada y puede ser usado para desarrollar aplicaciones, sean web o de escritorio.

El lenguaje posee una gran similitud con lenguajes anteriores como c, c++, java, gracias a esto php es una herramienta de muy fácil entendimiento, y capaz de ser muy poderosa para el desarrollo de sistemas y ambientes web.

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos.

PHP es una herramienta multiplataforma debido a que solo necesita un navegador, y las ejecuciones de los scripts se hacen en un servidor que soporte script php, no necesita que se escriba código de adaptación para cada sistemas operativo, así que php es un herramienta de fácil manejo para todo tipo de aplicaciones que necesiten adaptarse a diferentes tipo de sistemas operativos.

PHP es soportado por la mayoría de servidores entre estos esta: Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. PHP

tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI.

**Tabla 1 Versiones y cambios PHP**

| Versión                | Fecha                   | Cambios más importantes  |
|------------------------|-------------------------|--|
| PHP 1.0                | 8 de Junio de 1995      | Oficialmente llamado "Herramientas personales de trabajo (PHP Tools)". Es el primer uso del nombre "PHP".                    |
| PHP Versión 2 (PHP/FI) | 16 de Abril de 1996     | Considerado por el creador como la "más rápida y simple herramienta" para la creación de páginas webs dinámicas .            |
| PHP 3.0                | 6 de Junio de 1998      | Desarrollo movido de una persona a muchos desarrolladores. Zeev Suraski y Andi Gutmans reescriben la base para esta versión. |
| PHP 4.0                | 22 de Mayo de 2000      | Se agregan avanzadas de dos etapas analizar/ejecutar la etiqueta-análisis sistema llamado entorno motor Zend.                |
| PHP 4.1                | 10 de Diciembre de 2001 | Introducidas las variables superglobals (\$_GET, \$_SESSION, etc.)   |
| PHP 4.2                | 22 de Abril de 2002     | Se deshabilitan register_globals por defecto   |
| PHP 4.3                | 27 de Diciembre de 2002 | Introducido la CLI, en adición a la CGI  |
| PHP 4.4                | 11 de Julio de 2005     |  |
| PHP 5.0                | 13 de Julio de 2004     | Motor Zend II con un nuevo modelo de objetos.  |
| PHP 5.1                | 25 de Noviembre de 2005 |  |
| PHP 5.2                | 2 de Noviembre de 2006  | Habilitado el filtro de extensiones por defecto  |
| PHP 5.2.4              | 30 de agosto de 2007    |  |
| PHP 5.2.5              | 8 de Noviembre de 2007  | Versión centrada en mejorar la estabilidad (+60 errores solucionados)  |

PHP es una herramienta que ha evolucionado de ser un lenguaje para desarrollo de páginas web a ser una plataforma que soporta la programación orientada a objetos, aunque no con todos sus paradigmas, pero suficiente para ser capaz de

desarrollar aplicaciones y no solo paginas y ver que los nuevos sitios web sean herramientas de servicios, aplicando así las teorías de la web 2.0.

La principal ventaja de PHP es el soporte de muchos tipos de bases de datos y además del estándar ODBC. Como ejemplo podemos tomar:

- Adabas D
- dBase
- Empress
- FilePro (read-only)
- Hyperwave
- IBM DB2
- Informix
- Ingres
- InterBase
- FrontBase
- mSQL
- Direct MS-SQL
- MySQL
- Oracle (OCI7 and OCI8)
- Ovrimos
- PostgreSQL
- Solid
- Sybase
- Velocis
- Unix dbm

### **3.3. JAVASCRIPT.**

[16] JavaScript es un lenguaje creado por Brendan Eich en la empresa Netscape Communications, y su primera aparición fue Netscape Navigator 2.0., es un lenguaje interpretado sobre el navegador, mientras se descargan al ejecuciones HTML también se descargan las operaciones JavaScript, su nombre se deriva de la unión que tuvieron Netscape y Java para publicar y hacer popular el lenguaje. Al contrario que Java, JavaScript no es un lenguaje orientado a objetos propiamente dicho, ya que no dispone de Herencia, es más bien un lenguaje basado en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad.

Para evitar incompatibilidades, el World Wide Web Consortium diseñó el estándar Document Object Model (DOM, ó Modelo de Objetos del Documento en castellano), que incorporan Konqueror, las versiones 6 de Internet Explorer y Netscape Navigator, Opera versión 7, y Mozilla desde su primera versión, para que todos los navegadores pudieran interpretar el código JavaScript.

### 3.4. AJAX.

[17] AJAX, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

AJAX es una combinación de cuatro tecnologías:

- XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementación JavaScript, para mostrar e interactuar dinámicamente con la información presentada.
- El objeto XMLHttpRequest para intercambiar datos asíncronicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios.
- XML es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML pre formateado, texto plano, JSON y hasta EBML.

Navegadores que permiten Ajax:

- Navegadores basados en Gecko como Mozilla, Mozilla Firefox, SeaMonkey, Camino, K-Meleon, Flock, Epiphany, Galeon y Netscape versión 7.1 y superiores

- Microsoft Internet Explorer para Windows versión 5.0 y superiores, y los navegadores basados en él
- Navegadores con el API KHTML versión 3.2 y superiores implementado, incluyendo Konqueror versión 3.2 y superiores, Apple Safari versión 1.2 y superiores, y el Web Browser for S60 de Nokia tercera generación y posteriores.
- Opera versión 8.0 y superiores, incluyendo Opera Mobile Browser versión 8.0 y superiores.

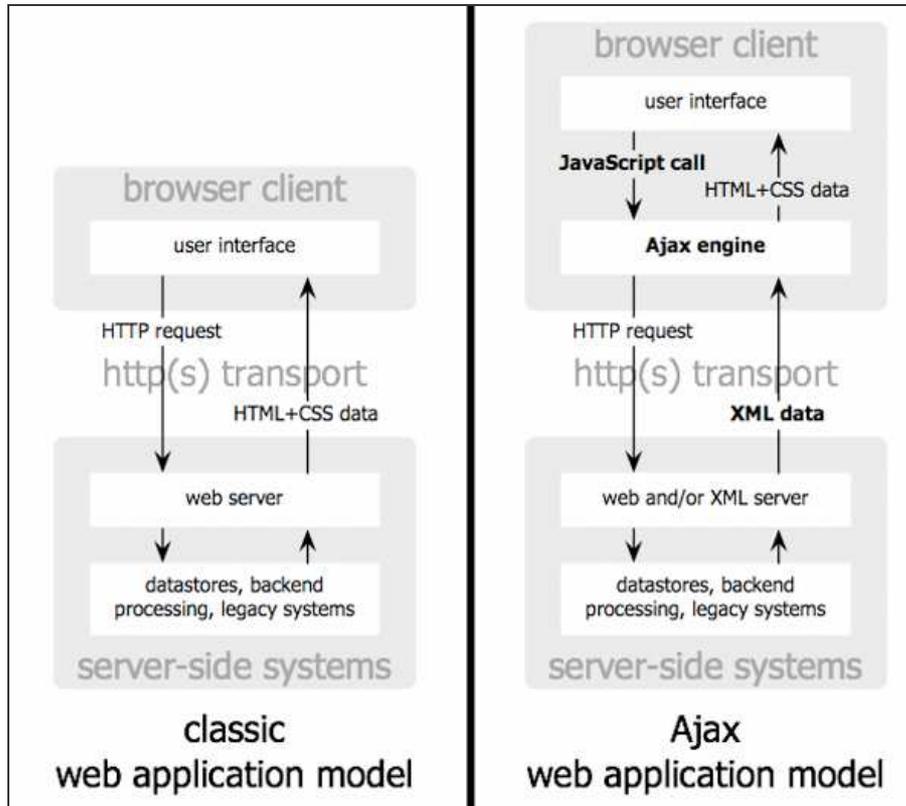
Navegadores que no permiten Ajax:

- Opera 7 y anteriores
- Microsoft Internet Explorer para Windows versión 4.0 y anteriores
- Microsoft Internet Explorer para Macintosh, todas las versiones
- Dillo
- Navegadores basados en texto como Lynx y Links
- Navegadores para incapacitados visuales (braille)

Ajax es una herramienta capaz de otorgar a los desarrolladores, formas de que sus ambientes sean aun mas amigables para los usuarios, sin la necesidad de las recargas y reduciendo el uso del ancho de banda ya que los llamados a operaciones al servidor hacen asíncronamente y no hay necesidad de enviar toda la pagina si no solo las modificaciones pertinentes a las actividades del usuario.

Por otra parte Ajax, genera tiempos de latencia en la recarga de los datos, así que deben crearse interfaces que le muestren al usuario el que se ejecuta un procesamiento, debido a que esta espera puede generar en el usuario impaciencia y hacer que ejecute otras operaciones indebidas.

Figura 11 Comparación entre una aplicación clásica y una aplicación Ajax



Wikipedia

### 3.5. XAJAX.

[18] Xajax es una biblioteca de código abierto para PHP que permite crear de manera fácil y simple aplicaciones Web basadas en AJAX usando además HTML, CSS, y Javascript. Las aplicaciones desarrolladas con Xajax pueden comunicarse asincrónicamente con funciones que se encuentran del lado del servidor y así actualizar el contenido de una página sin tener que recargarla nuevamente; el desarrollo de Ajax requiere un gran conocimiento del trabajo con javascript y XML, así que un forma fácil de realizar estas tareas es utilizando la biblioteca xajax, donde la operaciones se realizan fácilmente y no necesitan gran cantidad de código para realizar tareas como envió asíncrono de datos o modificaciones de CSS.

En un principio se crea una instancia de objeto Xajax (*xajax object*). Este objeto manejará todo el procesamiento a través de Xajax. En segundo lugar debemos

registrar todas las funciones que hemos definido previamente en el objeto Xajax, esto se puede hacer usando el método `xajax->registerFunction()`. Finalmente todas las respuestas serán procesadas utilizando el método `xajax->processRequest()`.

Xajax es compatible con Firefox, Mozilla, Internet Explorer, Opera y Safari, xajax puede actualizar ccs, botones, contenido HTML, casillas de verificación etc. Cada función registrada para ser accesible a través de Xajax puede tener distintos tipos de petición.

### **3.6. CSS: (CASCADING STYLE SHEETS).**

CSS: (*Cascading Style Sheets*) es un lenguaje que se usa para definir la presentación y apariencia de un documento HTML o XML (y por extensión en XHTML).

Existen tres tipos formas de aplicar estilos a unos documentos:

Una hoja de estilo externa, que es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página Web. Esta es la manera de programar más potente, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página.

Una hoja de estilo interna, que es una hoja de estilo que está incrustada dentro de un documento HTML. (Va a la derecha dentro del elemento `<head>`). De esta manera se obtiene el beneficio de separar la información del estilo, del código HTML propiamente dicho. Se puede optar por copiar la hoja de estilo incrustada de una página a otra, (esta posibilidad es difícil de ejecutar si se desea para guardar las copias sincronizadas). En general, la única vez que se usa una hoja de estilo interna, es cuando se quiere proporcionar alguna característica a una página Web en un simple fichero, por ejemplo, si se está enviando algo a la página web.

Un estilo en línea, que es un método para insertar el lenguaje de estilo de página, directamente, dentro de una etiqueta HTML. Esta manera de proceder no es excesivamente adecuada. Al incrustar el formateo dentro del documento de la página Web la descripción de la página, a nivel de código se convierte en una tarea larga, tediosa y poco elegante de resolver el problema de la programación de la página. Este modo de trabajo se podría usar de manera ocasional si se pretende aplicar un formateo con prisa, al vuelo.

Las hojas de estilo proporcionan un control centralizado de la presentación del sitio web, si se maneja las hojas de estilo en un archivo independiente el cambio de apariencia y las actualizaciones del sitio serán sencillas y prácticas. Las hojas de estilo proporcionan independencia al usuario del sitio, los navegadores pueden ejecutar hojas de estilo personalizadas por el usuario, así el usuario puede ver el sitio como él lo desea ver, colores tamaño de letra entre otros. Para los desarrolladores, crea independencia y facilidad de modificación de código, haciendo más entendible y con menos líneas los scripts HTML o XML.

### **3.7. JAVA ME.**

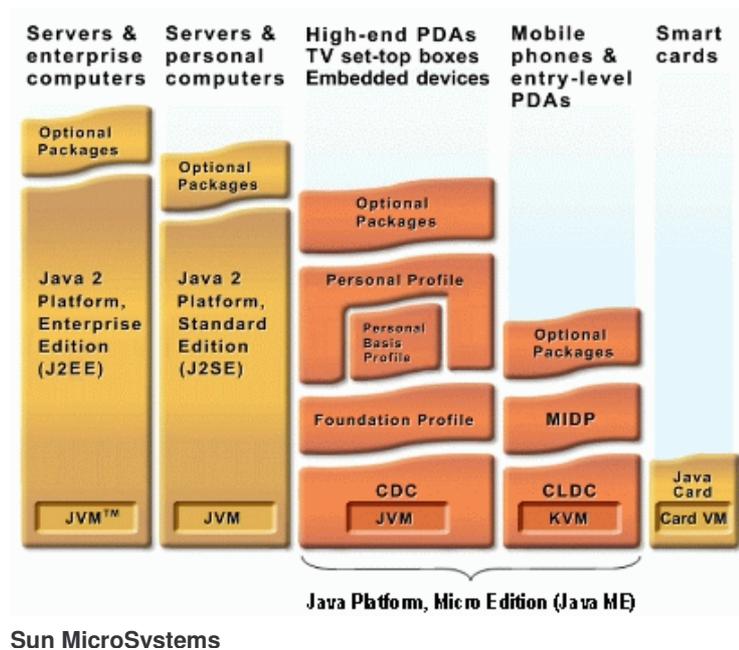
[19] Java Micro Edition es la plataforma que creó Sun Microsystems para el desarrollo de aplicaciones para dispositivos móviles, a fin de hacer frente ante las limitaciones de estos dispositivos en su parte física, Java Micro Edition colecciona tecnologías y especificaciones que le permiten construir aplicaciones según las restricciones de los equipos y las necesidades de los clientes, sin pensar en buscar soluciones en las grandes tecnologías como JSE y JEE, esto ofrece flexibilidad ante las cambiantes características del mercado y de las necesidades de los clientes.

La definición de la tecnología J2ME se basa en tres conceptos claves:

- Configuración: Que especifica las bibliotecas básicas y la máquina virtual que se necesita para la ejecución de las aplicaciones JME.
- Perfil: Es un conjunto de APIs que apoyan la configuración elegida y aumentan su capacidad y sus características
- Paquetes Opcionales: Son APIs que se basan en especificaciones y que enriquecen a un más la Configuración pero tienen características especializadas para las aplicaciones, como por ejemplo los APIs que manejan las conexiones Bluetooth, el envío de mensajes SMS o características multimedia. The Java Platform
- Estos conceptos permiten a JME que los recursos necesarios para cada aplicación y las limitaciones de los equipos no interfieren con el desarrollo, siendo que el desarrollador elegirá que configuración, perfil y paquetes opcionales, se adaptan de manera más eficiente a las características de su aplicación y a las restricciones de su equipo, sin necesidad de tener que cargar todo el conjunto de JME.

La Figura muestra la plataforma Java y donde se ubica la tecnología Java Micro Edition.

Figura 12 Plataforma Java

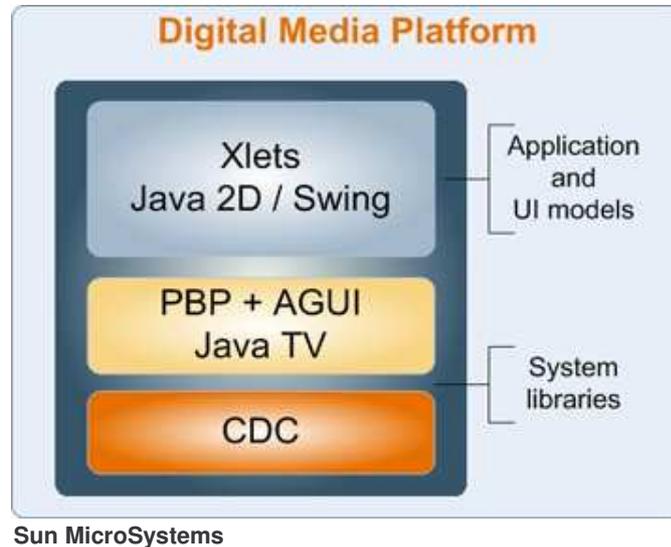


**3.7.1. CDC (Connected Device Configuration).** La configuración CDC es la más grande de las dos que abarca JME, fue desarrollada para soportar equipos con características de procesamiento “alto” en referencia a los dispositivos móviles, como son Teléfonos Inteligentes, Asistentes inteligentes entre otros, así desarrollar aplicaciones que definan características de conexión persistente y características de interfaces tomas de JSE como es El Paquete Swing(Paquete que define la interfaz gráfica de JSE);

Los perfiles que acompañan a CDC son:

- The Foundation Profile (JSR 219)
- The Personal Basis Profile (JSR 217)
- The Personal Profile (JSR 216)

Figura 13 CDC (Connected Device Configuration)



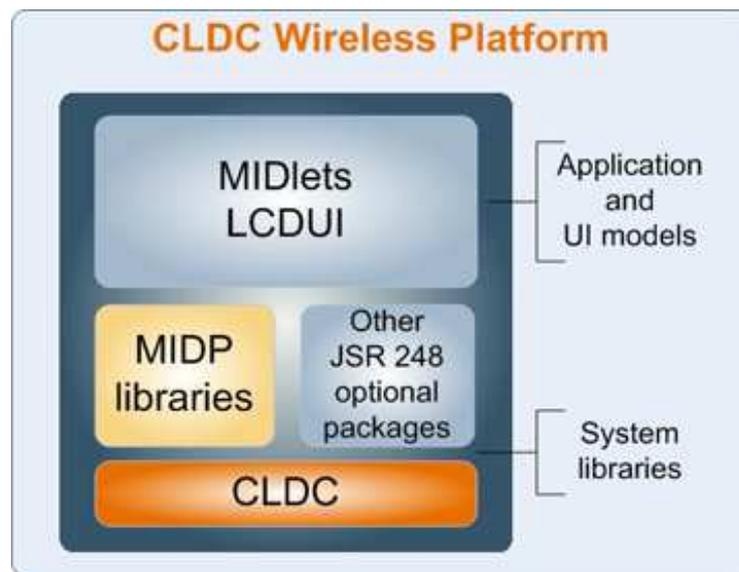
**3.7.2. CLDC (Connected Limited Device Configuration).** Es la configuración desarrollada para contener los teléfonos móviles y dispositivos electrónicos que no poseen gran capacidad y su conectividad es limitada y no persistente, CLDC 1.1 se define gracias a la especificación 132, las especificaciones son documentos que le dan pautas a los fabricantes de cómo implementar la configuración en sus equipos, CLDC 1.1 soporta operaciones de punto flotante y ejecución multi Hilo, el perfil que acompaña a CLDC 1.1 es el perfil MIDP 2.0 que define aplicaciones para juegos, soporte a datos persistentes, características gráficas, trabajo en red y multimedia.

Como mínimo los equipos que soporten CLDC deben tener las siguientes características:

- Procesador 16-bit o 32-bit con una velocidad de reloj de 16MHz o superior
- At least 160 KB of non-volatile memory allocated for the CLDC libraries and virtual machine 160 KB de memoria no volátil asignados para el CLDC máquina virtual y bibliotecas
- At least 192 KB of total memory available for the Java platform Por lo menos 192 KB de memoria total disponible para la plataforma Java

- Low power consumption, often operating on battery power Bajo consumo de energía, a menudo funcionando con la batería
- Connectivity to some kind of network, often with a wireless, intermittent connection and limited bandwidth Conectividad a algún tipo de red, con una frecuencia inalámbrica, conexión intermitente y el ancho de banda limitado.

Figura 14 CLDC (Connected Limited Device Configuration)



Sun Microsystems

### 3.8. APACHE.

El servidor HTTP Apache es un software (libre) servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que originalmente Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado") [20].

Las principales ventajas que ofrece este servidor, son:

- Modular.
- Open Source.

- Multi-plataforma.
- Extensible.
- Popular.

### **3.9. MYSQL.**

Es un sistema de gestión de base de datos relacional, multi hilo y multi usuario, desarrollado por MySQL AB que es propiedad de Sun Microsystems.

MySQL funciona sobre múltiples plataformas, incluyendo:

GNU/Linux, Mac OS X, NetBSD, OpenBSD, OS/2 Warp, Solaris, SunOS, SCO, OpenServer, SCO UnixWare, Windows, OpenVMS.

MYSQL Es muy utilizado en aplicaciones web, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), su popularidad como aplicación web está muy ligada a PHP. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones [21].

### **3.10. XAMPP.**

Paquete que permite instalar de manera fácil y sin grandes complicaciones de configuración un servidor Apache, el interpretador Php y Perl y el sistema Mysql, además de servidores de FTP como ProFTPD ó FileZilla FTP Server, las estadísticas manejadas por Webalizer soportado por plataformas como Windows. XAMPP está formado por un X (la inicial de un sistema operativo), A de Apache, M de MySQL, y P de PHP y Perl [22].

## **4. ANÁLISIS Y DISEÑO VSHOP CMS**

La disciplina de la ingeniería se define como la capacidad de desarrollar elementos innovadores y adaptables a las necesidades del ambiente, pero debido a la complejidad de los elementos que la ingeniería debe abarcar, es necesario que los procesos, de desarrollo de estos elementos lleven un orden y adquieran un estatus, un puntaje y un responsable, por este motivo es necesario que los participantes en el desarrollo de estos nuevos elementos definan metodologías a seguir y formas de realizar las tareas de innovación que describan las operaciones, los responsables de estas operaciones y los modelos conceptuales del resultado final y lograr así un mayor entendimiento de los procesos, las necesidades de los usuarios y del ambiente en general. La investigación de herramientas de modelado y metodologías de desarrollo, permita a los ingenieros lograr, que los productos finales y el proceso de creación de estos productos, se eficaz y eficiente, y llegar a un resultado de una calidad excepcional y que satisfaga las necesidades de los procesos, del ambiente y de los usuarios de forma exitosa [23].

### **4.1. LENGUAJE DE MODELADO UNIFICADO.**

(UML, por sus siglas en inglés, Unified Modeling Language). Los conceptos generales de los nuevos desarrollos, aplicaciones o elementos llevan consigo una complejidad, que a mayor complejidad lleve el nuevo elemento, mayor será la complejidad que tendrá la explicación y entendimiento de este para el ser humano, el modelado lleva esta complejidad a lenguaje comprensible y que permita entender de forma clara y concisa las características, generalidades y procesos de estos nuevos elementos. Así el UML es un lenguaje que permita visualizar, especificar, construir y modelar estos elementos para que las personas que trabajen en el desarrollo de estos, tengan un entendimiento y una visión general del desarrollo y una visión específica de cada una de las características y procesos de este.

UML ofrece un estándar para describir los conceptos de los nuevos desarrollos, incluyendo aspectos generales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación,

esquemas de bases de datos, componentes de software, requerimientos, entendimiento con el usuario y modelos de seguridad.

UML es un “lenguaje” para especificar las características y para el entendimiento de elementos con grandes complejidades, no es un método o un proceso para el desarrollo de estos elementos. UML se usa para definir un sistema de software; para detallar artefactos, relaciones de procesos; documentar y construir, es un lenguaje de tipo descriptivo. UML se puede usar en una gran variedad de formas para soportar una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar.

UML prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe como deben escribirse estas notaciones y diagramas (semántica, sintaxis, y significado).

**4.1.1. Diagramas de clases.** Una clase es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica. Los Diagramas de Clases son utilizados durante el proceso de Análisis y Diseño de los sistemas informáticos donde se crea el diseño conceptual de la información que se maneja en el sistema, los componentes que se encargaran del funcionamiento y la relación entre uno y otro. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. Durante el diseño, se usa el mismo diagrama, y se modifica para satisfacer los detalles de las implementaciones.

Un diagrama de clases está compuesto por los siguientes elementos:  
Clase: atributos, métodos y visibilidad.

Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

**4.1.2. Diagrama de paquetes.** Cualquier sistema grande se debe dividir en unidades más pequeñas, de modo que las personas puedan trabajar con una cantidad de información limitada, a la vez y de modo que los equipos de trabajo no interfieran con el trabajo de los otros. Un paquete es una parte de un modelo. Cada parte del modelo debe pertenecer a un paquete. Pero para ser funcional, la asignación debe seguir un cierto principio racional, tal como funcionalidad común,

implementación relacionada y punto de vista común. UML no impone una regla para componer los paquetes.

Los paquetes ofrecen un mecanismo general para la organización de los modelos/subsistemas agrupando elementos de modelado. Cada paquete corresponde a un sub modelo (subsistema) del modelo (sistema). Los paquetes son unidades de organización jerárquica de uso general de los modelos de UML. Pueden ser utilizados para el almacenamiento, el control de acceso, la gestión de la configuración y la construcción de bibliotecas que contengan fragmentos reutilizables del modelo. Un paquete puede contener otros paquetes, sin límite de anidamiento pero cada elemento pertenece a (está definido en) sólo un paquete. El Diagrama de Paquetes, muestra como un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema. Los Paquetes están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los paquetes. Los paquetes son buenos elementos de gestión. Cada paquete puede asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido.

Los paquetes se dibujan como rectángulos, las dependencias se muestran como flechas con líneas discontinuas. El operador "::" permite designar una clase definida en un contexto distinto del actual.

**4.1.3. Diagrama de casos de uso.** Este diagrama muestra el comportamiento del sistema visto desde la perspectiva del usuario, indica la funcionalidad a cumplir, es una especie de diagrama de comportamiento. UML define una notación gráfica para representar casos de uso llamada modelo de casos de uso, representados por elipses y los actores están representados por las figuras humanas. Una caja define los límites del sistema, por ejemplo, los casos de uso se muestran como parte del sistema que está siendo modelado, los actores no.

Existen 3 relaciones principales entre los casos de uso:

- **Include.** En una forma de interacción, un caso de uso dado puede "incluir" otro. El primer caso de uso a menudo depende del resultado del caso de uso incluido. Esto es útil para extraer comportamientos verdaderamente comunes desde múltiples casos de uso a una descripción individual. La

notación es una flecha rayada desde el caso de uso que lo incluye hasta el caso de uso incluido, con la etiqueta «include». Este uso se asemeja a una expansión de una macro donde el comportamiento del caso incluido es colocado dentro del comportamiento del caso de uso base. No hay parámetros o valores de retorno.

- **Extend.** En otra forma de interacción, un caso de uso dado, (la extensión) puede extender a otro. Esta relación indica que el comportamiento del caso de uso extensión puede ser insertado en el caso de uso extendido bajo ciertas condiciones. La notación es una flecha rayada desde el caso de uso extensión al caso de uso extendido, con la etiqueta «extend». Esto puede ser útil para lidiar con casos especiales, o para acomodar nuevos requisitos durante el mantenimiento del sistema y su extensión.
- **Generalization.** En la tercera forma de relación entre casos de uso, existe una relación generalización/especialización. Un caso de uso dado puede estar en una forma especializada de un caso de uso existente. La notación es una línea sólida terminada en un triángulo dibujado desde el caso de uso especializado al caso de uso general.

## 4.2. EL PROCESO RACIONAL UNIFICADO O RUP.

Los orígenes de RUP se remontan al modelo espiral, es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP es en realidad un refinamiento realizado por Rational Software del más genérico Proceso Unificado.

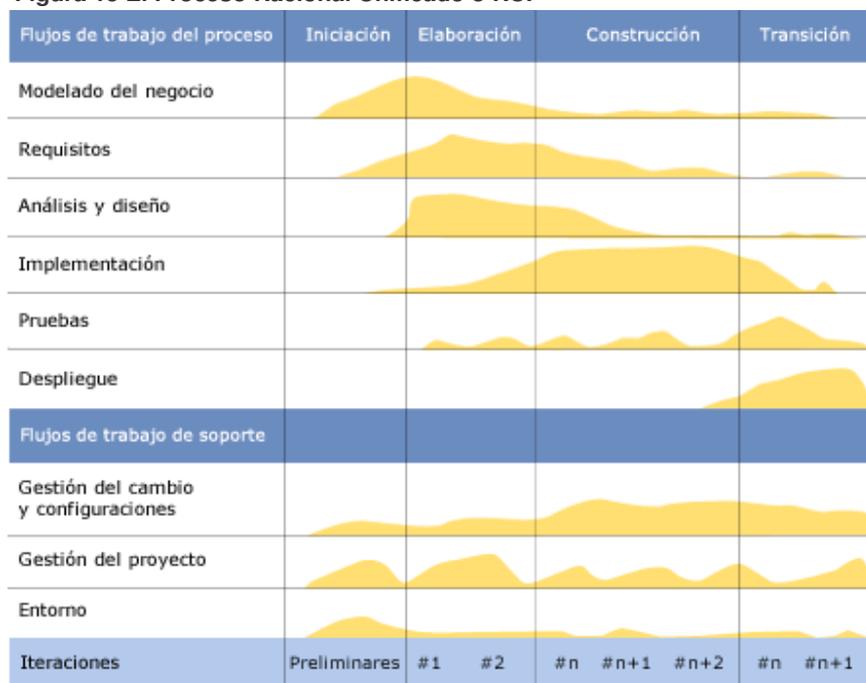
Sus principales características son:

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en Ingeniería de Software.
- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.
- Modelado visual del software.
- Verificación de la calidad del software.

Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso). El RUP divide el proceso de desarrollo en ciclos, teniendo un producto final al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante:

- **Inicio:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **Elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- **Construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- **Transición:** se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

Figura 15 El Proceso Racional Unificado o RUP



Wikipedia

Los principios que caracterizan al RUP son:

- **Adaptar el proceso:** El proceso deberá adaptarse a las características propias del proyecto u organización.
- **Balancear prioridades:** Los requerimientos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.
- **Demostrar valor iterativamente:** Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados
- **Elevar el nivel de abstracción:** Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks) por nombrar algunos. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML.
- **Enfocarse en la calidad:** El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

### 4.3. ANÁLISIS UML

Para el análisis y diseño de VSHOP-CMS se utilizó el RUP (Rational Unified Process) con la notación UML.

#### 4.3.1. Funciones

Tabla 2. Funciones del sistema

| Ref # | Función                            | Cat      | Atributo | Detalles y Restricciones | Cat.        |
|-------|------------------------------------|----------|----------|--------------------------|-------------|
| R1    | Ejecutar la Herramienta            | Evidente | Interfaz |                          | Obligatorio |
| R1.1  | Mostrar Ventana de logueo          | Evidente | Interfaz | Usuarios Autorizados     | Obligatorio |
| R1.2  | Mostrar Interfaz Gráfica Principal | Evidente | Interfaz |                          | Obligatorio |

|      |   |          |          |   |             |
|------|---|----------|----------|---|-------------|
| R2   | Mostrar mensajes de error                           | Evidente | Interfaz | Cuadros de Diálogo  | Deseable.   |
| R3   | Definir Conexión de Base de Datos                   | Oculto   | Interfaz | Activa solo cuando la conexión no exista<br>Usuarios Autorizados  | Obligatoria |
| R4.  | Definir el diseño de la tienda                      | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R4.1 | Definir el diseño de la página principal            | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R4.2 | Definir el diseño de la página de catálogos         | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R4.3 | Definir el diseño de la pagina de detalles          | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R5   | Crear Plantilla                                     | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R6   | Modificar Plantilla                                 | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R7   | Importar Plantilla                                  | Evidente | Interfaz | Usuarios Autorizados<br>Los archivos plantilla, imagen, y soporte deben tener la estructura de CSS, JPG o PNG y HTML respectivamente, y regirse con las guías de creación de los archivos de apariencia | Deseable    |
| R8   | Ingreso de Productos                                | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R9   | Registro Ventas                                     | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R10  | Creación de Tiendas                                 | Evidente | Interfaz | Usuarios Autorizados  | Obligatorio |
| R11  | Ingreso de Ofertas                                  | Evidente | Interfaz | Usuarios Autorizados<br>Deben existir productos   | Deseable    |
| R12  | Ingreso de Noticias                                 | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R13  | Ingreso Opciones Adicionales (Misión, Visión, etc.) | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R15  | Administración de Marcas                            | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R16  | Administración de Proveedores                       | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R17  | Administración Ubicación Geográfica                 | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R18  | Administración Catálogos                            | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R19  | Gestión Personal                                    | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R20  | Generación de Reportes                              | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R21  | Generación Estadísticas                             | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |
| R22  | Sincronización Móvil                                | Evidente | Interfaz | Usuarios Autorizados  | Deseable    |

**4.3.2. Casos de uso.** Se presentan los casos de uso que muestran el funcionamiento de VSHOP CMS tanto en la parte web como en la aplicación móvil.

**4.3.2.1. Aplicación web.** A continuación se muestra los casos de uso en los que se describe la parte web del sistema.

|  |   |
|--|---|
| ID   | CU01  |
| Caso de uso  | Diseño Tienda Pagina Principal                                |
| Actor principal  | Administrador   |
| Actores relacionados   |   |
| Precondiciones   |   |
| Pos condiciones  | Aplicación de la plantilla a la página principal de la tienda |
| Flujo Básico   |   |
| <p>El sistema muestra una lista de las plantillas existentes en el sistema para la apariencia la pagina principal, opciones de actualizar la lista de plantillas, importar plantilla, crear plantilla, vista previa</p> <ol style="list-style-type: none"> <li>I.             <ol style="list-style-type: none"> <li>1. El usuario elige una plantilla</li> <li>2. El sistema muestra la vista previa de la plantilla</li> <li>3. El usuario elige aplicar plantilla</li> <li>4. El sistema muestra un cuadro de confirmación de la aplicación de la plantilla</li> </ol> </li> <li>II. El usuario elige actualizar lista             <ol style="list-style-type: none"> <li>1. El sistema muestra la lista de plantillas con las nuevas plantillas ingresadas</li> </ol> </li> <li>III. El usuario elige importar plantilla             <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla importa plantilla</li> </ol> </li> <li>IV. El usuario elige crear plantilla             <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla crear plantilla</li> </ol> </li> <li>V. El usuario elige vista previa             <ol style="list-style-type: none"> <li>1. El sistema muestra la pagina principal con la plantilla seleccionada</li> </ol> </li> </ol> |   |
| Flujos Alternativos  |   |
| <ol style="list-style-type: none"> <li>1. No existe la vista previa de la plantilla             <ol style="list-style-type: none"> <li>a. El sistema muestra una imagen de vista previa no disponible</li> </ol> </li> </ol>   |   |
| Observaciones  |   |

|  |
|--|
|  |
|--|

|   |   |
|---|---|
| ID  | CU02  |
| Caso de uso   | Diseño de la Tienda Catálogos                 |
| Actor principal   | Administrador                                 |
| Actores relacionados  |   |
| Precondiciones  |   |
| Pos condiciones   | Aplicar plantilla a la página de de catálogos |
| Flujo Básico  |   |
| <p>El sistema muestra una lista de las plantillas existentes en el sistema para la apariencia la pagina de Catálogos, opciones de actualizar la lista de plantillas, importar plantilla. crear plantilla</p> <p>I.</p> <ol style="list-style-type: none"> <li>1. El usuario elige una plantilla</li> <li>2. El sistema muestra la vista previa de la plantilla</li> <li>3. El usuario elige aplicar plantilla</li> <li>4. El sistema muestra un cuadro de confirmación de la aplicación de la plantilla</li> </ol> <p>II. El usuario elige actualizar lista</p> <ol style="list-style-type: none"> <li>1. El sistema muestra la lista de plantillas con las nuevas plantillas ingresadas</li> </ol> <p>III. El usuario elige importar plantilla</p> <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla importa plantilla</li> </ol> <p>IV. El usuario elige crear plantilla</p> <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla crear plantilla</li> </ol> |   |
| Flujos Alternativos   |   |
| <ol style="list-style-type: none"> <li>1. No existe la vista previa de la plantilla</li> </ol> <p>El sistema muestra una imagen de vista previa no disponible</p>   |   |
| Observaciones   |   |
|   |   |

|                      |  |
|----------------------|--|
| ID                   | CU03   |
| Caso de uso          | Diseño de tienda Detalle Producto              |
| Actor principal      | Administrador                                  |
| Actores relacionados |  |
| Precondiciones       |  |
| Pos condiciones      | Seleccionar apariencia pagina detalle producto |

|   |
|---|
| Flujo Básico  |
| <p>El sistema muestra una lista de las plantillas existentes en el sistema para la apariencia la pagina detalle producto, opciones de actualizar la lista de plantillas, importar plantilla, crear plantilla, vista previa</p> <ol style="list-style-type: none"> <li>I. <ol style="list-style-type: none"> <li>1. El usuario elige una plantilla</li> <li>2. El sistema muestra la vista previa de la plantilla</li> <li>3. El usuario elige aplicar plantilla</li> <li>4. El sistema muestra un cuadro de confirmación de la aplicación de la plantilla</li> </ol> </li> <li>II. El usuario elige actualizar lista <ol style="list-style-type: none"> <li>1. El sistema muestra la lista de plantillas con las nuevas plantillas ingresadas</li> </ol> </li> <li>III. El usuario elige importar plantilla <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla importa plantilla</li> </ol> </li> <li>IV. El usuario elige crear plantilla <ol style="list-style-type: none"> <li>1. Se dispara el caso de uso plantilla crear plantilla</li> </ol> </li> <li>V. El usuario elige vista previa <p>El sistema muestra la pagina principal con la plantilla seleccionada</p> </li> </ol> |
| Flujos Alternativos   |
| <ol style="list-style-type: none"> <li>1. No existe la vista previa de la plantilla</li> </ol> <p>El sistema muestra una imagen de vista previa no disponible</p>   |
| Observaciones   |
|   |

|                      |   |
|----------------------|---|
| ID                   | CU04  |
| Caso de uso          | Diseño Tienda Efectos de Sección  |
| Actor principal      | Administrador   |
| Actores relacionados |   |
| Precondiciones       |   |
| Pos condiciones      | Seleccionar Efectos de de apariencia de la tienda   |
| Flujo Básico         | <p>El sistema muestra una lista de las secciones que están disponibles para la tienda.</p> <ol style="list-style-type: none"> <li>1. El usuario elige una sección</li> <li>2. El sistema muestra una lista de los efectos disponibles para la sección</li> <li>3. El usuario elige un efecto de sección</li> <li>4. El usuario elige guardar el cambio</li> <li>5. El sistema muestra un cuadro de confirmación de aplicación del efecto</li> </ol> |
| Flujos Alternativos  |   |
| Observaciones        |   |

|  |
|--|
|  |
|--|

|  |   |
|--|---|
| ID   | CU05  |
| Caso de uso  | Plantilla Crear plantilla                         |
| Actor principal  | Administrador                                     |
| Actores relacionados   |   |
| Precondiciones   |   |
| Pos condiciones  | Seleccionar Efectos de de apariencia de la tienda |
| Flujo Básico   |   |
| <p>El sistema muestra una lista de las secciones que están disponibles para la tienda en la página principal, una vista previa, una opción de guardar la plantilla</p> <ol style="list-style-type: none"> <li>1. El usuario elige una sección</li> <li>2. El sistema muestra las secciones contenidas con las características de apariencia como fondo fuente color alineación etc.</li> <li>3. El usuario elige las opciones de las características de sección</li> <li>4. El sistema pide el nombre de la plantilla</li> <li>5. El usuario digita el nombre de la plantilla</li> <li>6. El usuario elige guardar la plantilla</li> <li>7. El sistema muestra un cuadro donde solicita el archivo de distribución y la imagen de vista previa</li> <li>8. El sistema muestra un cuadro de confirmación donde verifica que se guardo la plantilla</li> </ol> |   |
| Flujos Alternativos  |   |
| <ol style="list-style-type: none"> <li>6. Si el usuario no ingresado el nombre de plantilla<br/>El sistema muestra un cuadro donde informa al usuario que el nombre de platilla es obligatorio</li> <li>7-8. Si el usuario no ingresa los archivos de vista previa y apoyo<br/>El sistema muestra un cuadro donde informa que los archivos de apoyo y vista previa no pueden ser guardados.</li> </ol>   |   |
| Observaciones  |   |
| <ul style="list-style-type: none"> <li>• El archivo de apoyo es una archivo HTML que permite modificar la ubicación de las secciones en la tienda</li> <li>• El archivo de vista previa es una imagen que utiliza el sistema para mostrarle al usuario la apariencia de la tienda con la platilla creada</li> </ul>  |   |

|                 |                             |
|-----------------|-----------------------------|
| ID              | CU06                        |
| Caso de uso     | Plantilla Editar Plantillas |
| Actor principal | Administrador               |

|   |                     |
|---|---------------------|
| Actores relacionados  |                     |
| Precondiciones  |                     |
| Pos condiciones   | Modificar Plantilla |
| Flujo Básico  |                     |
| <p>El sistema muestra una lista de las plantillas disponibles en el sistema</p> <ol style="list-style-type: none"> <li>1. El usuario elige una plantilla</li> <li>2. El sistema muestra una vista previa de la plantilla</li> <li>3. El usuario elige modificar plantilla</li> <li>4. El sistema muestra una lista de de las secciones de la plantilla</li> <li>5. El usuario elige una sección</li> <li>6. El sistema muestra las secciones internas las características, como son color fondo, fuente alineación etc.</li> <li>7. El usuario modifica los valores de las características</li> <li>8. El usuario elige guardar</li> <li>9. El sistema muestra un cuadro de confirmación de que se guardaron los cambios</li> </ol> |                     |
| Flujos Alternativos   |                     |
| <ol style="list-style-type: none"> <li>8 El usuario elige guardar como <ol style="list-style-type: none"> <li>a. El sistema solicita el nombre de la plantilla</li> <li>b. El usuario ingresa el nombre de la plantilla<br/>Si el usuario no ingresa el nombre de la plantilla, el sistema muestra un <u>cuadro</u> donde informa que el nombre de plantilla es obligatorio</li> <li>c. El usuario elige guardar</li> <li>d. El sistema muestra un cuadro de confirmación de que se guardaron los cambios</li> </ol> </li> </ol>  |                     |
| Observaciones   |                     |
| Este caso de uso de aplica para página principal, catálogos y pagina de detalle producto  |                     |

|   |                               |
|---|-------------------------------|
| ID  | CU07                          |
| Caso de uso   | Plantillas Importar Plantilla |
| Actor principal   | Administrador                 |
| Actores relacionados  |                               |
| Precondiciones  |                               |
| Pos condiciones   | Importar plantilla            |
| Flujo Básico  |                               |
| <ol style="list-style-type: none"> <li>1. El sistema solicita el nombre de plantilla</li> </ol> |                               |

|  |
|--|
| <ol style="list-style-type: none"> <li>2. El usuario ingresa el nombre de plantilla</li> <li>3. El sistema pide el archivo CSS</li> <li>4. El usuario ingresa el archivo CSS</li> <li>5. El sistema pide el archivó de distribución</li> <li>6. El usuario ingresa el archivo de distribución</li> <li>7. El sistema pide el archivo de vista previa</li> <li>8. El usuario ingresa el archivo de vista previa</li> <li>9. El usuario elige importa la plantilla</li> <li>10.El sistema muestra un cuadro de confirmación de la importación de la plantilla</li> </ol> |
| Flujos Alternativos  |
| <ol style="list-style-type: none"> <li>9 el usuario no ingresa el nombre de platilla <ol style="list-style-type: none"> <li>a. el sistema muestra un cuadro donde informa que el nombre de plantilla es obligatorio</li> </ol> </li> <li>9 el usuario no ingresa el archivo de CSS. <ol style="list-style-type: none"> <li>b. el sistema muestra un cuadro donde informa que el archivo CSS es obligatorio</li> </ol> </li> </ol>  |
| Observaciones  |
| <p>Los archivos de vista previa y distribución no son obligatorios</p> <p>Este caso de uso se aplica igualmente para catálogos, detalle producto y pagina principal</p>  |

|  |  |
|--|--|
| ID   | CU08   |
| Caso de uso  | Gestión Productos                                |
| Actor principal  | Administrador                                    |
| Actores relacionados   |  |
| Precondiciones   |  |
| Pos condiciones  | Ingreso, Modificación, Desactivación de producto |
| Flujo Básico   |  |
| <p>El sistema muestra una lista de los producto activos</p> <ol style="list-style-type: none"> <li>I. El usuario elige crear nuevo producto <ol style="list-style-type: none"> <li>1. El sistema pide el nombre, descripción, marca referencia</li> <li>2. El usuario ingresa los datos del producto</li> <li>3. El sistema pide las imágenes relacionadas al producto</li> <li>4. El usuario ingresa las imágenes del producto</li> <li>5. El sistema pide el precio inicial, palabras claves y impuesto</li> <li>6. El usuario ingresa los datos del producto</li> <li>7. El sistema pide los catálogos a los que pertenece el producto</li> <li>8. El usuario ingresa los catálogos</li> <li>9. El usuario elige guardar los cambios</li> </ol> </li> </ol> |  |

|   |
|---|
| <ul style="list-style-type: none"> <li>10. El sistema muestra un cuadro de confirmación de que se guardaron los cambios</li> <li>11. El sistema actualiza la lista de productos activos</li> </ul>  |
| <ul style="list-style-type: none"> <li>II. El usuario elige modificar el producto <ul style="list-style-type: none"> <li>1. El sistema muestra los datos del producto</li> <li>2. El usuario modifica los datos del producto</li> <li>3. El usuario elige guardar los cambios</li> <li>4. El sistema muestra un cuadro de confirmación de que se guardaron los cambios</li> <li>5. El sistema actualiza la lista de productos</li> </ul> </li> </ul>      |
| <ul style="list-style-type: none"> <li>III. El usuario elige desactivar el producto <ul style="list-style-type: none"> <li>1. El sistema actualiza la lista de producto activos</li> </ul> </li> </ul>  |
| Flujos Alternativos   |
| <ul style="list-style-type: none"> <li>I-7,II-2 el catalogo no existe <ul style="list-style-type: none"> <li>a) El usuario elige crear catalogo</li> <li>b) Se dispara el caso de uso gestión de catálogos</li> </ul> </li> <li>I-II el usuario no ingresa los datos de nombre referencia y marca del producto <ul style="list-style-type: none"> <li>a. El sistema muestra un cuadro donde informa que los datos son obligatorios</li> </ul> </li> </ul> |

|   |                                 |
|---|---------------------------------|
| ID  | CU09                            |
| Caso de uso   | Gestión Compra Producto         |
| Actor principal   | Administrador                   |
| Actores relacionados  |                                 |
| Precondiciones  |                                 |
| Pos condiciones   | Ingreso, Eliminación de Compras |
| Flujo Básico  |                                 |
| <p>El sistema muestra una lista de las compra activas</p> <ul style="list-style-type: none"> <li>I. El usuario elige ingresar compra <ul style="list-style-type: none"> <li>1. El sistema solicita proveedor, fecha de compra, costo adicional, observaciones</li> <li>2. El usuario ingresa los datos de la compra</li> <li>3. El sistema solicita el producto, cantidad, valor, descuento</li> <li>4. El usuario ingresa los datos del producto</li> <li>5. Se repite punto 3,4</li> <li>6. El usuario elige Registrar compra</li> <li>7. El sistema refresca la lista de compras activas</li> </ul> </li> <li>II. El usuario elige eliminar compra <ul style="list-style-type: none"> <li>1. El sistema refresca la lista de compras activas</li> </ul> </li> <li>III. El usuario elige mostrar los detalles de la compra</li> </ul> |                                 |

|   |
|---|
| 1. El sistema muestra los detalles de la compra: fecha, proveedor, costo, observaciones, productos, valor.  |
| Flujos Alternativos   |
| 6. El usuario no ingreso los datos de proveedor, fecha<br>a. El sistema informa que los campos son obligatorios<br>6. el usuario no ingreso productos a la compra<br>a. el sistema informa que el ingreso de productos es obligatorio |
| Observaciones   |
|   |

|  |                                       |
|--|---------------------------------------|
| ID   | CU10                                  |
| Caso de uso  | Gestión Registro Venta                |
| Actor principal  | Administrador                         |
| Actores relacionados   |                                       |
| Precondiciones   |                                       |
| Pos condiciones  | Ingresar, Modificar, Desactivar Venta |
| Flujo Básico   |                                       |
| <p>El sistema muestra una lista de las ventas activas con su fecha, cliente, estado, detalles y las opciones de ingresar, modificar y desactivar venta.</p> <p>I. El Usuario elige Ingresar Venta</p> <ol style="list-style-type: none"> <li>1. El sistema solicita vendedor, fecha.</li> <li>2. El usuario ingresa los datos de la venta</li> <li>3. El sistema solicita el producto, cantidad, descuento</li> <li>4. El usuario ingresa los datos de producto venta.</li> <li>5. El sistema muestra el subtotal, impuesto, valor de descuento, y el total</li> <li>6. El usuario agrega el producto a la venta</li> <li>7. El sistema muestra una lista de los productos agregados a la venta, y la opción de eliminar productos.</li> <li>8. Se repiten los pasos de 3-8</li> <li>9. El sistema pide el número de identificación del cliente.</li> <li>10. El usuario ingresa el número de identificación del cliente</li> <li>11. El sistema solicita el nombre, teléfono, dirección, e-mail, móvil, ciudad</li> <li>12. El usuario ingresa los datos del cliente</li> <li>13. El usuario elige registrar venta</li> <li>14. El sistema muestra total de la compra, total de impuesto, y el descuento</li> <li>15. El sistema solicita los costos adicionales</li> <li>16. El usuario ingresa el valor de los costos adicionales</li> <li>17. El sistema muestra el total de la venta</li> </ol> |                                       |

|   |
|---|
| <p>18. El usuario elige confirmar venta.</p> <p>19. El sistema actualiza la lista de ventas activas.</p> <p>II. El usuario elige modificar venta</p> <ol style="list-style-type: none"> <li>1. El sistema muestra los datos de venta, vendedor, fecha</li> <li>2. El usuario modifica los datos de la venta</li> <li>3. El sistema muestra una lista de los productos agregados a la venta, y la opción de desactivarlos</li> <li>4. El sistema solicita el producto, cantidad, descuento</li> <li>5. El usuario ingresa los datos de producto venta.</li> <li>6. El sistema muestra el subtotal, impuesto, valor de descuento, y el total</li> <li>7. El usuario agrega el producto a la venta</li> <li>8. El sistema actualiza la lista de los datos agregados a la venta.</li> <li>9. El sistema muestra los datos del cliente.</li> <li>10. El usuario modifica los datos del cliente.</li> <li>11. El usuario elige registrar venta</li> <li>20. El sistema muestra total de la compra, total de impuesto, y el descuento</li> <li>21. El sistema solicita el valor de los costos adicionales</li> <li>22. El usuario ingresa el valor de los costos adicionales</li> <li>23. El sistema muestra el total de la venta</li> <li>24. El usuario elige confirmar venta</li> <li>25. El sistema actualiza la lista de ventas activas</li> </ol> <p>III. El usuario elige desactivar venta.</p> <ol style="list-style-type: none"> <li>1. El sistema desactiva la venta</li> <li>2. El sistema actualiza la lista de ventas activas.</li> </ol> |
| Flujos Alternativos   |
| <p>I.1. - II.2. Si el vendedor no existe</p> <p>a. se dispara el caso de uso Gestión Vendedor</p> <p>I.13 Si no se ingreso la fecha de venta</p> <p>a. el sistema informa que el campo fecha es obligatorio</p> <p>I.13 no se ingreso productos a la venta</p> <p>a. el sistema informa que es obligatorio el ingreso de productos a la venta</p> <p>I.6 si no se ha ingresado un producto o cantidad.</p> <p>a. el sistema informa que es necesario un producto y cantidad.</p> <p>I.10 - II.10. si el cliente existe</p> <p>a. el sistema muestra los datos del cliente: Nombre, teléfono, dirección, ciudad, móvil , e-mail</p> <p>I.18.- II.24. El usuario elige cancelar venta</p> <p>a. el sistema descarta la venta ingresada, no se actualiza la lista de ventas activas</p>  |
| Observaciones   |
| <ul style="list-style-type: none"> <li>• Los datos de cliente no son obligatorios, si el usuario no los ingresa se registra la venta con cliente anónimo</li> </ul>   |

- El dato vendedor no es obligatorio.
- El valor de costos adicionales no es obligatorio, el valor por defecto es cero.
- Si el usuario elige cerrar la venta de modal referente a la confirmación de la venta, el sistema si registra la venta y actualiza la lista de ventas disponibles.

|  |  |
|--|--|
| ID   | CU11   |
| Caso de uso  | Gestión Registro Venta Oferta                |
| Actor principal  | Administrador                                |
| Actores relacionados   |  |
| Precondiciones   |  |
| Pos condiciones  | Ingresar, Modificar, Desactivar Venta Oferta |
| Flujo Básico   |  |
| El sistema muestra una lista de las ventas de ofertas activas con su fecha, cliente, estado, detalles y las opciones de ingresar, modificar y desactivar venta.  |  |
| <p>I. El Usuario elige Ingresar Venta Oferta</p> <ol style="list-style-type: none"> <li>1. El sistema solicita vendedor, fecha.</li> <li>2. El usuario ingresa los datos de la venta</li> <li>3. El sistema solicita la oferta</li> <li>4. El usuario selecciona la oferta</li> <li>5. El sistema pide el número de identificación del cliente.</li> <li>6. El usuario ingresa el número de identificación del cliente</li> <li>7. El sistema solicita el nombre, teléfono, dirección, e-mail, móvil, ciudad</li> <li>8. El usuario ingresa los datos del cliente</li> <li>9. El usuario elige registrar venta</li> <li>10. El sistema muestra total de la compra</li> <li>11. El sistema solicita los costos adicionales</li> <li>12. El usuario ingresa el valor de los costos adicionales</li> <li>13. El sistema muestra el total de la venta</li> <li>14. El usuario elige confirmar venta.</li> <li>15. El sistema actualiza la lista de ventas de ofertas activas.</li> </ol> <p>II. El usuario elige modificar venta</p> <ol style="list-style-type: none"> <li>1. El sistema muestra los datos de venta, vendedor, fecha</li> <li>2. El usuario modifica los datos de la venta</li> <li>3. El sistema muestra la oferta seleccionada</li> <li>4. El usuario modifica la oferta seleccionada</li> <li>5. El sistema muestra los datos del cliente.</li> <li>6. El usuario modifica los datos del cliente.</li> </ol> |  |

|   |
|---|
| <p>7. El usuario elige registrar venta</p> <p>8. El sistema muestra total de la compra, total de impuesto, y el descuento</p> <p>9. El sistema solicita el valor de los costos adicionales</p> <p>10. El usuario ingresa el valor de los costos adicionales</p> <p>11. El sistema muestra el total de la venta</p> <p>12. El usuario elige confirmar venta</p> <p>13. El sistema actualiza la lista de ventas activas</p> <p>III. El usuario elige desactivar venta.</p> <p>1. El sistema desactiva la venta</p> <p>2. El sistema actualiza la lista de ventas activas.</p> |
| Flujos Alternativos   |
| <p>I.1. - II.2. Si el vendedor no existe</p> <p>a. se dispara el caso de uso Gestión Vendedor</p> <p>I.13 Si no se ingreso la fecha de venta</p> <p>a. el sistema informa que el campo fecha es obligatorio</p> <p>I.10 - II.10. si el cliente existe</p> <p>a. el sistema muestra los datos del cliente: Nombre, teléfono, dirección, ciudad, móvil , e-mail</p> <p>I.18.- II.24. El usuario elige cancelar venta</p> <p>a. el sistema descarta la venta ingresada, no se actualiza la lista de ventas activas</p>   |
| Observaciones   |
| <ul style="list-style-type: none"> <li>• Los datos de cliente no son obligatorios, si el usuario no los ingresa se registra la venta con cliente anónimo</li> <li>• El dato vendedor no es obligatorio.</li> <li>• El valor de costos adicionales no es obligatorio, el valor por defecto es cero.</li> <li>• Si el usuario elige cerrar la venta de modal referente a la confirmación de la venta, el sistema si registra la venta y actualiza la lista de ventas disponibles.</li> </ul>  |

|  |  |
|--|--|
| ID   | CU12                                     |
| Caso de uso  | Gestión Costo Producto                   |
| Actor principal  | Administrador                            |
| Actores relacionados   |  |
| Precondiciones   |  |
| Pos condiciones  | Modificación valor del costo de producto |
| Flujo Básico   |  |
| <p>El sistema muestra una lista con los productos, con su referencia descripción y costo de producto, y la opción modificar el costo.</p> <p>1. El usuario elige modificar el costo producto</p> <p>2. El sistema muestra los datos del producto, referencia, descripción y el costo</p> |  |

|   |
|---|
| <p>actual</p> <ol style="list-style-type: none"> <li>3. El sistema solicita el nuevo costo del producto</li> <li>4. El usuario ingresa el costo del producto</li> <li>5. El usuario selecciona actualizar el producto</li> <li>6. El sistema actualiza la lista de los productos y su costo.</li> </ol> |
| Flujos Alternativos   |
| <ol style="list-style-type: none"> <li>5. El usuario no ingreso el costo del producto</li> </ol> <p>El sistema informa que el campo costo producto es obligatorio.</p>  |
| Observaciones   |
|   |

|   |   |
|---|---|
| ID  | CU13                                    |
| Caso de uso   | Gestión de Ofertas                      |
| Actor principal   | Administrador                           |
| Actores relacionados  |   |
| Precondiciones  |   |
| Pos condiciones   | Ingresar, Modificar, Desactivar Ofertas |
| Flujo Básico  |   |
| <p>El sistema muestra una lista de las ofertas activas, y las opciones de crear, modificar y desactivar oferta.</p> <ol style="list-style-type: none"> <li>I. El usuario elige ingresar oferta <ol style="list-style-type: none"> <li>1. El sistema solicita descripción, fecha, precio, cantidad disponible</li> <li>2. El usuario ingresa los datos de la oferta</li> <li>3. El sistema solicita nombre, descripción y ubicación del archivo de la imagen relacionada a la oferta</li> <li>4. El usuario ingresa los datos de la imagen oferta</li> <li>5. El sistema muestra la imagen relacionada con la oferta</li> <li>6. El solicita el producto perteneciente a la oferta y la cantidad de productos por oferta</li> <li>7. El usuario ingresa los datos de producto oferta</li> <li>8. El usuario selección agregar</li> <li>9. El sistema muestra una lista de los productos pertenecientes a la oferta</li> <li>10. Se repiten los punto 6-9</li> <li>11. El usuario elige guardar oferta</li> <li>12. El sistema actualiza la lista de ofertas activas</li> </ol> </li> <li>II. El usuario elige modificar oferta <ol style="list-style-type: none"> <li>1. El sistema muestra los datos de oferta, descripción, fecha precio cantidad disponible</li> <li>2. El usuario modifica los datos de la oferta</li> </ol> </li> </ol> |   |

|   |
|---|
| <ol style="list-style-type: none"> <li>3. El sistema muestra la imagen relacionada con la oferta</li> <li>4. El usuario modifica los datos de la imagen relacionada con la oferta</li> <li>5. El sistema muestra una lista de los producto pertenecientes a la oferta con su cantidad correspondiente, y la opción de desactivar</li> <li>6. El solicita el producto perteneciente a la oferta y la cantidad de productos por oferta</li> <li>7. El usuario ingresa los datos de producto oferta</li> <li>8. El sistema actualiza la lista de productos pertenecientes a la oferta</li> <li>9. El usuario elige guardar oferta</li> <li>10. El sistema actualiza la lista de oferta activas</li> </ol> <p>III. El usuario elige desactivar oferta<br/>El sistema desactiva la oferta y actualiza la lista de ofertas activas.</p> |
| Flujos Alternativos   |
| <p>I.11-II.9</p> <ol style="list-style-type: none"> <li>a. el usuario no ingres los datos de oferta, descripción, fecha, cantidad, precio<br/>El sistema informa que los datos de oferta son obligatorios</li> <li>b. el usuario no ingreso productos a la oferta<br/>El sistema informa que es obligatorio ingresar productos a la oferta</li> <li>c. el usuario no ingres la imagen relacionada con la oferta<br/>el sistema informa que la imagen es obligatorio</li> </ol>  |
| Observaciones   |

|  |  |
|--|--|
| ID   | CU14                                   |
| Caso de uso  | Gestión de Noticias                    |
| Actor principal  | Administrador                          |
| Actores relacionados   |  |
| Precondiciones   |  |
| Pos condiciones  | Ingresa, Modificar, Desactivar Noticia |
| Flujo Básico   |  |
| <p>El sistema muestra una lista de las noticias activas con su titulo, descripción corta, fecha, hora y estado y las opciones de ingresar modificar y desactivar tienda</p> <ol style="list-style-type: none"> <li>I. El usuario elige ingresar noticia <ol style="list-style-type: none"> <li>1. El sistema solicita el titulo, tipo, fecha y hora de la noticia</li> <li>2. El usuario ingresa los datos de la noticia</li> <li>3. El sistema solicita la descripción corta y el texto de la noticia</li> <li>4. El usuario ingresa los datos de noticia</li> <li>5. El sistema solicita el nombre, descripción y ubicación del archivo de la imagen relacionada con la noticia</li> <li>6. El usuario ingresa los datos de la imagen</li> </ol> </li> </ol> |  |

|  |
|--|
| <ul style="list-style-type: none"> <li>7. El usuario selecciona adicionar</li> <li>8. El sistema muestra la imagen relacionada con la noticia y la opción de eliminar</li> <li>9. Se repiten los punto 5-9</li> <li>10.El usuario elige guardar noticia</li> <li>11.El sistema actualiza la lista de noticias activas</li> </ul> <p>II. El usuario elige modificar noticias</p> <ul style="list-style-type: none"> <li>1. El sistema muestra los datos de la noticia : titulo, tipo, fecha y hora de la noticia</li> <li>2. El usuario modifica los datos de la noticia</li> <li>3. El sistema muestra la descripción corta y el texto de la noticia</li> <li>4. El usuario modifica los datos de la noticia</li> <li>5. El sistema muestra una las imágenes relacionada con la noticia y la opción de eliminar la imagen</li> <li>6. El sistema solicita el nombre, descripción y ubicación del archivo de la imagen relacionada con la noticia</li> <li>7. El usuario ingresa los datos de la imagen</li> <li>8. El usuario elige agregar</li> <li>9. El sistema actualiza las imágenes relacionadas con la noticia y muestra la imagen agregada</li> <li>10.El usuario elige guardar noticia</li> <li>11.El sistema actualiza la lista de noticias activas</li> </ul> <p>III. El usuario elige desactivar noticia</p> <p>El sistema desactiva la noticia y actualiza la lista de noticias activas</p> |
| Flujos Alternativos  |
| <p>I.1-II.2</p> <ul style="list-style-type: none"> <li>a. el usuario no ingreso los datos de la noticia: titulo, tipo, fecha y hora de la noticia el sistema informa que los datos son obligatorios</li> <li>b. el usuario no ingreso la descripción obligatorios</li> </ul>   |
| Observaciones: Cada imagen ingresa en la noticia tiene la opción de ser eliminada si el usuario la selección se actualiza las imágenes relacionadas con la noticia   |

|                      |  |
|----------------------|--|
| ID                   | CU15   |
| Caso de uso          | Gestión Opciones Adicionales                         |
| Actor principal      | Administrador  |
| Actores relacionados |  |
| Precondiciones       |  |
| Pos condiciones      | Ingresar, Modificar, Desactivar opciones adicionales |

|   |
|---|
| Flujo Básico  |
| <p>El sistema muestra una lista de opciones adicionales, y las opciones de ingresar, modificar y desactivar la opción adicional</p> <p>I. El usuario elige Ingresar opción adicional</p> <ol style="list-style-type: none"> <li>1. El sistema solicita el título y el texto de la opción adicional</li> <li>2. El usuario ingresa los datos de la opción adicional</li> <li>3. El sistema solicita el nombre, descripción y la ubicación del archivo de la imagen relacionada con la opción adicional</li> <li>4. El usuario ingresa los datos de la imagen</li> <li>5. El usuario elige agregar</li> <li>6. El sistema muestra la imagen agregada y la opción de eliminar la imagen</li> <li>7. Se repite los puntos 3-6</li> <li>8. El usuario elige guardar opción</li> <li>9. El sistema actualiza la lista de opciones activas</li> </ol> <p>II. El usuario elige modificar opción</p> <ol style="list-style-type: none"> <li>1. El sistema muestra los datos de la opción: título , texto</li> <li>2. El usuario modifica los datos de la opción</li> <li>3. El sistema muestra las imágenes relacionadas con la opción adicional, y la opción de eliminar la imagen</li> <li>4. El sistema solicita el nombre, descripción y la ubicación del archivo de la imagen relacionado con la opción adicional</li> <li>5. El usuario ingresa los datos de la imagen</li> <li>6. El usuario elige agregar</li> <li>7. El sistema muestra la imagen agregada y la opción de eliminar la imagen</li> <li>8. Se repite los punto 3-7</li> <li>9. El usuario elige guardar opción</li> <li>10. El sistema refresca la lista de opciones adicionales</li> </ol> <p>III. El usuario elige desactivar opción</p> <p>El sistema desactiva la opción y actualiza la lista de opciones adicionales</p> |
| Flujos Alternativos   |
|   |
| Observaciones   |
|   |

|                      |                          |
|----------------------|--------------------------|
| ID                   | CU16                     |
| Caso de uso          | Administración de Marcas |
| Actor principal      | Administrador            |
| Actores relacionados |                          |

|  |  |
|--|--|
| Precondiciones   |  |
| Pos condiciones  | Ingreso, Modificación, Eliminación de marcas |
| Flujo Básico   |  |
| <p>El sistema muestra una lista de las marcas existentes con el nombre de la marca y la opción eliminar por cada marcar, muestra un campo vacío y la opción guardar.</p> <p>I. Opción ingresar</p> <ol style="list-style-type: none"> <li>1. El sistema pide el nombre de la marca</li> <li>2. El usuario ingresa el nombre de la marca.</li> <li>3. El usuario selecciona el botón guardar</li> <li>4. El sistema actualiza la lista de marcas.</li> </ol> <p>II. Opción Modificar</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona una marcar.</li> <li>2. El sistema habilita el campo para ingreso del dato con el valor actual de la marca</li> <li>3. El usuario modifica la marca</li> <li>4. El sistema actualiza la lista de marcas.</li> </ol> <p>III. Opción Eliminar</p> <ol style="list-style-type: none"> <li>1. El usuario selección el botón eliminar</li> <li>2. El sistema elimina la marca, actualiza la lista de marcas.</li> </ol> |  |
| Flujos Alternativos  |  |
|  |  |
| Observaciones  |  |
| No existe la validación del campo vacío para ingresar la marca.  |  |

|  |  |
|--|--|
| ID   | CU17   |
| Caso de uso  | Administración de proveedores                  |
| Actor principal  | Administrador                                  |
| Actores relacionados   |  |
| Precondiciones   |  |
| Pos condiciones  | Ingreso, Modificación, Eliminación Proveedores |
| Flujo Básico   |  |
| <p>El sistema muestra una lista de los proveedores existentes con el nombre, asesor, dirección, teléfono del proveedor y la opción eliminar por cada proveedor, muestra los campos vacíos para el ingreso de un nuevo proveedor y la opción guardar.</p> <p>I. Opción ingresar</p> <ol style="list-style-type: none"> <li>1. El sistema pide el nombre, asesor, dirección, teléfono del proveedor</li> </ol> |  |

|   |
|---|
| <ol style="list-style-type: none"> <li>2. El usuario ingresa los datos del proveedor</li> <li>3. El usuario selecciona el botón guardar</li> <li>4. El sistema actualiza la lista de proveedores.</li> </ol> <p>II. Opción Modificar</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona un campo cualquiera del proveedor: nombre, asesor, dirección, teléfono.</li> <li>2. El sistema habilita el campo para ingreso del dato con el valor actual.</li> <li>3. El usuario modifica el valor del campo.</li> <li>4. El sistema actualiza la lista de proveedores.</li> </ol> <p>III. Opción Eliminar</p> <ol style="list-style-type: none"> <li>1. El usuario selección el botón eliminar</li> <li>2. El sistema elimina el proveedor, actualiza la lista de proveedores.</li> </ol> |
| Flujos Alternativos   |
| Observaciones   |
| No existe la validación del campo vacío para ingresar los datos del proveedor   |

|                      |  |
|----------------------|--|
| ID                   | CU18   |
| Caso de uso          | Administración de Ubicación Geográfica   |
| Actor principal      | Administrador  |
| Actores relacionados |  |
| Precondiciones       |  |
| Pos condiciones      | Ingreso, Modificación, Eliminación, de países; Ingreso, Modificación, Eliminación ciudades.  |
| Flujo Básico         | <p>El sistema muestra una lista de los países existentes, con el nombre del país y las opciones de eliminar y desplegar por cada país. Un campo vacío y la opción de guardar para los nuevos países.</p> <p>I. Opción Ingresar</p> <ol style="list-style-type: none"> <li>1. El sistema solicita el nombre del país.</li> <li>2. El usuario ingres el nombre del país</li> <li>3. El usuario elige la opción guardar</li> <li>4. El sistema actualiza la lista de países existentes.</li> </ol> <p>II. Opción Modificar</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona un país.</li> <li>2. El sistema habilita el campo con el valor actual</li> <li>3. El usuario modifica el valor del campo</li> <li>4. El sistema actualiza la lista de países.</li> </ol> <p>III. Opción Eliminar</p> |

|   |
|---|
| <ol style="list-style-type: none"> <li>1. El usuario elige la opción eliminar</li> <li>2. El sistema elimina el país, actualiza la lista de países.</li> </ol> <p>IV. Opción desplegar</p> <ol style="list-style-type: none"> <li>1. El usuario da seleccione desplegar:<br/>El sistema muestra una lista de las ciudades pertenecientes al país, y las opción de eliminar por cada ciudad, un campo vacío y la opción de guardar para las nuevas ciudades. <ol style="list-style-type: none"> <li>I. Opción Ingresar <ol style="list-style-type: none"> <li>1. El sistema solicita el nombre de la ciudad</li> <li>2. El usuario ingresa el nombre de la ciudad</li> <li>3. El usuario selecciona la opción guardar</li> <li>4. El sistema actualiza la lista de ciudades</li> </ol> </li> <li>II. Opción Modificar <ol style="list-style-type: none"> <li>1. El usuario seleccione una ciudad</li> <li>2. El sistema habilita un campo con el dato actual</li> <li>3. El usuario modifica el valor del campo</li> <li>4. El sistema actualiza la lista de ciudades</li> </ol> </li> <li>III. Opción Eliminar <ol style="list-style-type: none"> <li>1. El usuario selecciona la opción eliminar</li> <li>2. El sistema elimina la ciudad y actualiza la lista de ciudades.</li> </ol> </li> </ol> </li> </ol> |
| Flujos Alternativos   |
| Observaciones: Este caso de uso es un uso simplificado del componente xMultiGridAdmin   |

|  |   |
|--|---|
| ID   | CU19  |
| Caso de uso  | Administración de Catálogos                     |
| Actor principal  | Administrador                                   |
| Actores relacionados   |   |
| Precondiciones   |   |
| Pos condiciones  | Ingreso, Modificación, Eliminación de Catálogos |
| Flujo Básico   |   |
| El sistema muestra una lista de los catálogos existentes, con su nombre y descripción; las opciones de eliminación y despliegue por cada catalogo, campos para ingresar los datos de nombre y descripción de los nuevos catálogos y la opción de guardar los nuevos catálogos. |   |

|   |
|---|
| <p>I. Opción Ingresar.</p> <ol style="list-style-type: none"> <li>1. El sistema solicita el nombre y la descripción del catálogo.</li> <li>2. El usuario ingresa los datos del catalogo.</li> <li>3. El usuario selecciona la opción guardar.</li> <li>4. El sistema actualiza la lista de catálogos.</li> </ol> <p>II. Opción Modificar.</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona un campo de un catalogo sea nombre o descripción.</li> <li>2. El sistema habilita el campo para ingreso con el valor actual del campo.</li> <li>3. El usuario modifica el valor del campo.</li> <li>4. El sistema actualiza la lista de catálogos.</li> </ol> <p>III. Opción Eliminar.</p> <ol style="list-style-type: none"> <li>1. El usuario elige la opción eliminar catalogo.</li> <li>2. El sistema elimina el catalogo y actualiza la lista de catálogos.</li> </ol> <p>IV. Opción Desplegar</p> <ol style="list-style-type: none"> <li>1. El usuario elige la opción desplegar catalogo.</li> <li>2. Se reinicia el caso de Uso Administración de catálogos.</li> </ol> |
| Flujo Alternativo   |
| Observaciones: Esta es una aplicación completa del componente xMasterDetail   |

#### 4.3.2.2. Aplicación móvil.

A continuación se muestran los casos de uso correspondientes a la aplicación móvil del sistema.

|   |                                |
|---|--------------------------------|
| ID  | CU1                            |
| Caso de uso   | Selección de Opción            |
| Actor principal   | Preventista                    |
| Actores relacionados  |                                |
| Precondiciones  |                                |
| Pos condiciones   | Ingreso a opciones del sistema |
| Flujo Básico  |                                |
| <ol style="list-style-type: none"> <li>1. El sistema muestra las opciones de: <ul style="list-style-type: none"> <li>• Nuevo Cliente</li> <li>• Pre venta</li> <li>• Sincronización</li> </ul> </li> <li>2. El usuario elige la opción</li> </ol> |                                |
| Flujo Alternativo   |                                |
| 2. El usuario elige la opción Nuevo Cliente   |                                |

|  |
|--|
| <p>Se lanza el caso de uso Ingreso Clientes</p> <p>2. El usuario elige la opción Pre venta</p> <p>    Se lanza el caso de uso Mostrar Clientes</p> <p>2. El usuario elige la opción Sincronizar</p> <p>    Se lanza el caso de uso Realizar Sincronización</p> |
| Observaciones:   |

|   |                           |
|---|---------------------------|
| ID  | CU2                       |
| Caso de uso   | Ingreso de Clientes       |
| Actor principal   | Preventista               |
| Actores relacionados  |                           |
| Precondiciones  |                           |
| Pos condiciones   | Creación de nuevo Cliente |
| Flujo Básico  |                           |
| <p>El sistema solicita el nombre, la identificación, el numero del móvil, el teléfono y la dirección del cliente</p> <p>El usuario ingresa los datos del cliente</p> <p>El usuario elige guardar los datos del cliente</p> <p>El sistema muestra la confirmación del ingreso de los datos</p> <p>Se lanza el caso de uso Mostrar Opciones</p> |                           |
| Flujo Alternativo   |                           |
| Observaciones: En el caso de que los datos estén vacíos el sistema no deja continuar el ingreso   |                           |

|  |                                |
|--|--------------------------------|
| ID   | CU3                            |
| Caso de uso  | Mostrar Clientes               |
| Actor principal  | Preventista                    |
| Actores relacionados   |                                |
| Precondiciones   | Selección ingreso de pre venta |
| Pos condiciones  | Selección de un cliente        |
| Flujo Básico   |                                |
| <p>El sistema muestra una lista de los clientes existentes</p> <p>El usuario elige un cliente</p> <p>Se dispara el caso de uso mostrar productos</p> |                                |
| Flujo Alternativo  |                                |
| Observaciones: En el caso de que no existan clientes se muestra una lista con datos vacíos   |                                |

|  |                          |
|--|--------------------------|
| ID   | CU4                      |
| Caso de uso  | Mostrar Productos        |
| Actor principal  | Preventista              |
| Actores relacionados   |                          |
| Precondiciones   | Seleccionar un cliente   |
| Pos condiciones  | Selección de un producto |
| Flujo Básico   |                          |
| <p>El sistema muestra el nombre y la identificación del cliente seleccionado</p> <p>El sistema muestra una lista de los productos existentes</p> <p>El usuario elige un producto</p> <p>El sistema lanza el caso de mostrar subtotal pre venta</p> |                          |
| Flujo Alternativo  |                          |
| Observaciones: Si no existen datos de productos en el sistema informa que debe realizarse una sincronización del sistema   |                          |

|   |                            |
|---|----------------------------|
| ID  | CU5                        |
| Caso de uso   | Mostrar Subtotal Pre venta |
| Actor principal   | Preventista                |
| Actores relacionados  |                            |
| Precondiciones  | Seleccionar un producto    |
| Pos condiciones   | Mostrar subtotal pre venta |
| Flujo Básico  |                            |
| <p>El sistema muestra el nombre y la identificación del cliente</p> <p>El sistema muestra una lista de los productos agregados a la pre venta con su nombre precio actual y impuesto</p> <p>El sistema muestra el subtotal de la pre venta, el total de impuesto y el total de la pre venta</p> <p>El sistema muestra las opciones de finalizar pre venta, agregar producto, descartar producto</p> <p>El sistema muestra la confirmación de del ingreso de pre venta, los datos del cliente y el total de la pre venta</p> <p>Se lanza el caso de uso seleccionar opción</p> |                            |
| Flujo Alternativo   |                            |
| <p>4. El usuario elige agregar producto</p> <p>Se lanza el caso de uso listar productos</p> <p>4. El usuario elige descartar producto</p> <p>Se reinicia el caso de uso Mostrar Subtotal Pre venta</p>  |                            |
| Observaciones:  |                            |

|   |   |
|---|---|
| ID  | CU6   |
| Caso de uso   | Realizar Sincronización                               |
| Actor principal   | Preventista   |
| Actores relacionados  |   |
| Precondiciones  | Selección ingreso de Sincronización                   |
| Pos condiciones   | Sincronizar los datos de el móvil y la aplicación web |
| Flujo Básico  |   |
| <p>El sistema muestra el proceso de sincronización</p> <p>El sistema muestra la confirmación de la sincronización</p> <p>Se lanza el caso de uso Mostrar Opciones.</p>  |   |
| Flujo Alternativo   |   |
| <p>1. Se produjo un error en la sincronización</p> <p>a. El sistema informa del error de sincronización</p> <p>b. el sistema pregunta si se intenta hacer la sincronización nuevamente</p> <p>c. se reinicia el caso de uso Realizar Sincronización</p> |   |
| Observaciones: En caso de que no pudo realizar la sincronización, le pide al usuario volver a instalar la aplicación  |   |

**4.3.3. Diagramas de casos de uso.** A continuación de la figura 16 a la figura 19 se presentan los diagramas que integran los casos de uso que se especificaron en el anterior punto.

Figura 16 Diagrama de casos de uso 1

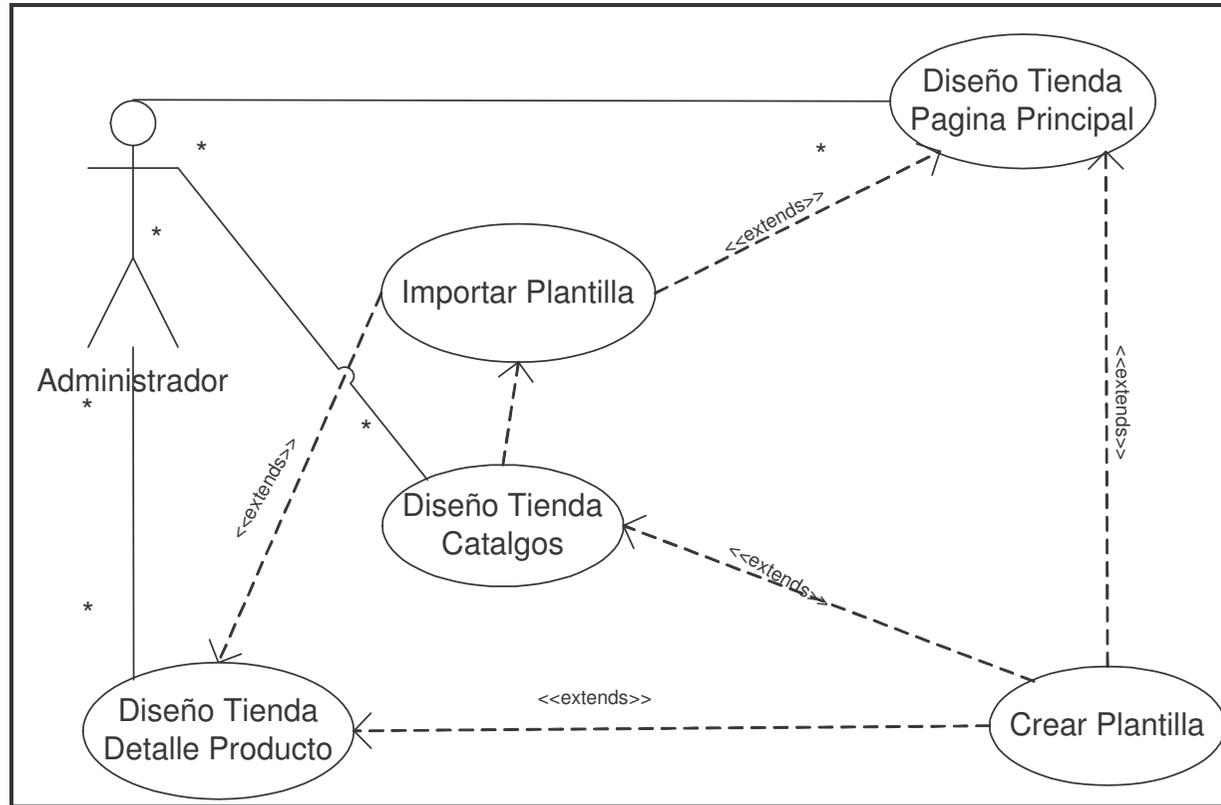


Figura 17 Diagrama de casos de uso 2

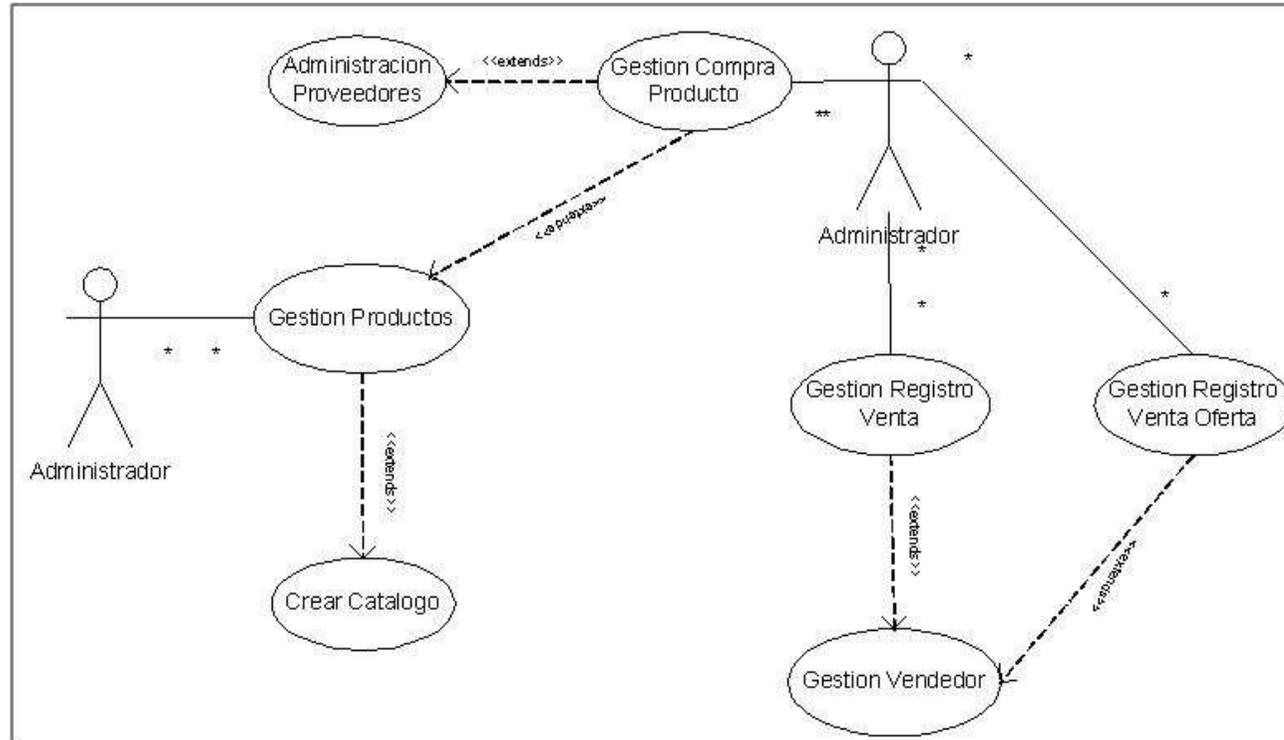
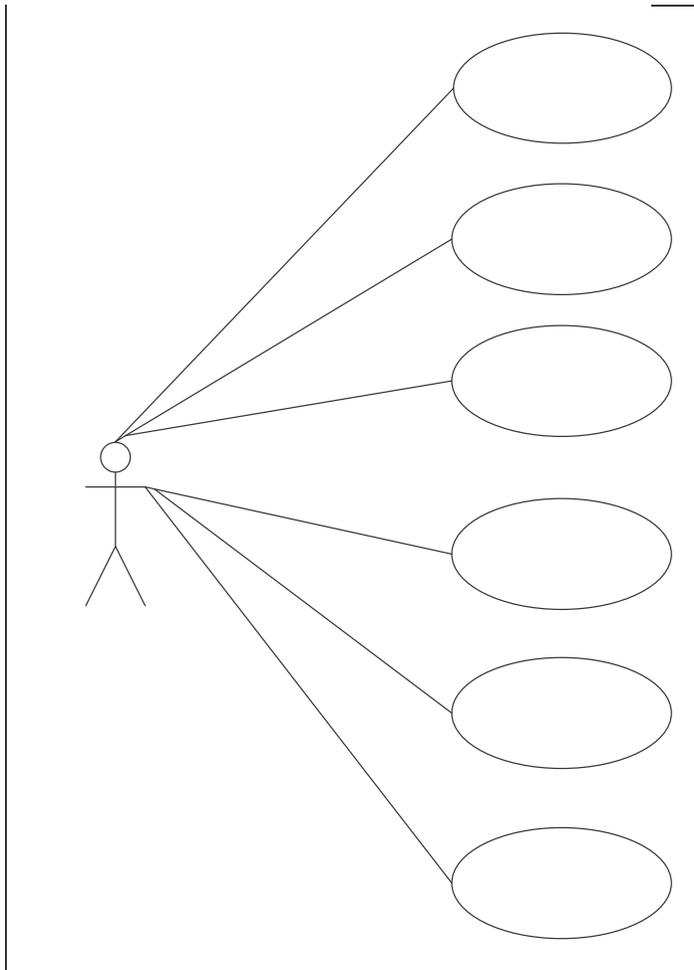


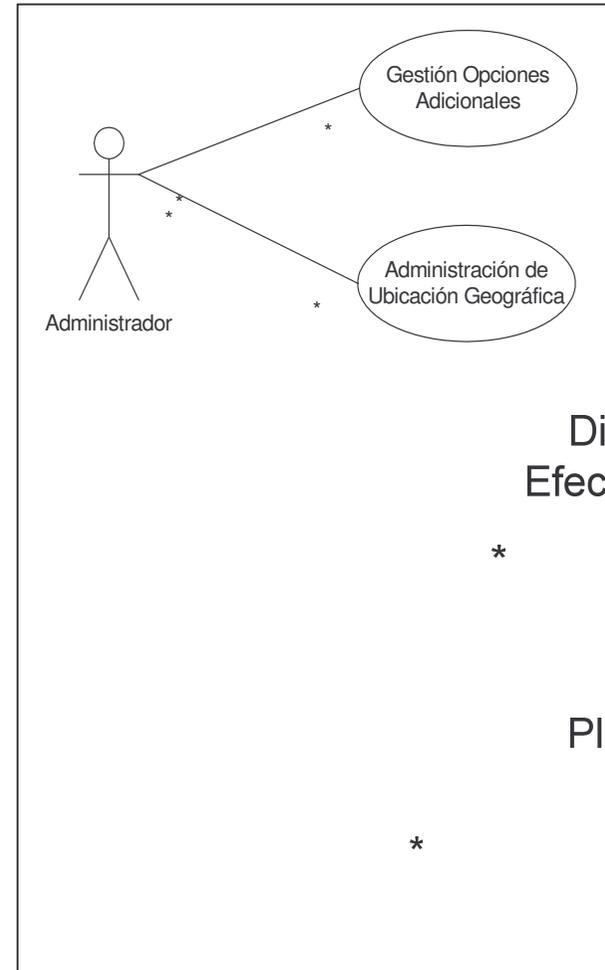
Figura 18 Diagrama de casos de uso 3



\*

\*\*

Figura 19 Diagrama de casos de uso 4



Diseño Tienda  
Efectos de Seccion

\*

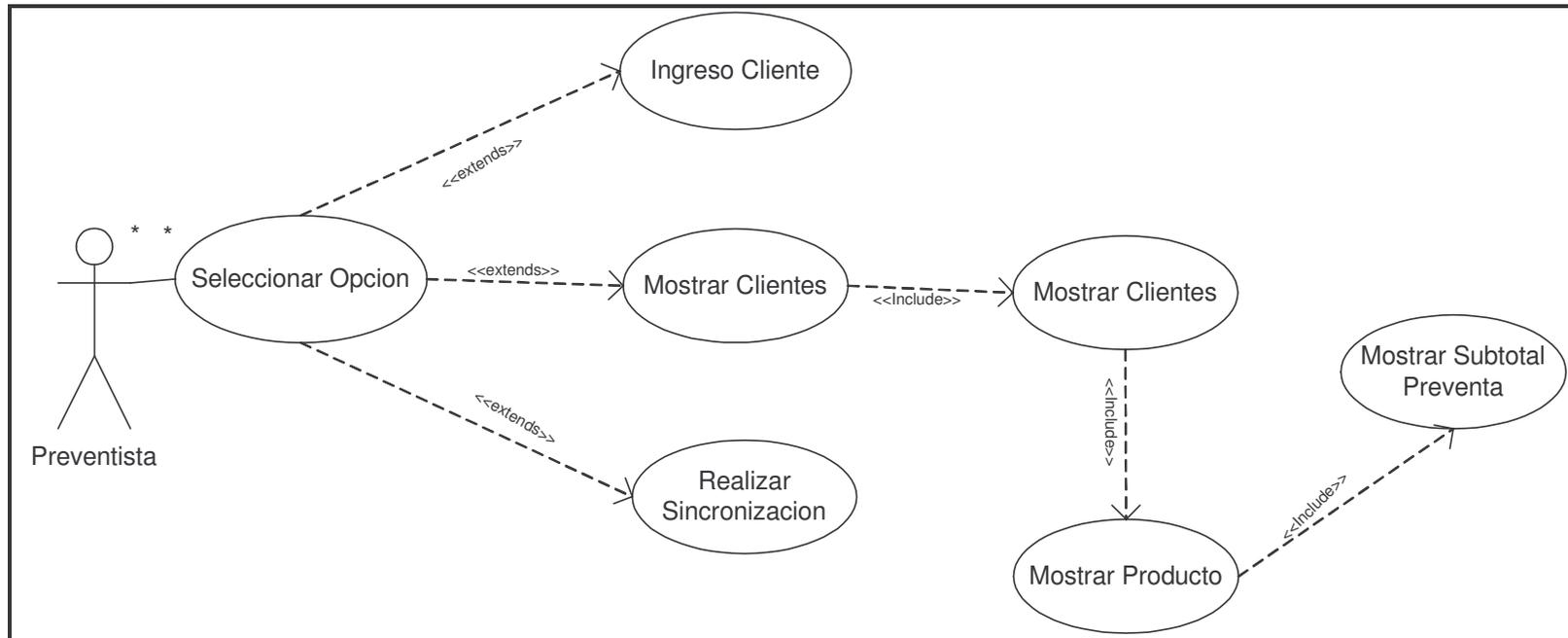
Plantilla Editar  
Plantillas

\*

Plantilla Editar  
Plantillas

\*

Figura 20 Diagrama de casos de uso 5 Aplicación Móvil





**4.3.4. Diagramas de clases.** Se presentan los diagramas de clases del sistema clasificados de la siguiente manera:

- Lógica Comercial
- Manejo de apariencia de la tienda
- Administración de contenidos
- xWebComponents
- Aplicación Móvil



Figura 21 Diagrama de Clases - Lógica Comercial

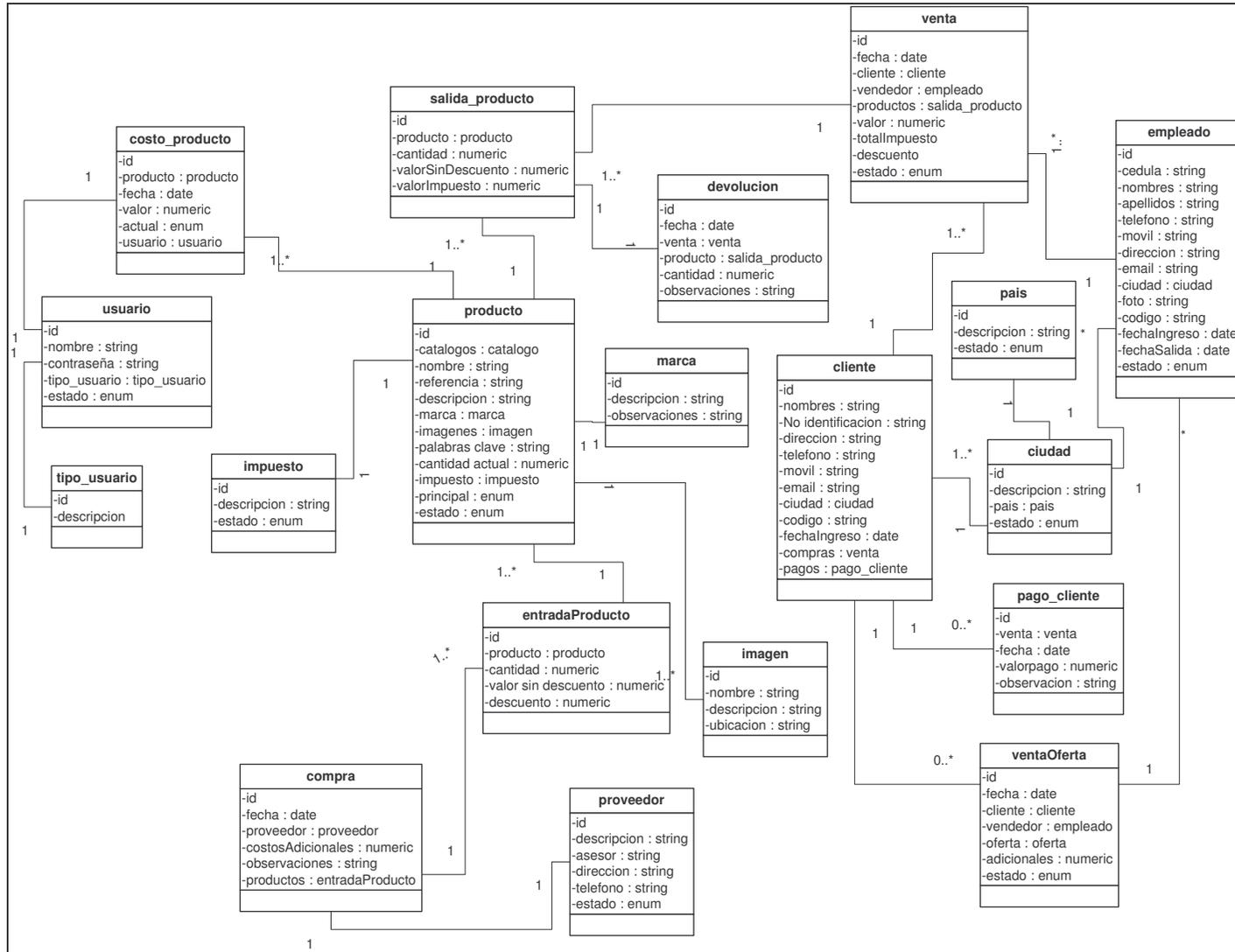


Figura 22 Diagrama de clases detallado - Lógica Comercial

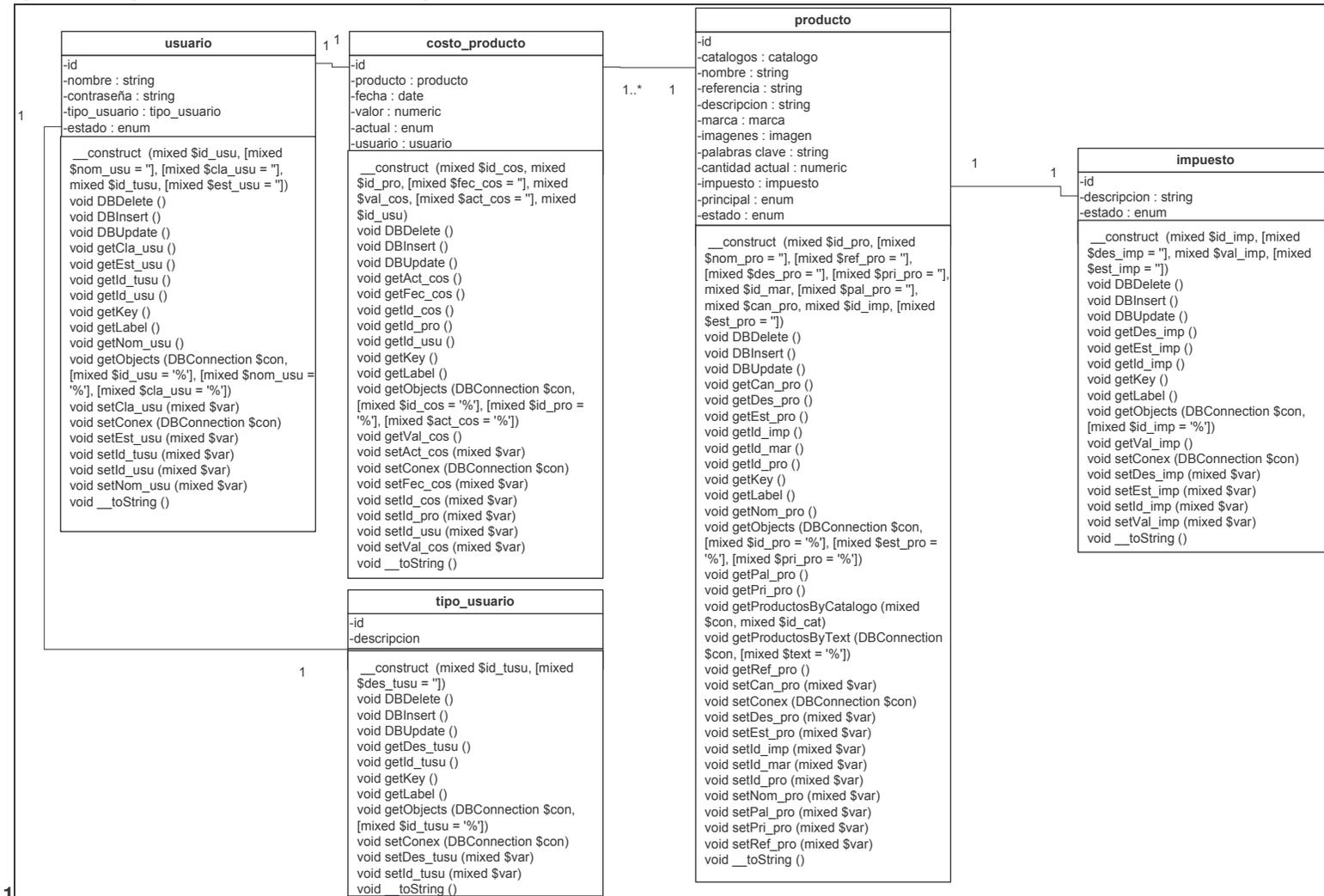


Figura 23 Diagrama de clases detallado - Lógica Comercial 2

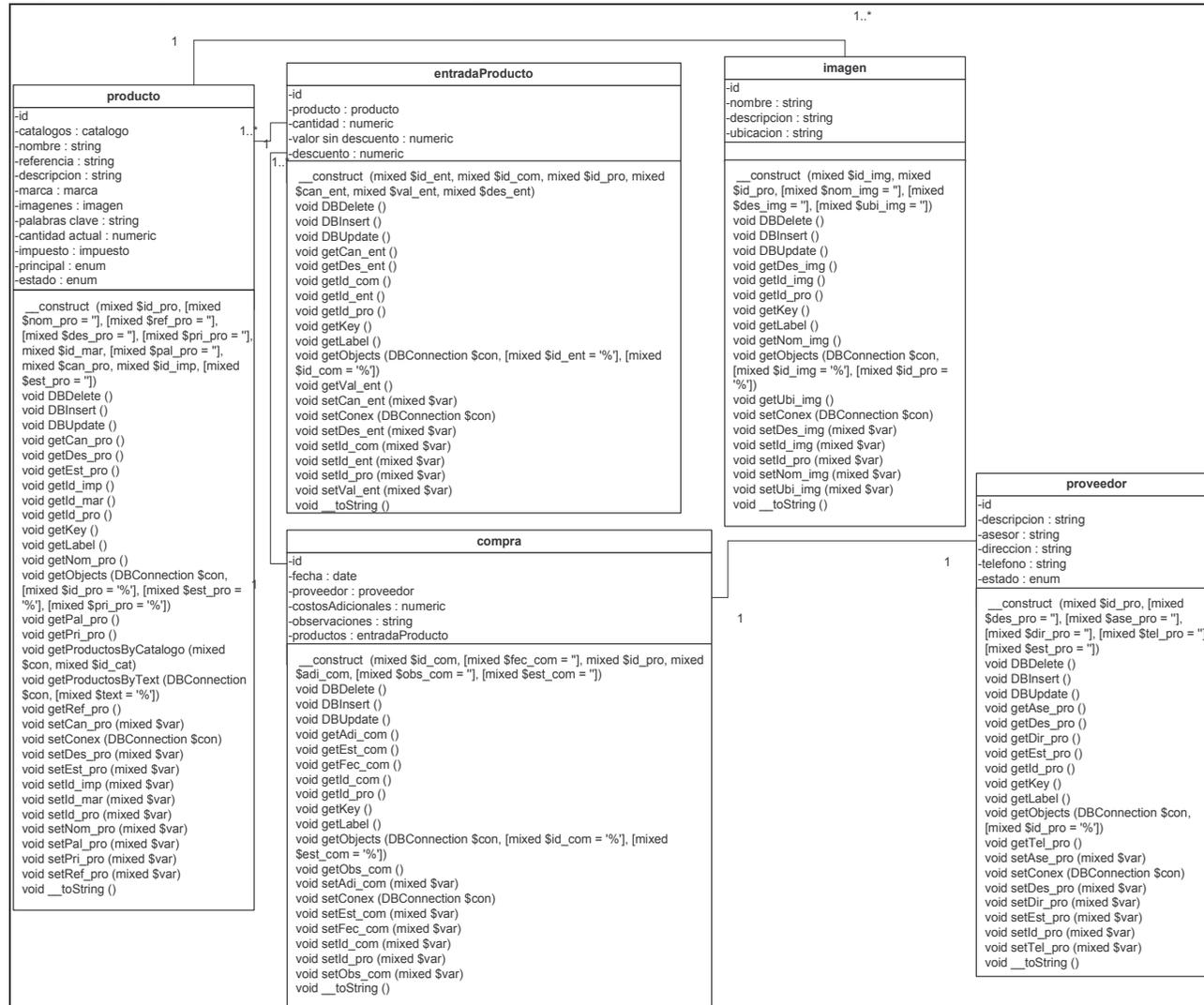


Figura 24 Diagrama de clases detallado - Lógica Comercial 3

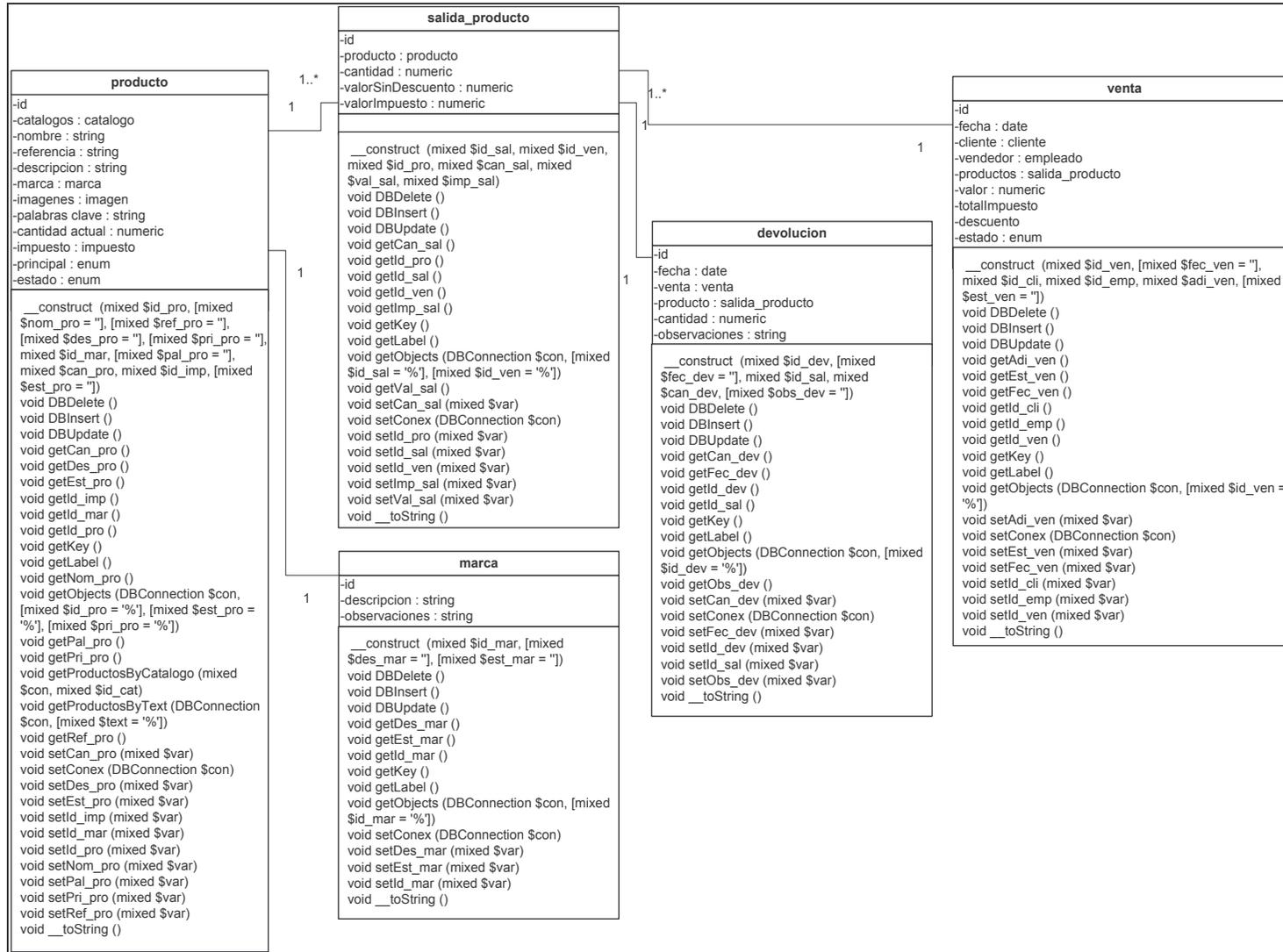
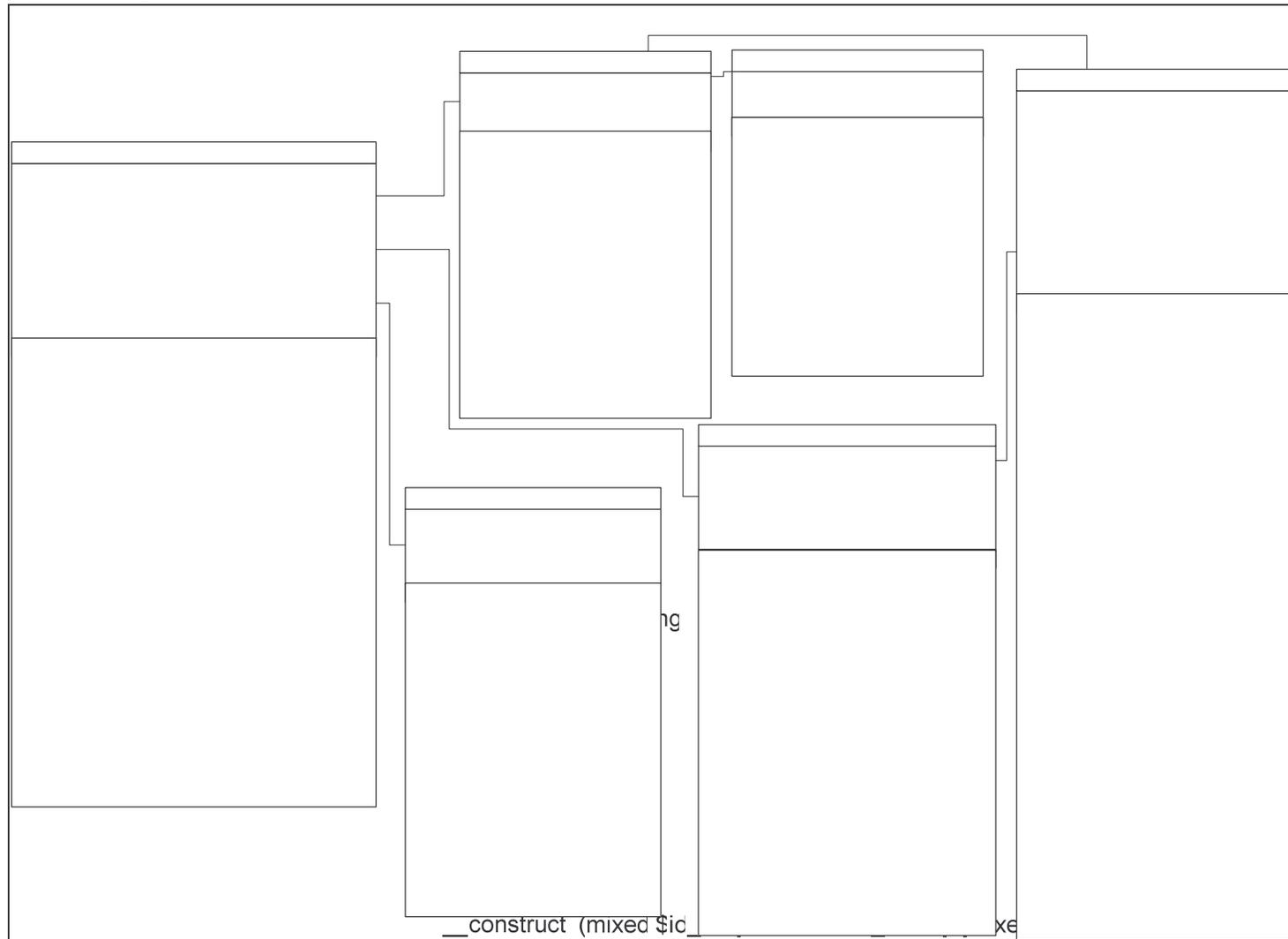


Figura 25 Diagramas de Clases detallado - Lógica comercial 4



```

__construct (mixed $ic
    [mixed $dir_cl = '' |mixed $tel_cl = '' |mixed $mov_cl = ''
    [mixed $ema_cl = '' |mixed $ic_ciu |mixed $coc_cl = '' |mixed
    $fin_cl ='])
void DBDelete ()
void DBInsert ()
void DBUpdate ()
void getCod_cl ()
void getDir_cl ()
void getEma_cl ()
    
```

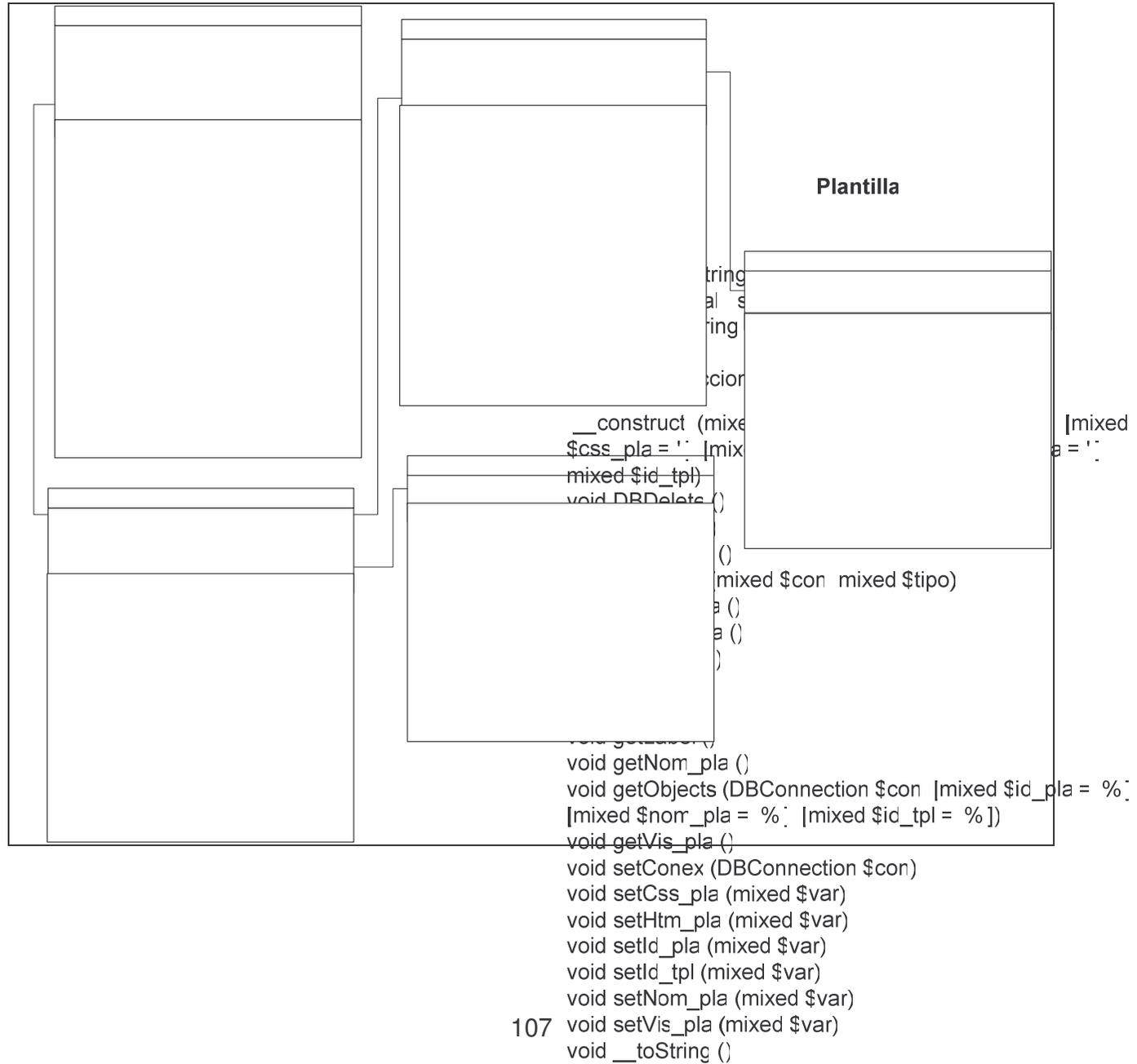
```

ciud
-ic
-descripcion string
-pais pais
-estado enum

__construct (mixed $ic
= '' |mixed $ic_pai |mix
void DBDelete ()
void DBInsert ()
void DBUpdate ()
void getDes_ciu ()
void getEst_ciu ()
void getIc_ciu ()
void getIc_pai ()
void getKey ()
void getLabel ()
void getObjects (DBCon
$ic_ciu = %' |mixed $i
void setConex (DBConr
void setDes_ciu (mixed
void setEst_ciu (mixed
void setIc_ciu (mixed $
void setIc_pai (mixed $
void __toString ()
    
```



Figura 26 Diagrama de clases - Manejo de apariencia de la tienda



-id  
 -plantilla Plantilla  
 -seccion seccion  
 -propiedad propiedad  
 -valor string

\_\_construct (mixed \$id\_pro, mixed \$css\_pla = '', [mixed \$id\_tpl])  
 void DBDelete ()  
 void DBInsert (mixed \$con, mixed \$tipo)  
 void DBUpdate (mixed \$con, mixed \$tipo)  
 void getId\_css ()  
 void getId\_pla ()  
 void getId\_pro ()  
 void getId\_se ()  
 void getId\_val ()  
 void getKey ()  
 void getLabel ()  
 void getObject (mixed \$id\_pla)  
 void getNom\_pla ()  
 void getObjects (DBConnection \$con [mixed \$id\_pla = %], [mixed \$nom\_pla = %], [mixed \$id\_tpl = %])  
 void getVis\_pla ()  
 void setConex (DBConnection \$con)  
 void setCss\_pla (mixed \$var)  
 void setId\_pla (mixed \$var)  
 void setId\_tpl (mixed \$var)  
 void setNom\_pla (mixed \$var)  
 void setVis\_pla (mixed \$var)  
 void \_\_toString ()

Figura 27 Diagrama de clases - Gestión de contenidos

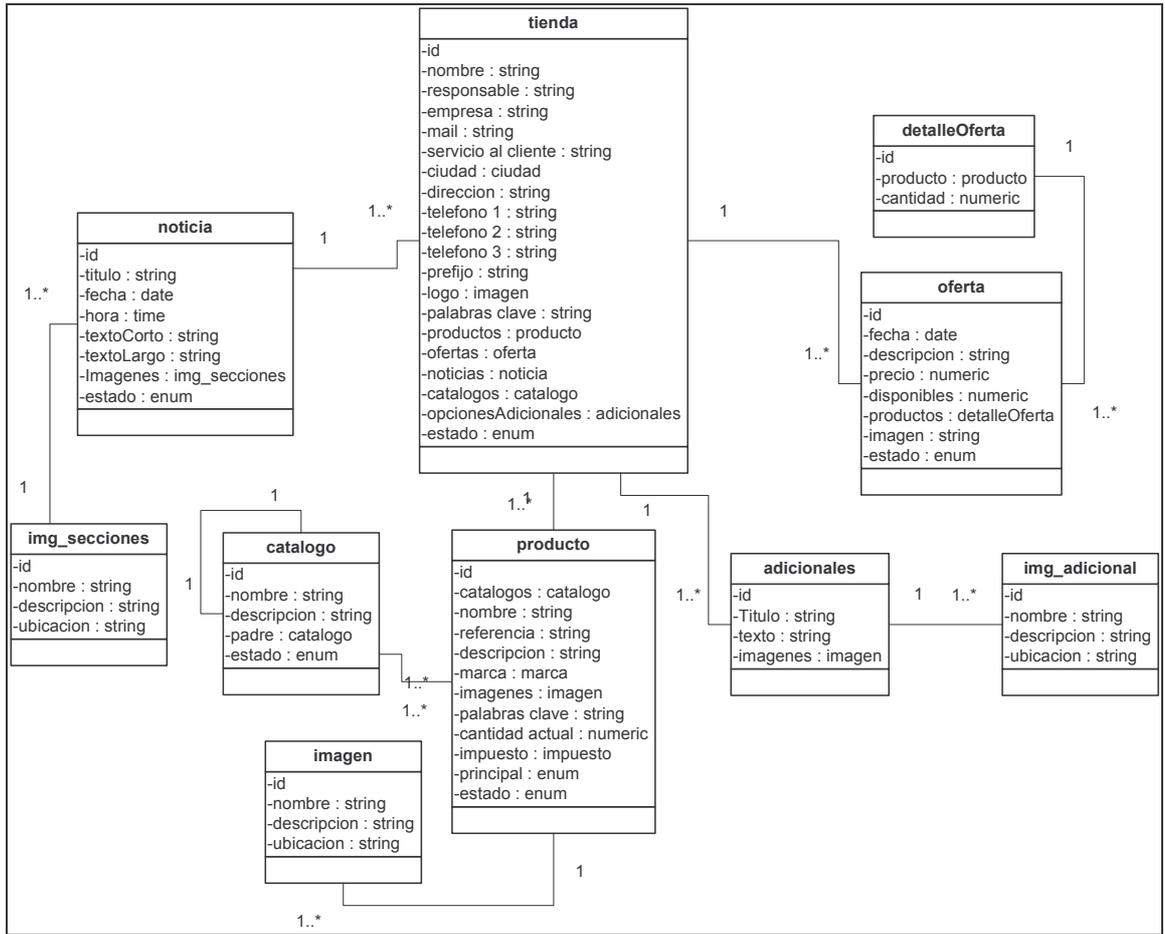




Figura 28 Diagrama de clases detallado - Gestión de contenidos 1





Figura 29 Diagrama de clases detallado - Gestión de contenidos 2

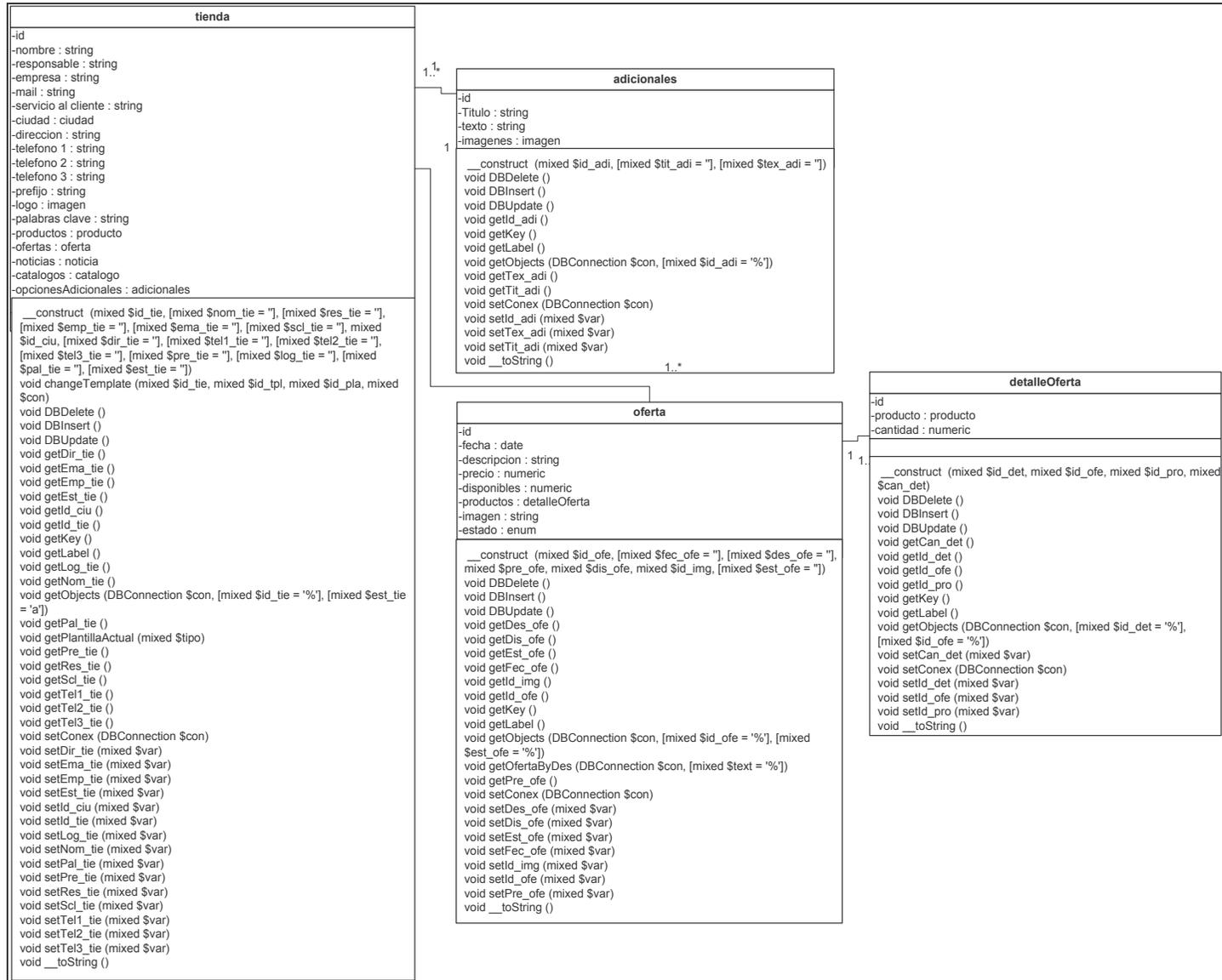


Figura 30 Diagrama de clases detallado – XwebComponents

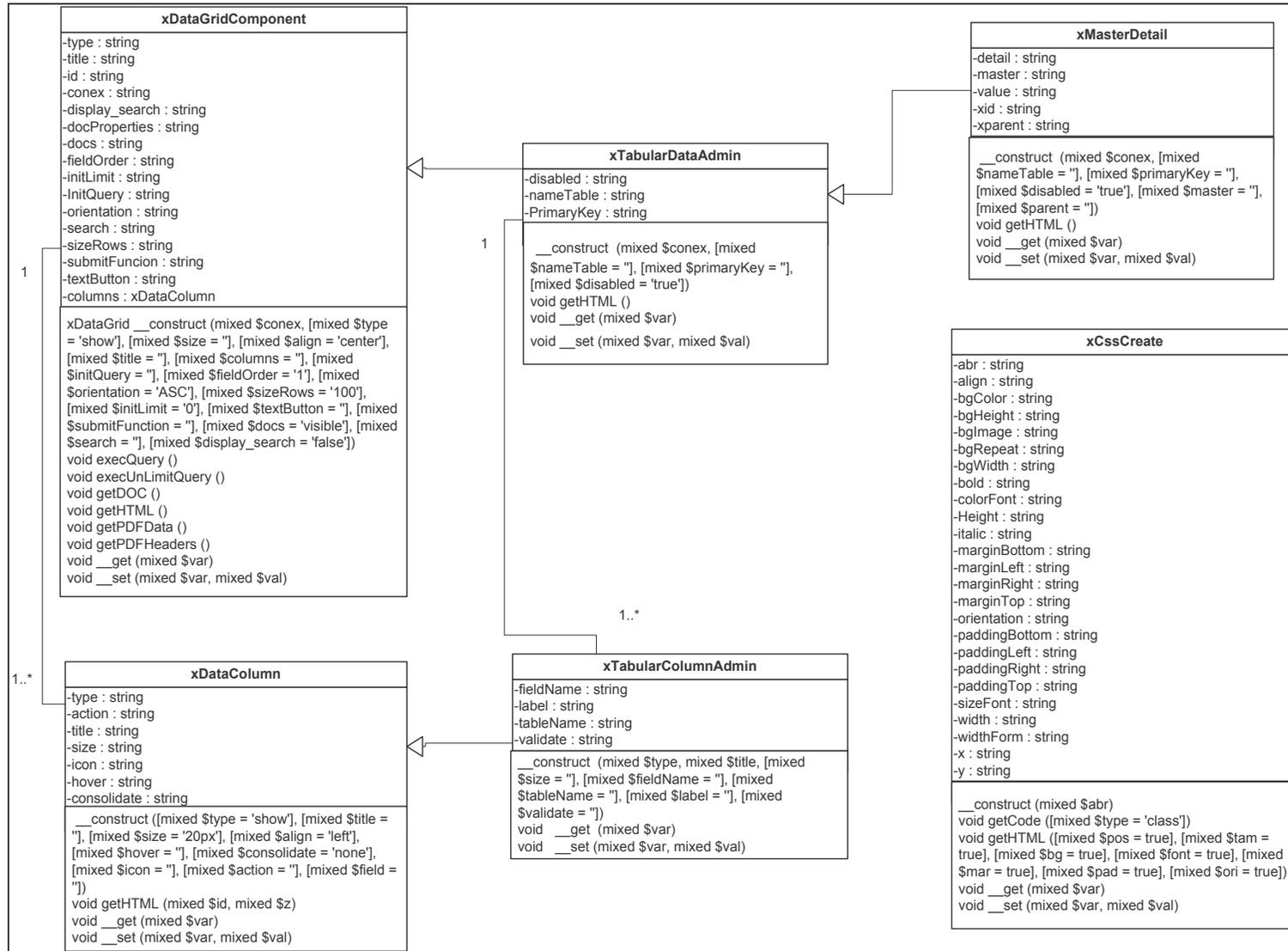


Figura 31 Diagrama de clases detallado - Aplicación Móvil 1

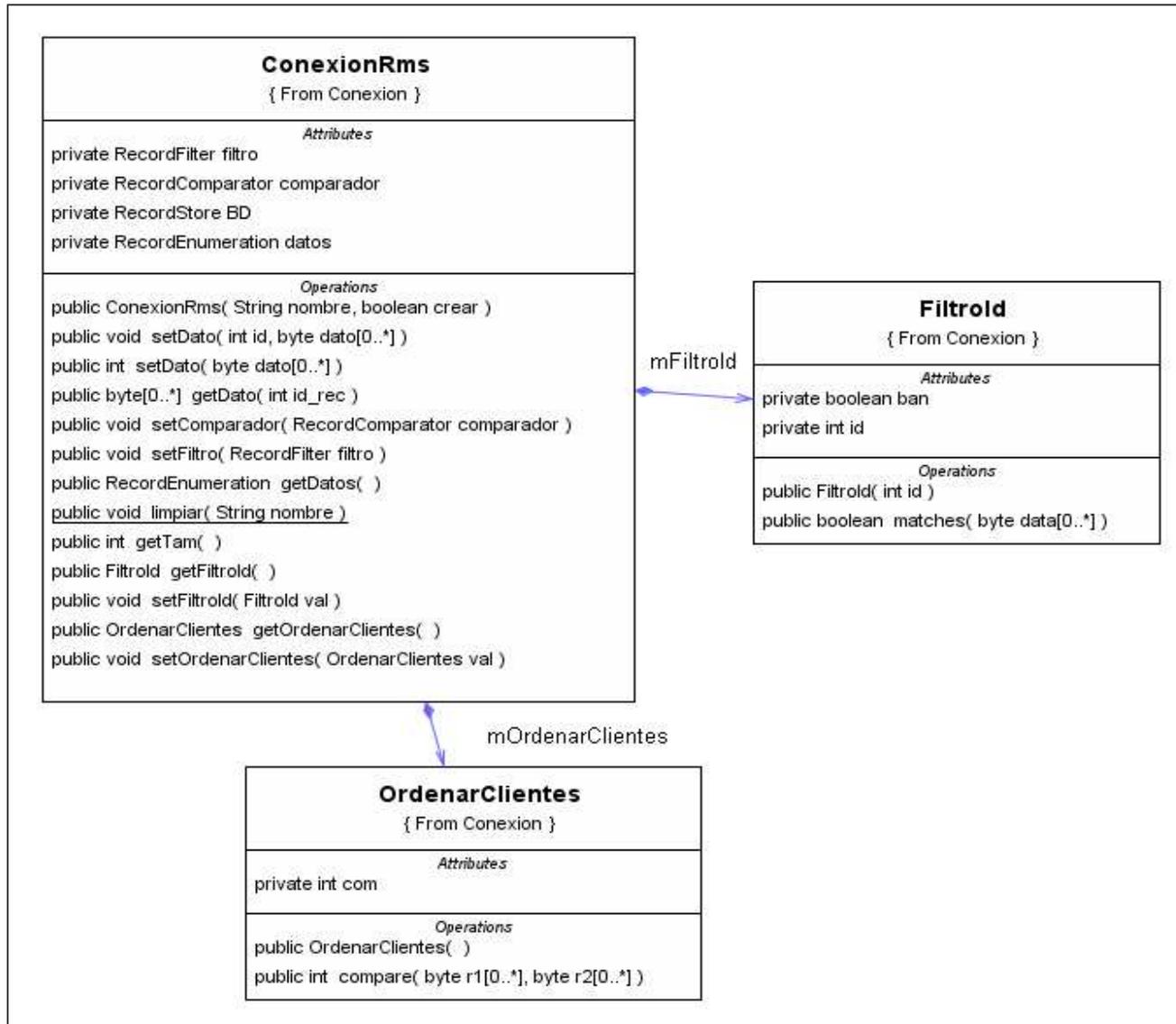


Figura 32 Diagrama de clases detallado - Aplicación Móvil 2

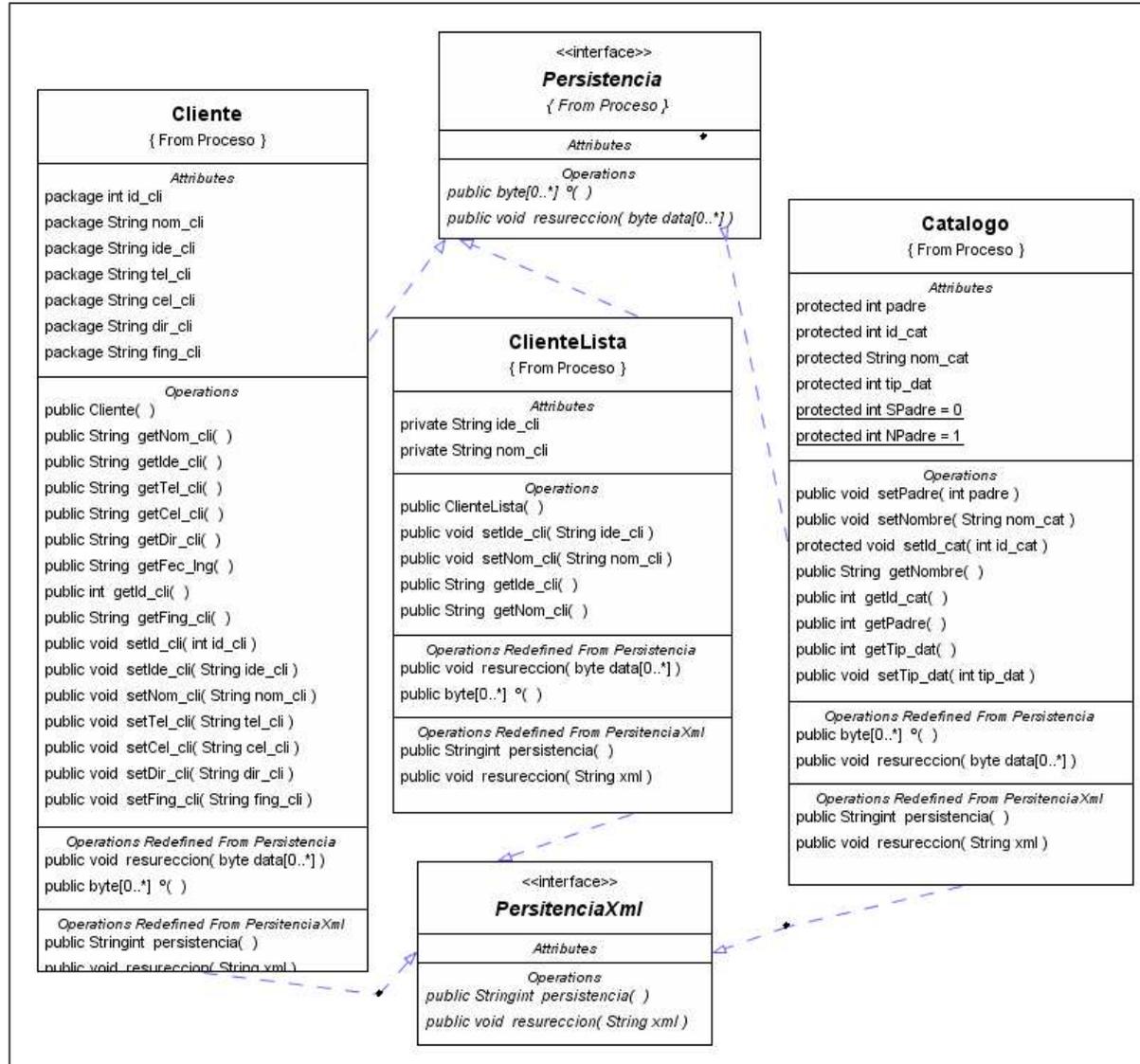


Figura 33 Diagrama de clases detallado - Aplicación Móvil 3

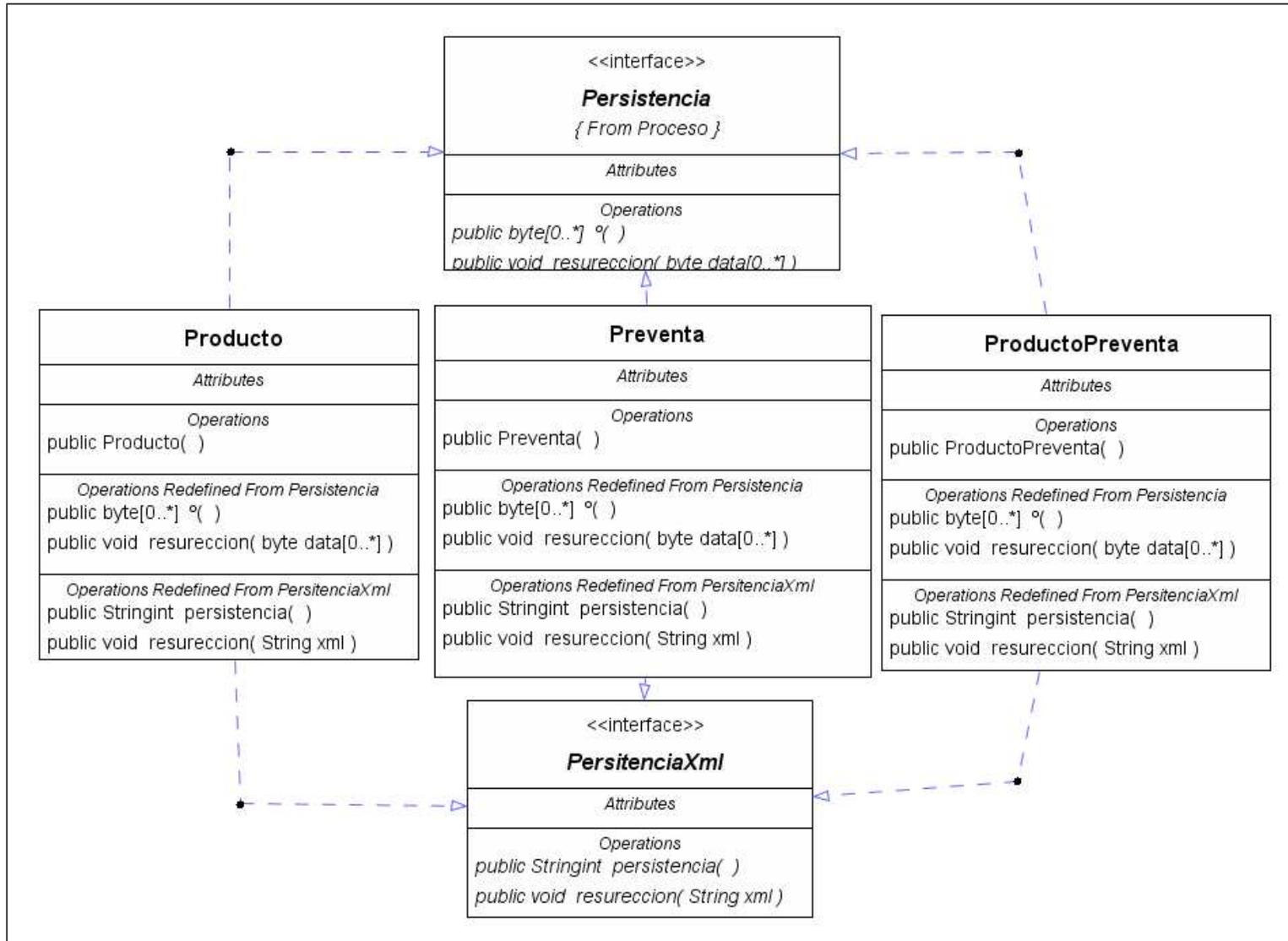


Figura 34 Diagrama de clases detallado - Aplicación Móvil 4

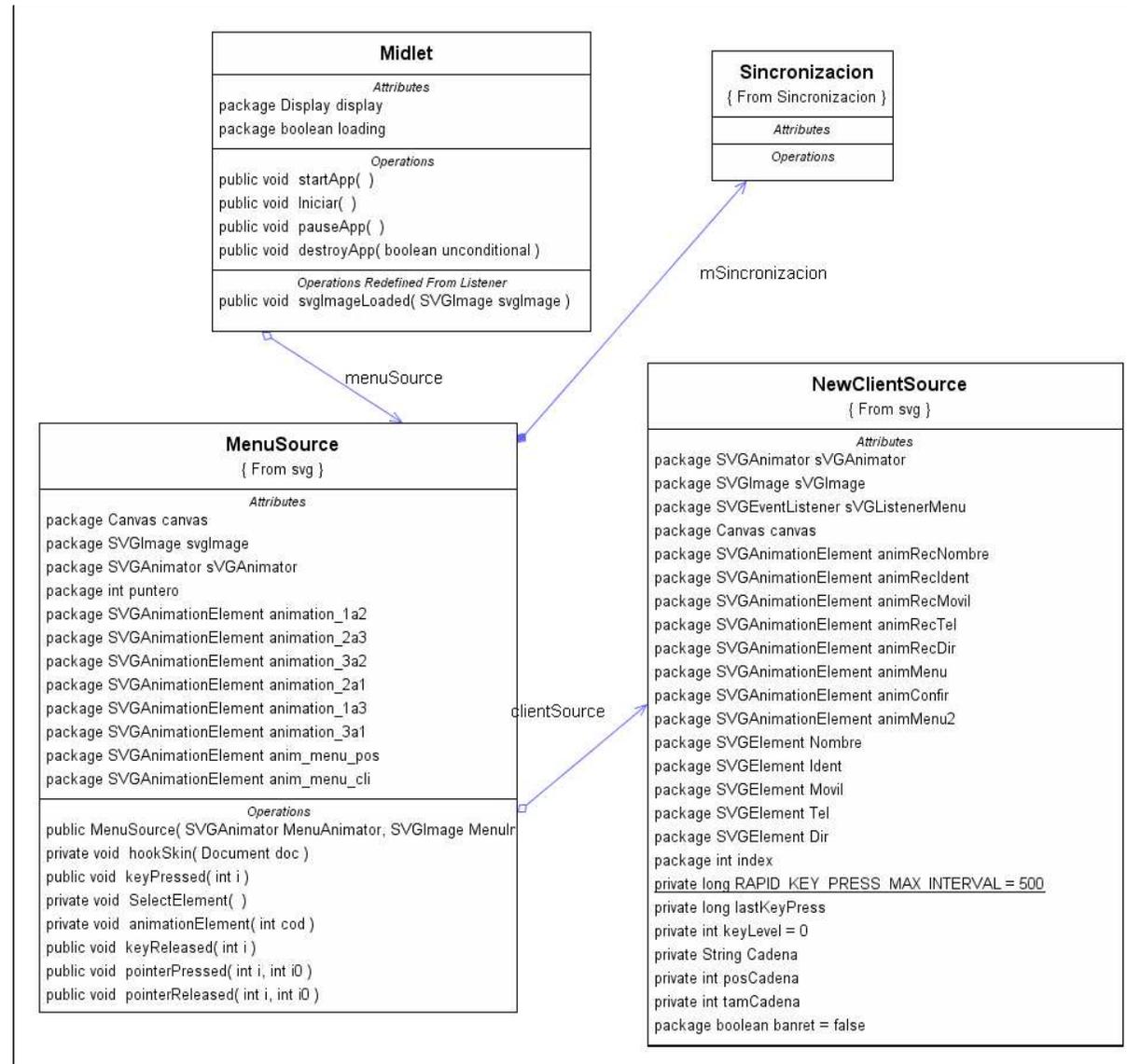


Figura 35 Diagrama de clases detallado - Aplicación Móvil 5

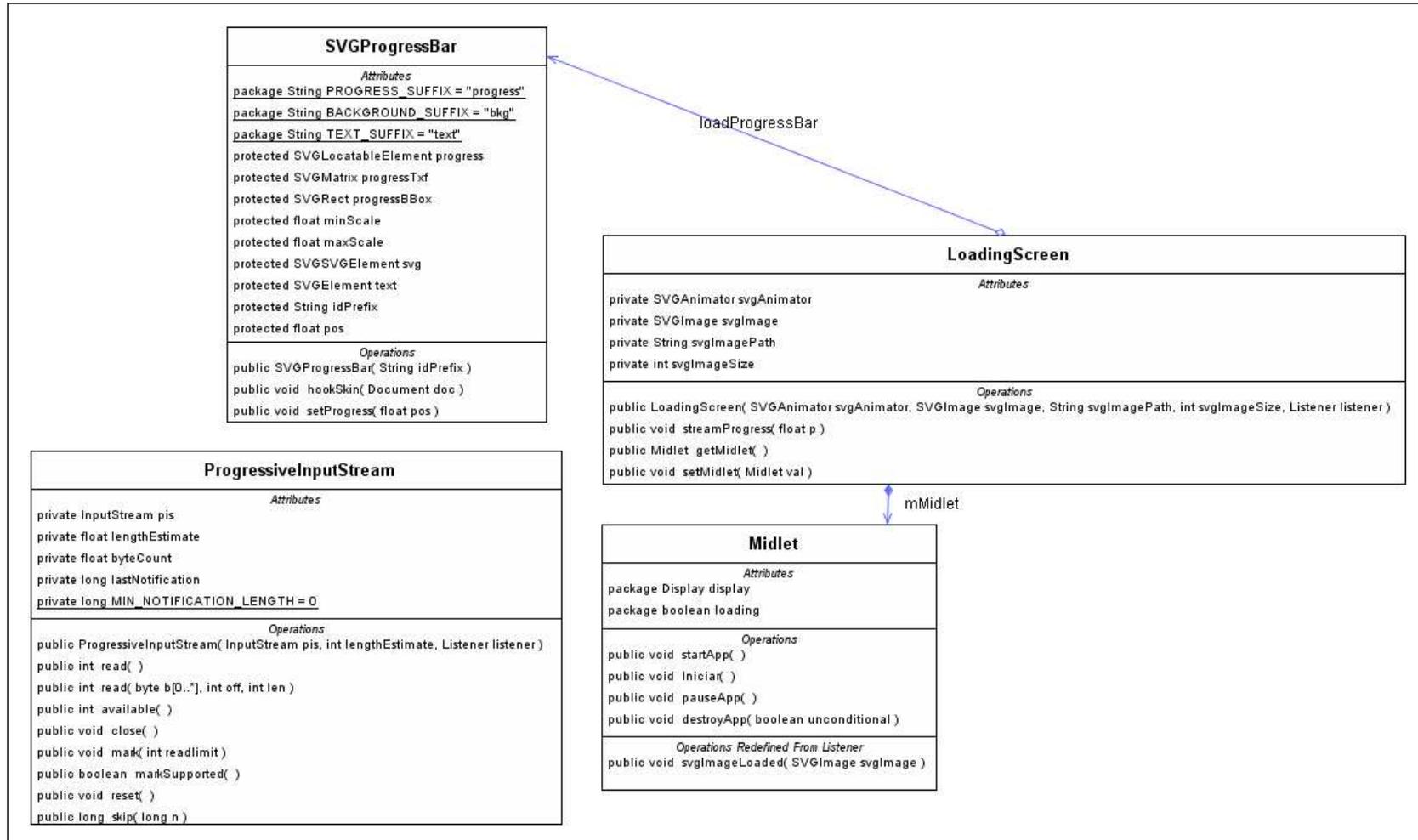
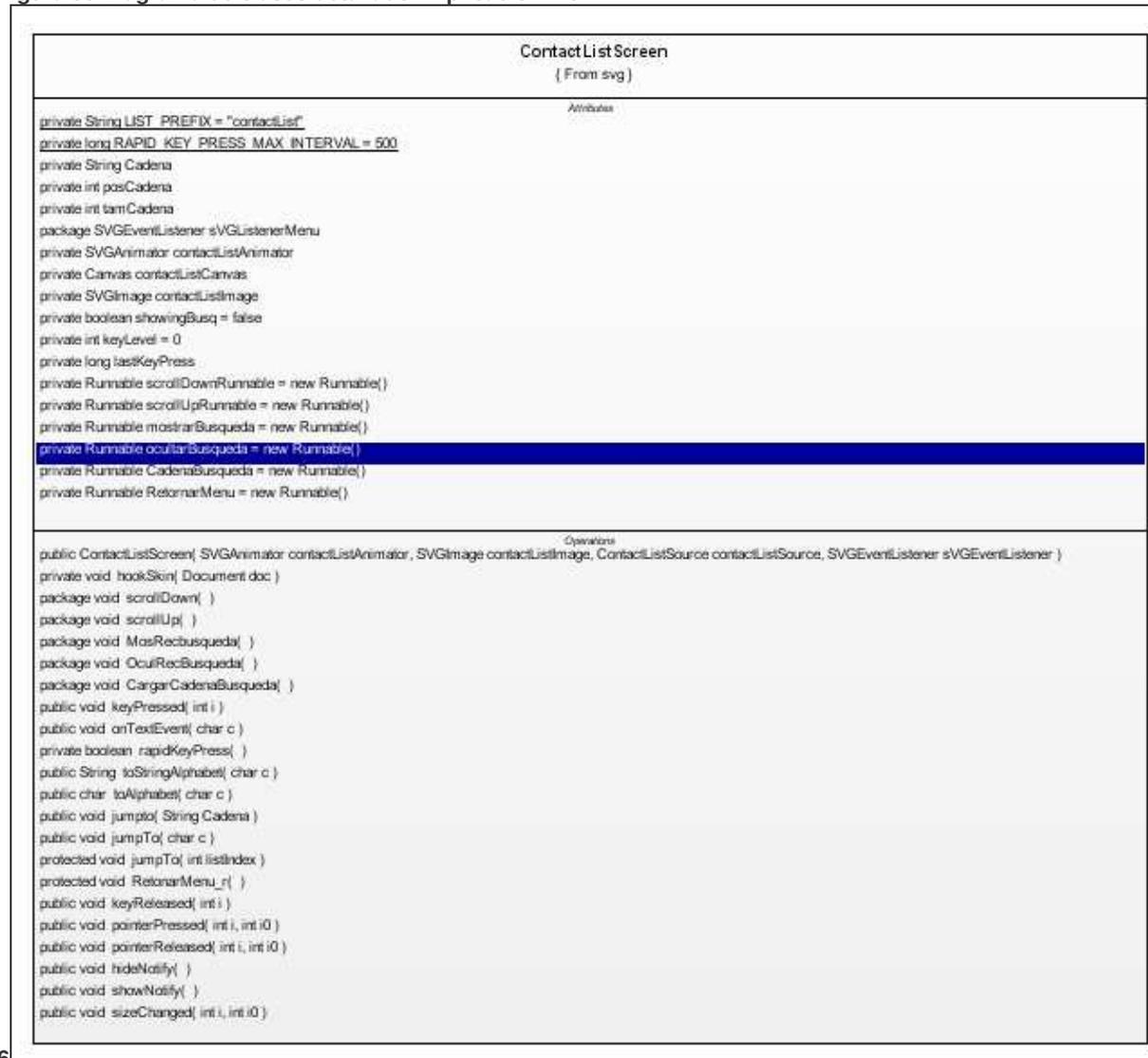
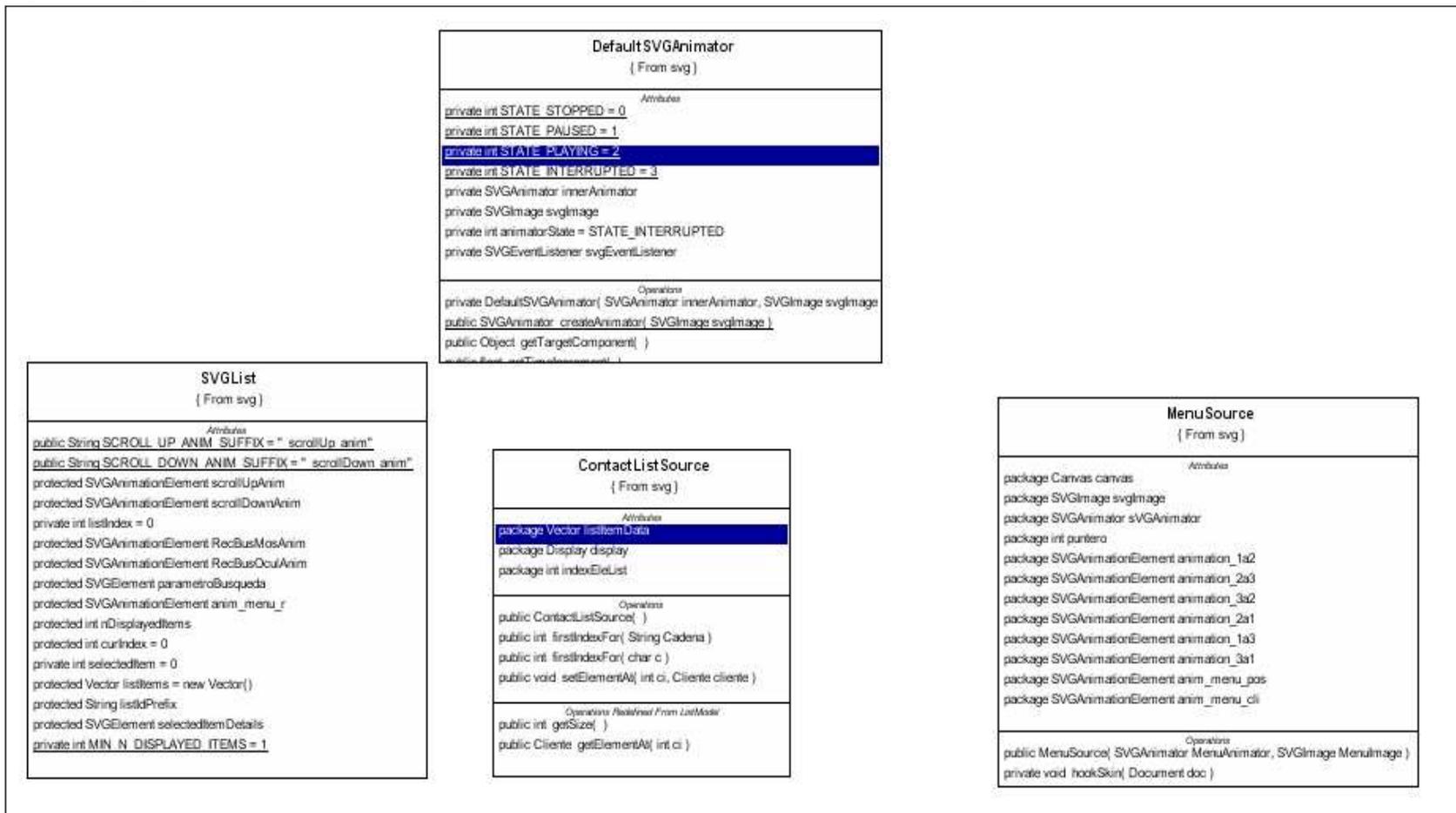


Figura 36 Diagrama de clases detallado - Aplicación Móvil



6

Figura 37 Diagrama de clases detallado - Aplicación Móvil 7





**4.3.5. Diagramas de paquetes.** Se presentan los diagramas de paquetes de la aplicación Web y la aplicación móvil.

**4.3.5.1. Aplicación web** A continuación de la figura 34 a la figura 37 se muestran los diagramas de paquetes de la aplicación web.

Figura 38 Diagrama de paquetes - Gestión Comercial

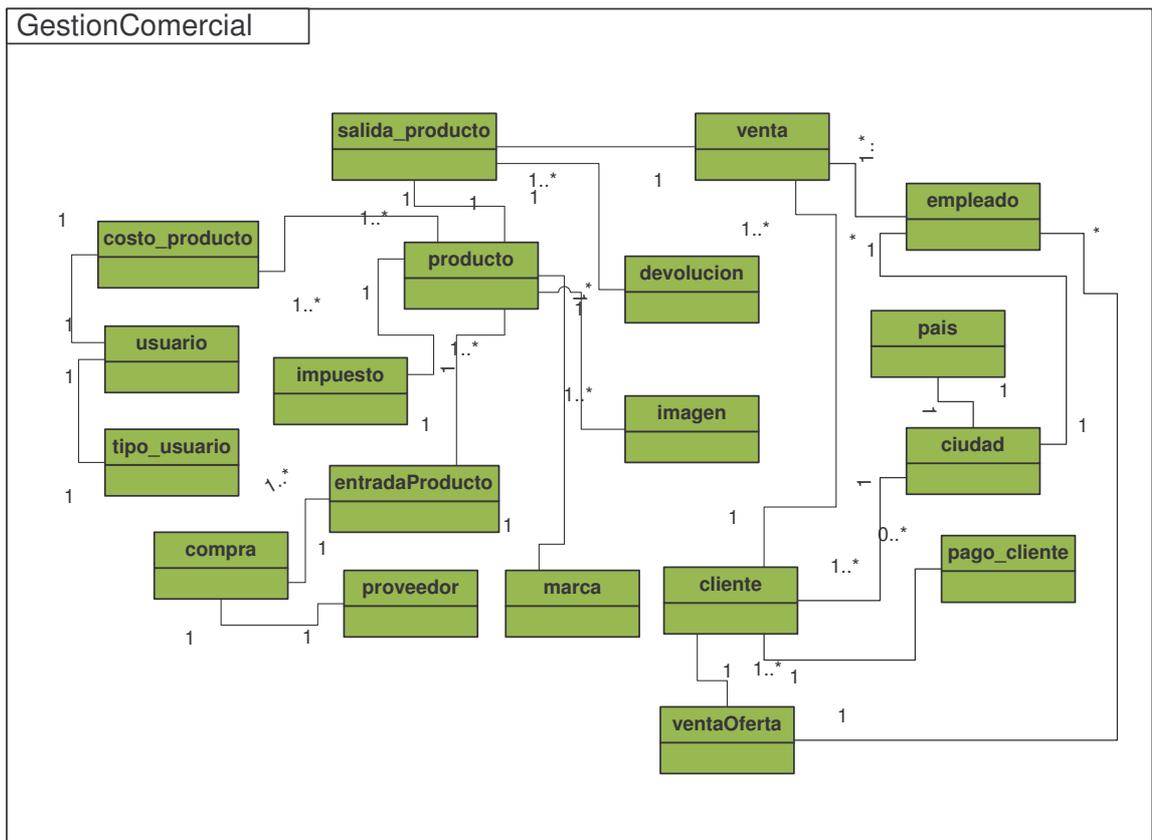


Figura 39 Diagrama de Paquetes - FaceStore

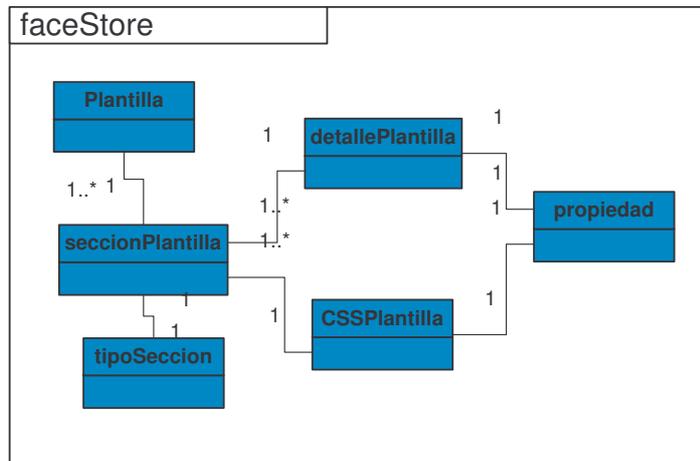


Figura 40 Diagrama de paquetes – Content

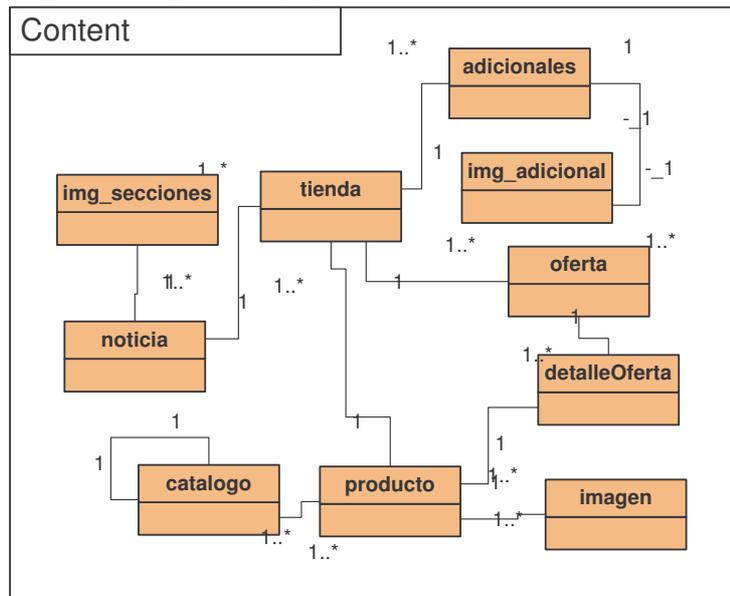
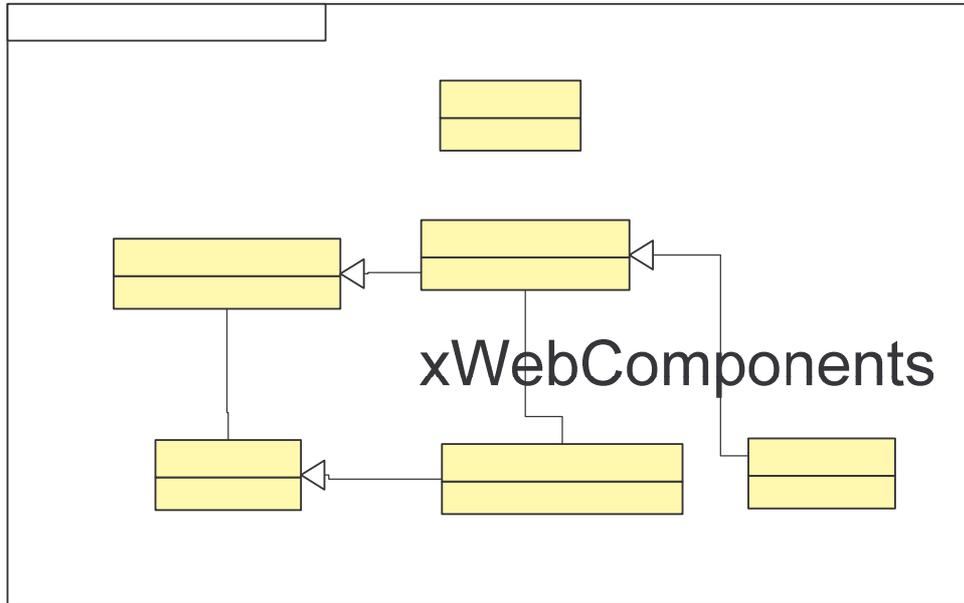


Figura 41 Diagrama de paquetes – XwebComponents



**4.3.5.2. Aplicación móvil.** A continuación de la figura 38 a la figura 40 se muestran los diagramas de paquetes de la aplicación móvil.

**xDataGridComponent**

1

1..\*

**xDataColumn**

Figura 42 Diagrama de paquetes - Aplicación Móvil - Proceso

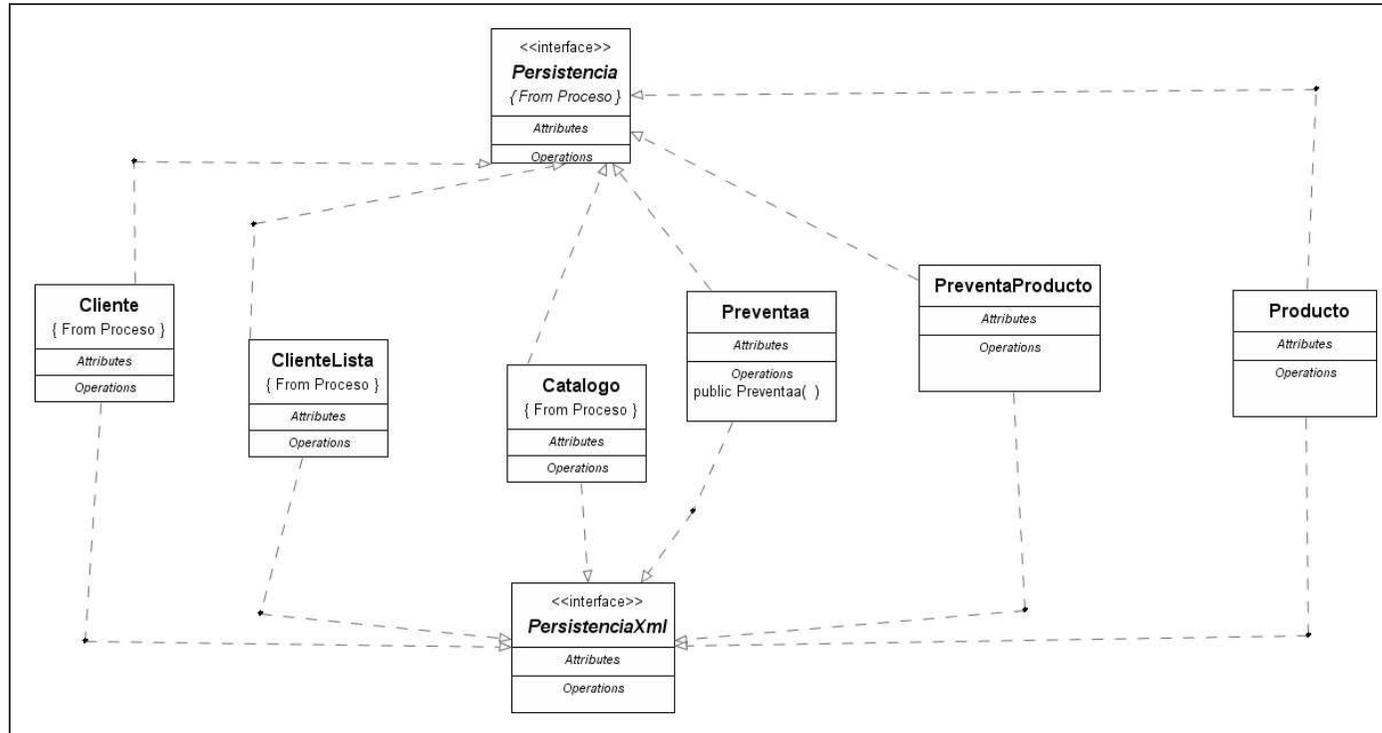
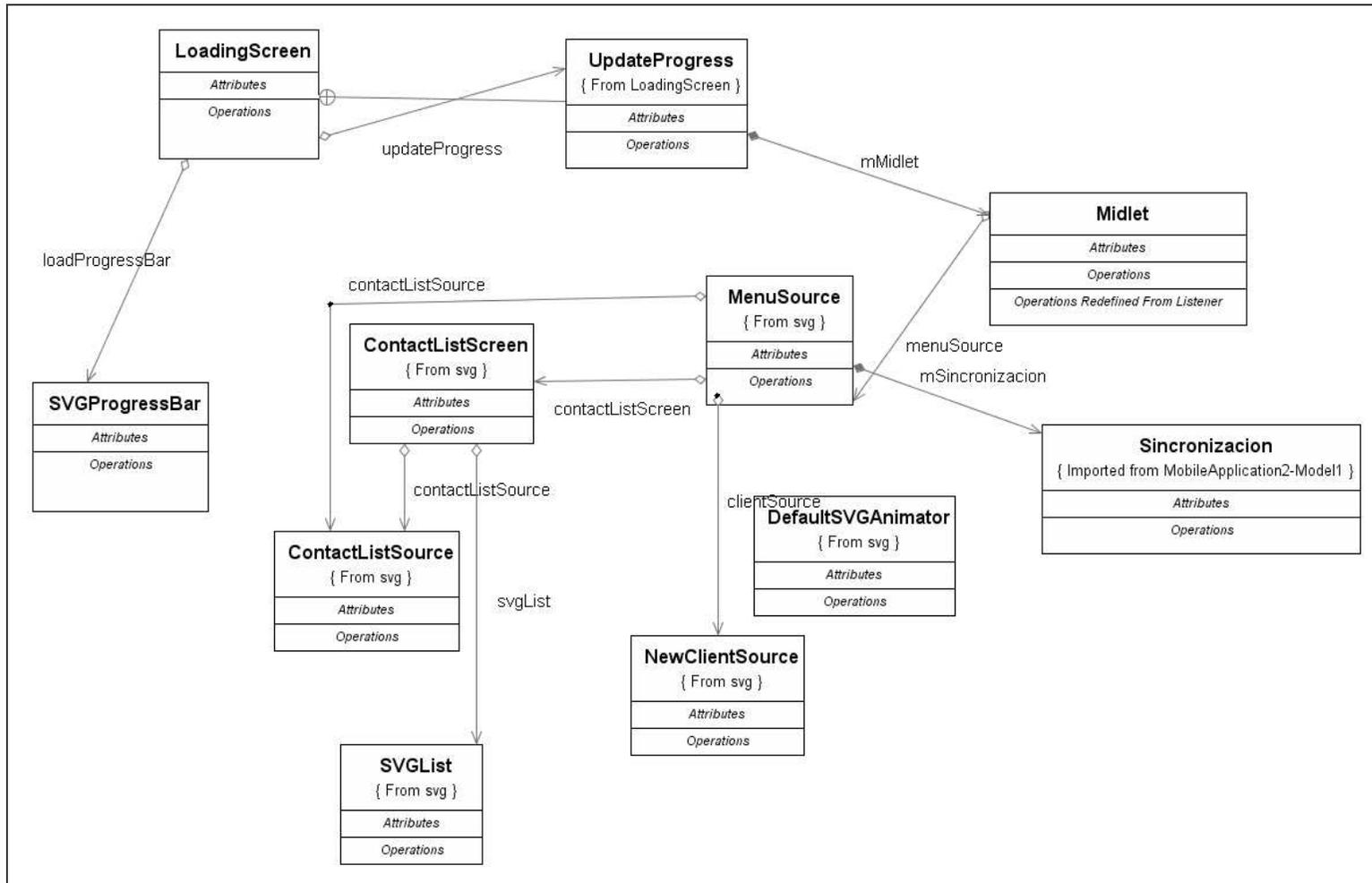
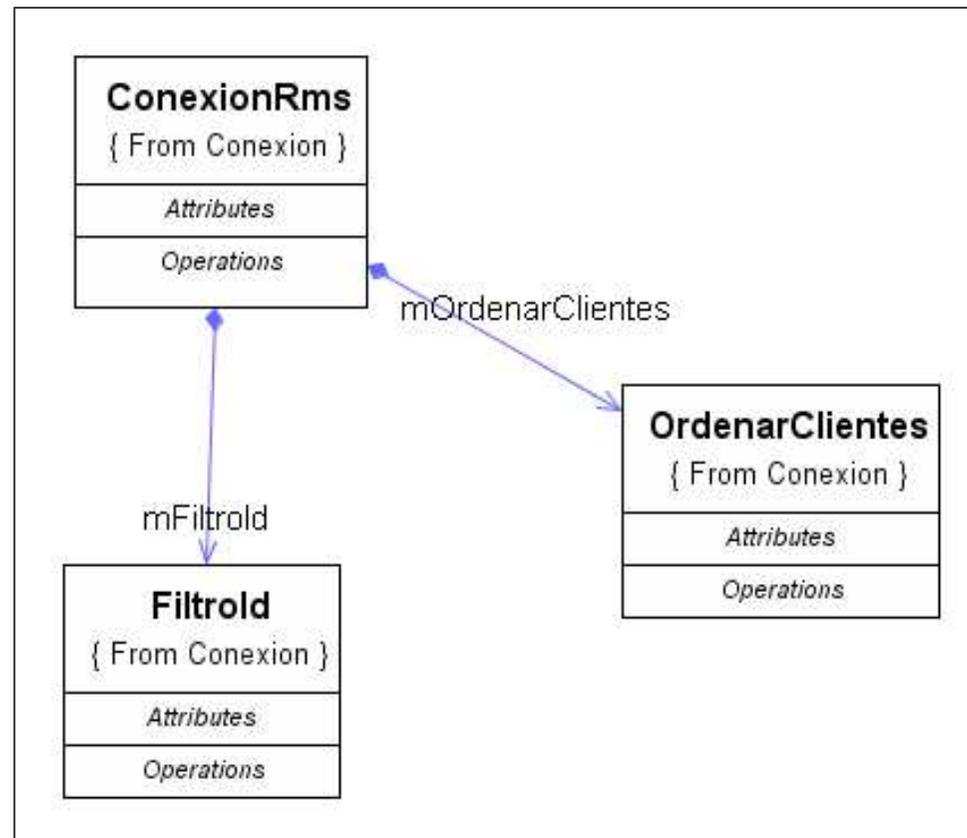


Figura 43 Diagrama de paquetes - Aplicación Móvil - Paquete SVG

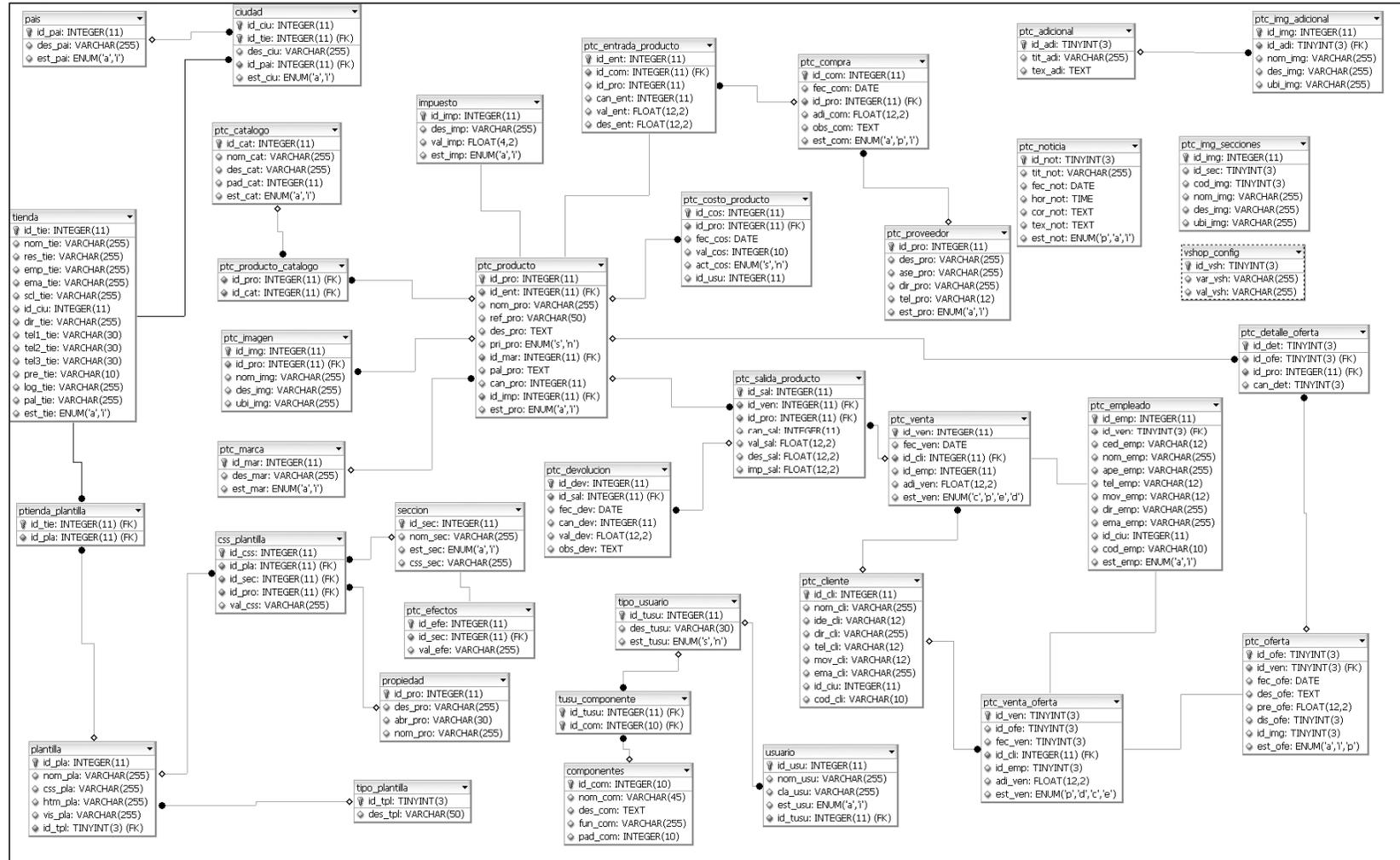


**Figura 44 Diagrama de paquetes - Aplicación Móvil - Paquete Conexión**



**4.3.6. Diagrama Entidad Relación** En el diagrama se presenta las tablas del sistema y un ejemplo de las tablas propias de una tienda generada cuyo prefijo inicial es ptc, cada tienda generada tendrá su grupo de tablas propias.

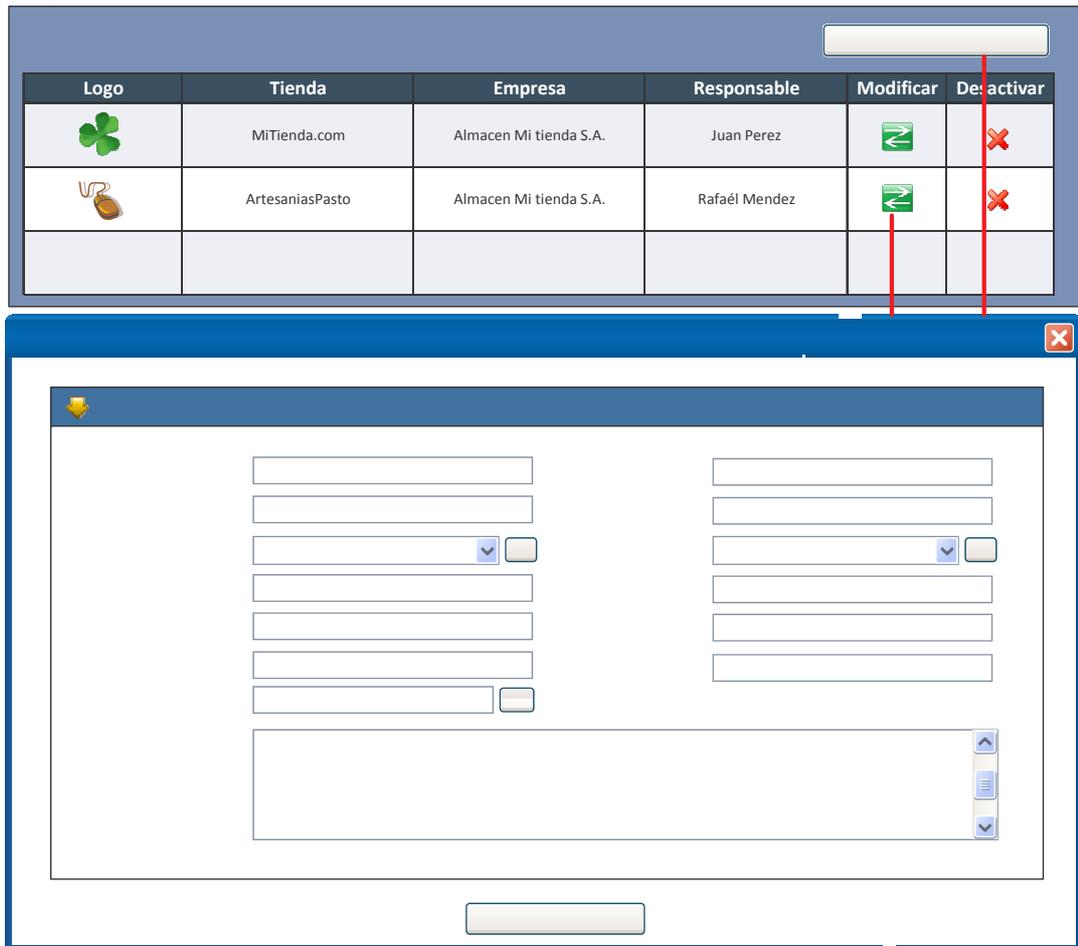
**Figura 45 Diagrama entidad relación**



#### 4.4. DISEÑO

En este punto de la figura 46 a la figura 70 se muestra el diseño de pantalla que se implementaron después para desarrollar las opciones de VSHOP CMS.

Figura 46 Diseño de pantalla - Administración de tiendas



Nueva Tienda

130

Nombre:

Responsable:

País: Colombia

Dirección:

Figura 47. Diseño de pantalla - Diseño de la tienda

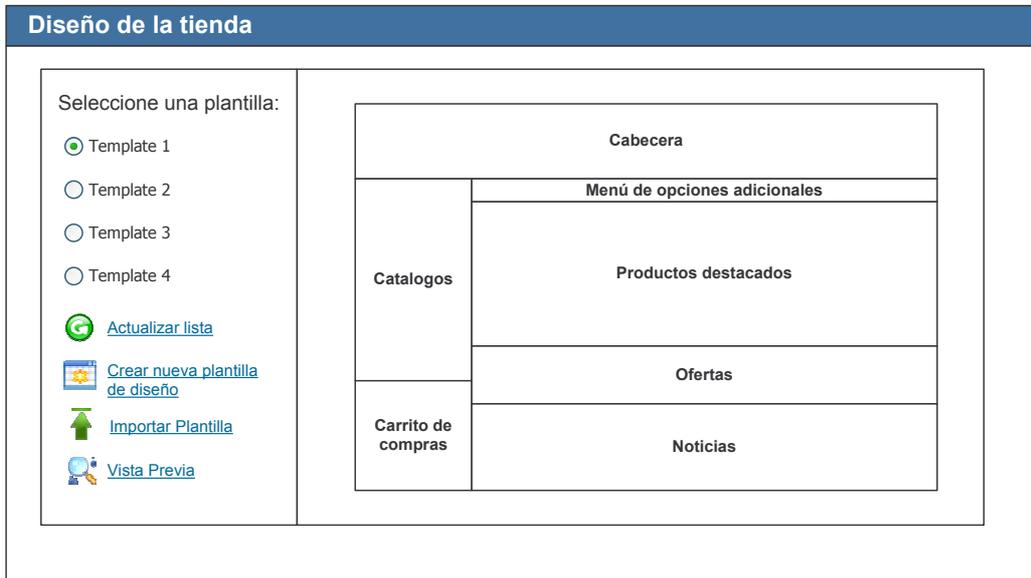


Figura 48 Diseño de pantalla - Diseño de catálogos

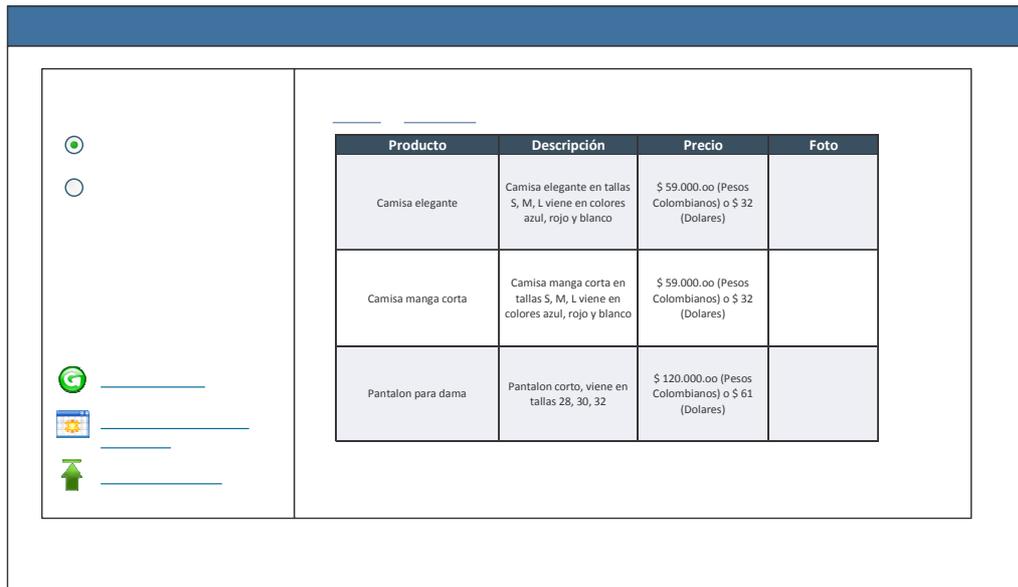
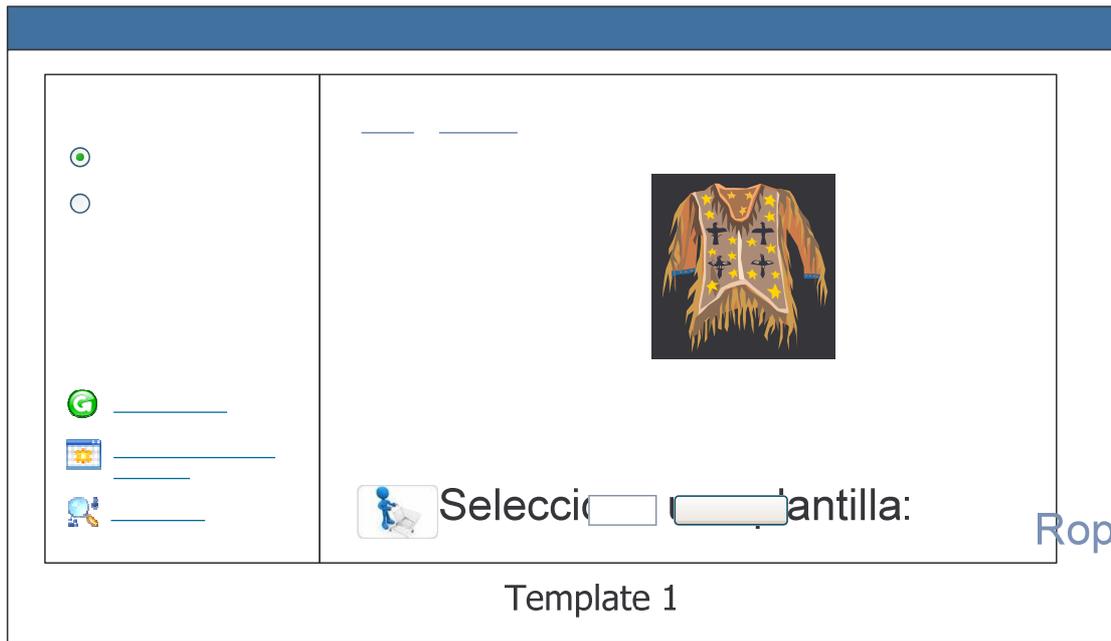


Figura 49 Diseño de pantalla - Diseño sección Detalle – producto



Template 2

Camisa e

Figura 50 Diseño de pantalla - Datos de conexión a servidor



Actualizar lista

Crear nueva plantilla de diseño

Vista Previa

Ca

Figura 51 Diseño de pantalla - Crear plantilla de tienda

### Nueva plantilla CSS de diseño de la tienda

|   |   |
|---|---|
| Encabezado  | <b>Posición:</b> X: <input type="text"/> Y: <input type="text"/><br><b>Tamaño:</b> Ancho: <input type="text"/> Alto: <input type="text"/><br><b>Fondo:</b> Color: <input type="text"/><br>Imagen: <input type="text"/><br>Repetir en X: <input type="checkbox"/> Repetir en Y: <input type="checkbox"/><br>Ancho Mínimo: <input type="text"/> Alto Mínimo: <input type="text"/><br><b>Fuente:</b> Familia: <input type="text"/> <input type="text"/><br>Tamaño: <input type="text"/><br>Negrita: <input type="checkbox"/> Cursiva: <input type="checkbox"/><br>Color: <input type="text"/><br><b>Margenes:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/><br><b>Relleno:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/> |
| Catalogos   |   |
| Opciones adicionales  |   |
| Productos principales   |   |
| Ofertas   |   |
| Carrito de compras  |   |
| Noticias  |   |
|   |   |
|   |   |

### Guardar plantilla

Nombre:

Archivo de apoyo a la distribución:

Imagen de Vista previa:

Figura 52 Diseño de pantalla - Crear plantilla de catálogos

**Fondo:** Color:    
Imagen:   
Repetir en X:  Repetir en Y:   
Ancho Mínimo:  Alto Mínimo:

**Fuente:** Familia:   
Tamaño:   
Negrita:  Cursiva:   
Color:  

**Margenes:** Izq:  Der:  Sup:  Inf:   
**Relleno:** Izq:  Der:  Sup:  Inf:

Fila par

Fila impar

Sobre la fila

Figura 53 Diseño de pantalla - Crear plantilla de detalle producto

### Nueva plantilla CSS de detalles de productos

|   |   |
|---|---|
| Ruta de Catalogo  | <b>Posición:</b> X: <input type="text"/> Y: <input type="text"/>  |
| Nombre  | <b>Tamaño:</b> Ancho: <input type="text"/> Alto: <input type="text"/>   |
| Descripción   | <b>Fondo:</b> Color: <input type="text"/>  |
| Precio  | Imagen: <input type="text"/>  |
| Imágenes  | Repetir en X: <input type="checkbox"/> Repetir en Y: <input type="checkbox"/>   |
| Detalle compra  | Ancho Mínimo: <input type="text"/> Alto Mínimo: <input type="text"/>  |
|   | <b>Fuente:</b> Familia: <input type="text" value="Arial, Helvetica, sans-serif"/>   |
|   | Tamaño: <input type="text" value="8"/>  |
|   | Negrita: <input type="checkbox"/> Cursiva: <input type="checkbox"/>   |
|   | Color: <input type="text"/>                |
|   | <b>Margenes:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/>      |
|   | <b>Relleno:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/>       |

### Guardar plantilla

Nombre:

Archivo de apoyo a la disdribución:

Imagen de Vista previa:

Figura 54 Diseño de pantalla - Editar Plantillas

Editar plantillas VSHOP CMS

|   |  |  |
|---|--|--|
| <b>PAGINA PRINCIPAL</b><br>   | <b>CATALOGOS</b><br>   | <b>DETALLE PRODUCTO</b><br>  |
| <p> <input checked="" type="radio"/> Template 1<br/> <input type="radio"/> Template 2                 </p> <p style="text-align: center; border: 1px solid gray; padding: 2px 10px;">                     Editar Plantilla                 </p> | <p> <input checked="" type="radio"/> Template 1<br/> <input type="radio"/> Template 2<br/> <input checked="" type="radio"/> Template 1<br/> <input type="radio"/> Template 2                 </p> <p style="text-align: center; border: 1px solid gray; padding: 2px 10px;">                     Editar Plantilla                 </p> | <p> <input checked="" type="radio"/> Template 1<br/> <input type="radio"/> Template 2<br/> <input checked="" type="radio"/> Template 1<br/> <input type="radio"/> Template 2                 </p> <p style="text-align: center; border: 1px solid gray; padding: 2px 10px;">                     Editar Plantilla                 </p> |

Nueva plantilla CSS de detalles de productos

|                  |  |
|------------------|--|
| Ruta de Catalogo | <b>Posición:</b> X: <input type="text"/> Y: <input type="text"/>   |
| Nombre           | <b>Tamaño:</b> Ancho: <input type="text"/> Alto: <input type="text"/>  |
| Descripción      | <b>Fondo:</b> Color: <input type="text"/><br>Imagen: <input type="text"/>  |
| Precio           | Repetir en X: <input type="checkbox"/> Repetir en Y: <input type="checkbox"/>  |
| Imágenes         | Ancho Mínimo: <input type="text"/> Alto Mínimo: <input type="text"/>   |
| Detalle compra   | <b>Fuente:</b> Familia: <input type="text" value="Arial, Helvetica, sans-serif"/>  |
|                  | Tamaño: <input type="text" value="8"/>   |
|                  | Negrita: <input type="checkbox"/> Cursiva: <input type="checkbox"/>  |
|                  | Color: <input type="text"/>  |
|                  | <b>Margenes:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/> |
|                  | <b>Relleno:</b> Izq: <input type="text"/> Der: <input type="text"/> Sup: <input type="text"/> Inf: <input type="text"/>  |

Figura 55 Diseño pantalla - Importar plantilla

**Importar plantillas VSHOP CMS**

|  |   |  |
|--|---|--|
| <b>PAGINA PRINCIPAL</b><br> | <b>CATALOGOS</b><br> | <b>DETALLE PRODUCTO</b><br> |
|--|---|--|

Nombre Plantilla:

Archivo CSS:

Archivo de distribución:

Archivo Vista previa:



Figura 56 Diseño de pantalla - Gestión de productos

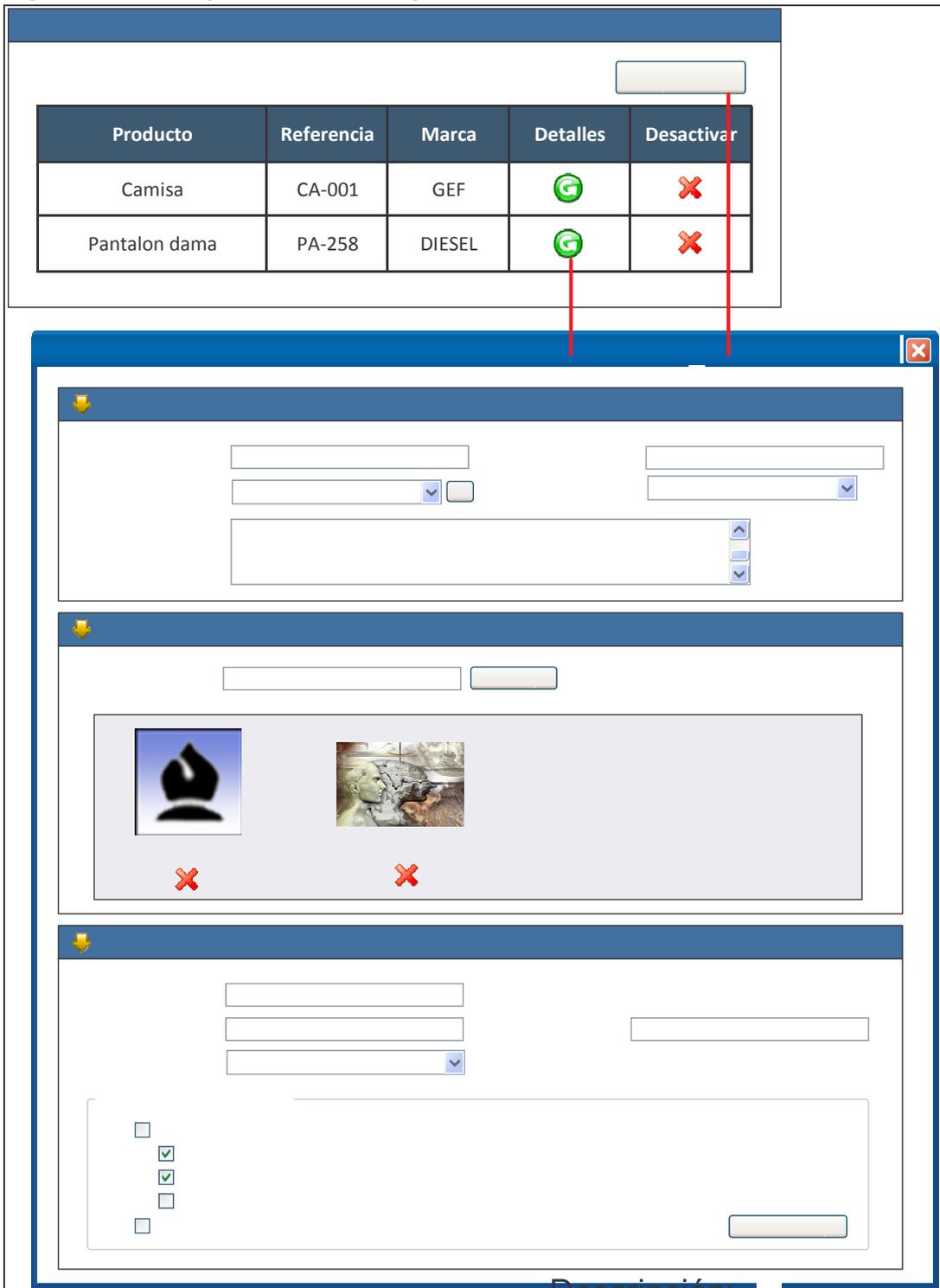


Figura 57 Diseño de pantalla - Registrar compra

| Fecha      | Proveedor            | Costos adicionales | Observaciones           | Detalles | Eliminar |
|------------|----------------------|--------------------|-------------------------|----------|----------|
| 20/02/2008 | SANYO                | 2500000            | Pago de fletes          |          |          |
| 21/02/2008 | Distribuciones Éxito | 450000             | Pago adicional comision |          |          |
|            |                      |                    |                         |          |          |

| Referencia | Producto              | Cantidad | Valor Compra | Descuento | Eliminar |
|------------|-----------------------|----------|--------------|-----------|----------|
| h3025      | Portatil Compaq h3025 | 2        | 4800000      | 200000    |          |
|            |                       |          |              |           |          |

Proveedor: Seleccione un proveedor

Costo Adicional:

Buscar Producto:



Figura 58 Diseño de pantalla - Registrar Venta

| Fecha      | Cliente       | Estado     | Detalles | Eliminar | Confirmar despacho |
|------------|---------------|------------|----------|----------|--------------------|
| 20/02/2008 | Andrea Serna  | Pendiente  |          | ✖        |                    |
| 21/02/2008 | Rafaél Mendez | Despachada |          | ✖        |                    |

---

**Registrar Venta**

+ **Información productos**

Vendedor:  Fecha:  ...

Buscar Producto:

Cantidad:  Descuento:

Subtotal: 100000 Impuestos: 16000 Descuento: 2000  
Total Pago: 114000

| Referencia | Producto  | Cantidad | Valor Sin Dto | Impuesto | Eliminar |
|------------|-----------|----------|---------------|----------|----------|
| h3025      | Mouse usb | 1        | 100000        | 16000    | ✖        |
|            |           |          |               |          |          |

+ **Datos Cliente**

Identificación:

Nombre:

Dirección:  Ciudad:

Teléfono:  Movil:

E mail:

---

**Registrar Venta**

|                     |  |
|---------------------|--|
| Total Compra:       | 100000                                   |
| Impuesto:           | 16000                                    |
| Costos Adicionales: | <input style="width: 50%;" type="text"/> |
| Total Pago:         | 96000                                    |

Figura 59 Diseño de pantalla - Registrar venta oferta

| Fecha      | Cliente       | Estado     | Detalles | Eliminar | Confirmar despacho |
|------------|---------------|------------|----------|----------|--------------------|
| 20/02/2008 | Andrea Serna  | Pendiente  |          | ✖        |                    |
| 21/02/2008 | Rafaél Mendez | Despachada |          | ✖        |                    |

| Fecha      | Cliente       | Estado     | Detalles | Eliminar | Confirmar despacho |
|------------|---------------|------------|----------|----------|--------------------|
| 20/02/2008 | Andrea Serna  | Pendiente  |          | ✖        |                    |
| 21/02/2008 | Rafaél Mendez | Despachada |          | ✖        |                    |

Registrar Venta Oferta

**Información productos**

Vendedor:  Fecha:  ...

Buscar Oferta:

|                       |              |
|-----------------------|--------------|
| Combo Compaq + 2 USB  | \$ 1'590.000 |
| Combo Televisor + DVD | \$ 780.000   |

| Referencia            | Producto              | Cantidad |
|-----------------------|-----------------------|----------|
| h3025                 | Portatil Compaq h3025 | 1        |
| usb02                 | MEMORIA USB 1GB       | 2        |
| <b>Total a pagar:</b> |                       | 1590000  |

**Datos Cliente**

Identificación:

Nombre:

Dirección:  Ciudad:

Teléfono:  Movil:

E mail:

Registrar Venta

Combo Compaq + 2 USB

|                            |                      |
|----------------------------|----------------------|
| <b>Total Compra:</b>       | 100000               |
| <b>Costos Adicionales:</b> | <input type="text"/> |
| <b>Total Pago:</b>         | 96000                |

Figura 60 Diseño de pantalla - Costo de productos

| Referencia | Producto              | Costo Actual | Modificar Costo   |
|------------|-----------------------|--------------|---|
| h3025      | Portatil Compaq h3025 | 2500000      |  |
| m48op      | Impresora Epson m48op | 450000       |  |
| ui789      | Mouse USB Genius      | 22000        |  |
|            |                       |              |   |

**Modificar Costo de Productos**

Portatil Compaq h3025  
Referencia: h3025  
Valor actual: 2500000

Nuevo Costo:

Figura 61 Diseño de pantalla - Administración de noticias

| Título                     | Descripción Corta  | Fecha      | Hora  | Estado    | Modificar | Eliminar |
|----------------------------|--|------------|-------|-----------|-----------|----------|
| Tragedia en Birmania       | Imagen del estado en que quedó una calles de Yangón (antigua Rangún) este lunes tras el paso del ciclón Nargis. Se calculan 22 mil muertos en el país. | 2008-06-01 | 14:00 | Principal |           |          |
| Lluvia de cenizas en Chile | Así quedaron las calles de Futaleufú, afectada por una lluvia de cenizas provocada por la erupción del volcán Chaitén.                                 | 2008-06-02 | 16:00 | Otras     |           |          |

**Modificar Noticias**

**Información General**

Título:       Tipo:

Fecha:        Hora:

Descripción Corta:

Texto:

**Imágenes**

Buscar Archivo:

Imagen de la marca

Otra imagen

Figura 62 Diseño de pantalla - Administración de ofertas

| Descripción                  | Precio    | Fecha      | Cantidad Disponible | Imagen  | Modificar   | Cancelar  |
|------------------------------|-----------|------------|---------------------|---|---|---|
| Combo Portatil - memoria USB | \$ 85.000 | 2008-06-01 | 5                   |  |  |  |

**Modificar Oferta**

**Información General**

Descripción:  Fecha:  ...

Precio:  Cantidad Disponible:

Imagen:   

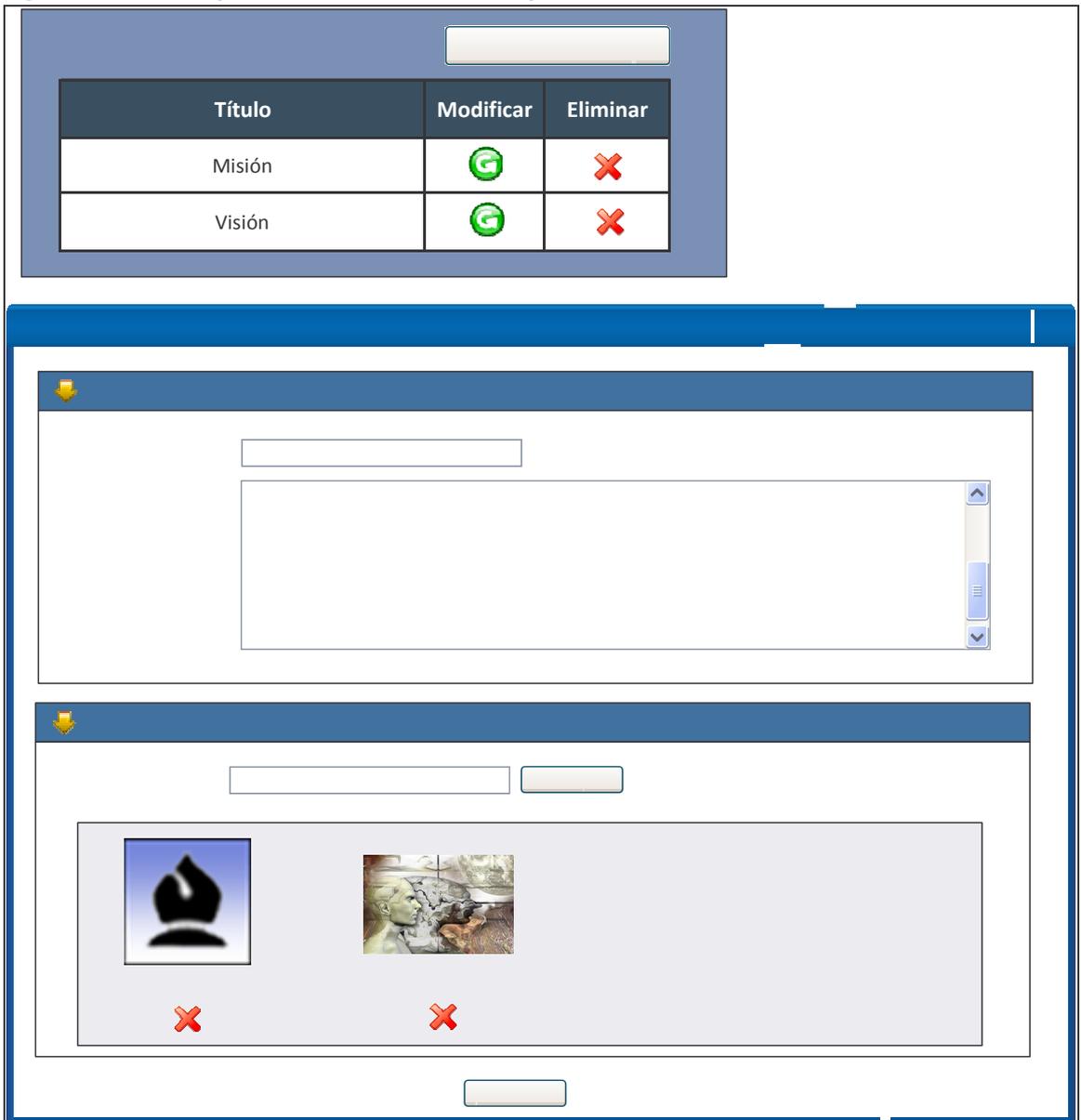
**Información productos**

Buscar Producto:

Cantidad:

| Descripción                    | Cantidad | Eliminar  |
|--------------------------------|----------|---|
| PC PORTATIL COMPAQ - REF 20154 | 2        |  |

Figura 63 Diseño de pantalla - Administración de opciones adicionales



Título:

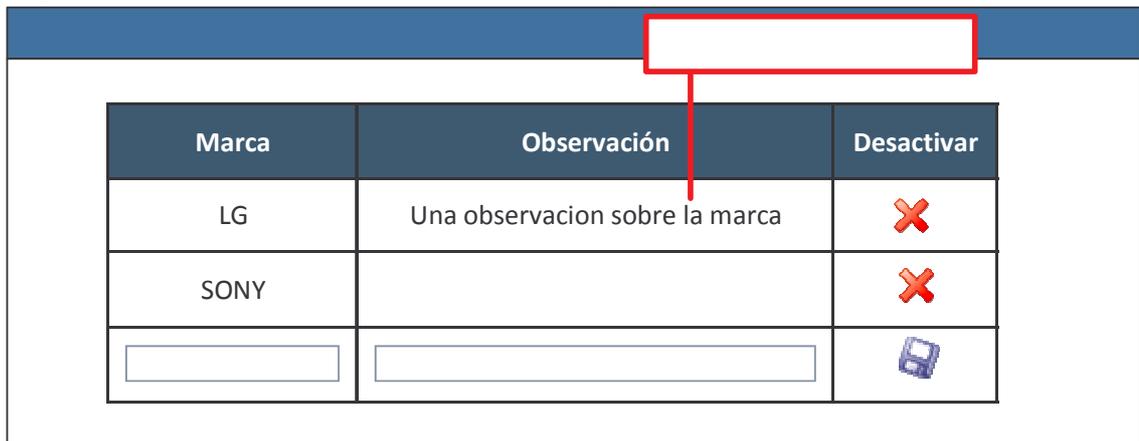
Texto:

Las opciones de administración de:

- Marcas
- Proveedores
- Clientes
- Vendedores
- Usuarios

Se realiza mediante el componente xTabularDataAdmin, por lo tanto tendrán formularios similares en los cuales variará las columnas, en la siguiente figura se muestra la interfaz básica del componente.

Figura 64 Diseño de pantalla - Interfaz básica de componente xTabularDataAdmin



| Marca                | Observación                    | Desactivar  |
|----------------------|--------------------------------|---|
| LG                   | Una observacion sobre la marca | ✘   |
| SONY                 |                                | ✘   |
| <input type="text"/> | <input type="text"/>           |  |

Las opciones de administración de:

- Ubicación geográfica
- Catálogos

Se realiza mediante el componente xMasterDetail, por lo tanto tendrán formularios similares en los cuales variará las columnas, en la siguiente figura se muestra la interfaz básica del componente.

Figura 65 Diseño de pantalla - Interfaz básica de componente xMasterDetail

**Gestión de catálogos**

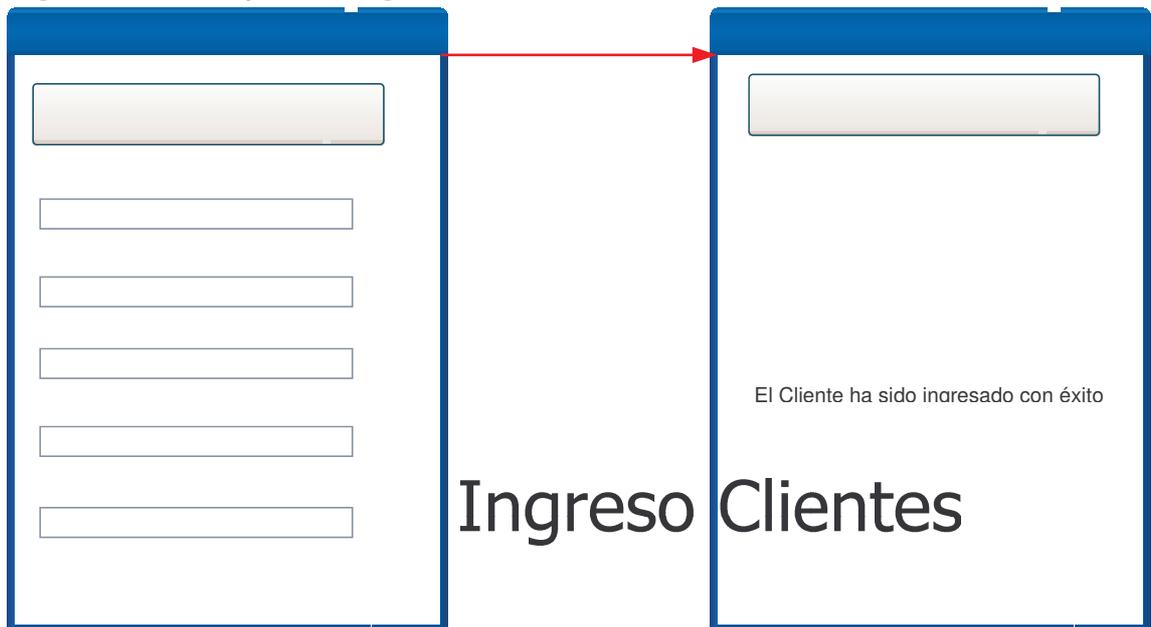
Celdas editables

| Nombre     | Descripción                                       | Desactivar |
|------------|---|------------|
| 📁 Deportes | Se muestra la información de productos deportivos | ✗          |
| Natación   | Muestra todos los productos relativos a natación  | ✗          |
| Futbol     | Muestra todos los productos relativos a Futbol    | ✗          |
| Tennis     | Muestra todos los productos relativos a Tennis    | ✗          |
| 📁 Ropa     | Encuentre toda lo relaiconado a prendas de vestir | ✗          |

Figura 66 Diseño de pantalla – Menú Móviles



Figura 67 Diseño de pantalla – Ingreso Clientes



## Ingreso Clientes

Nombres

Indetificacion

Movil

Telefono

Sincronización

Figura 68 Diseño de pantalla – Pre venta - Seleccionar Cliente

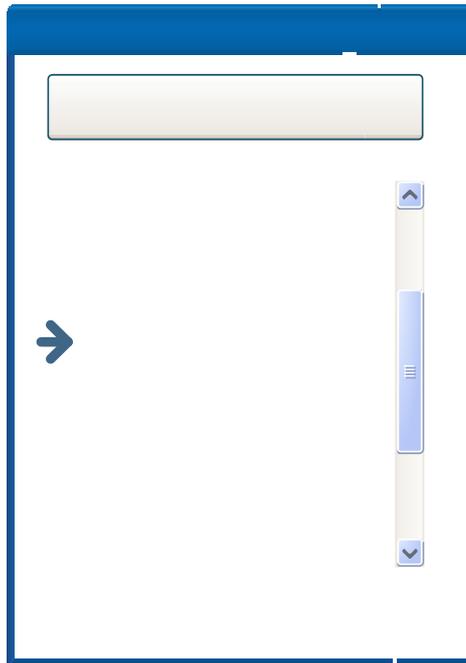
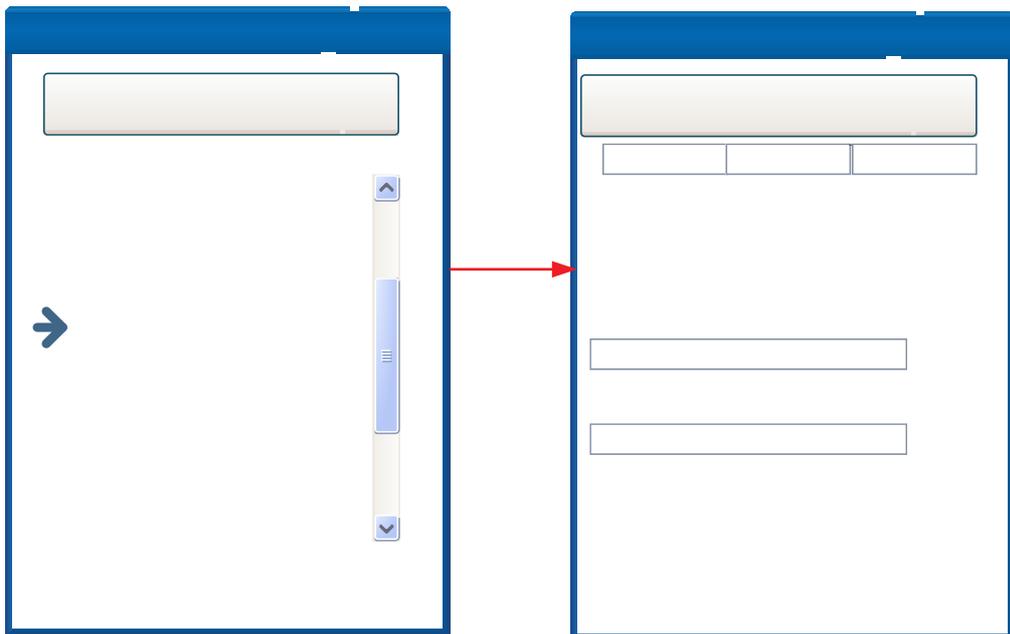


Figura 69 Diseño de pantalla – Pre venta - Ingreso Producto Pre venta



Alex V

Carlos

Maric  
59311

Maritza

Doris D

Sand

Figura 70 Diseño de pantalla – Pre venta

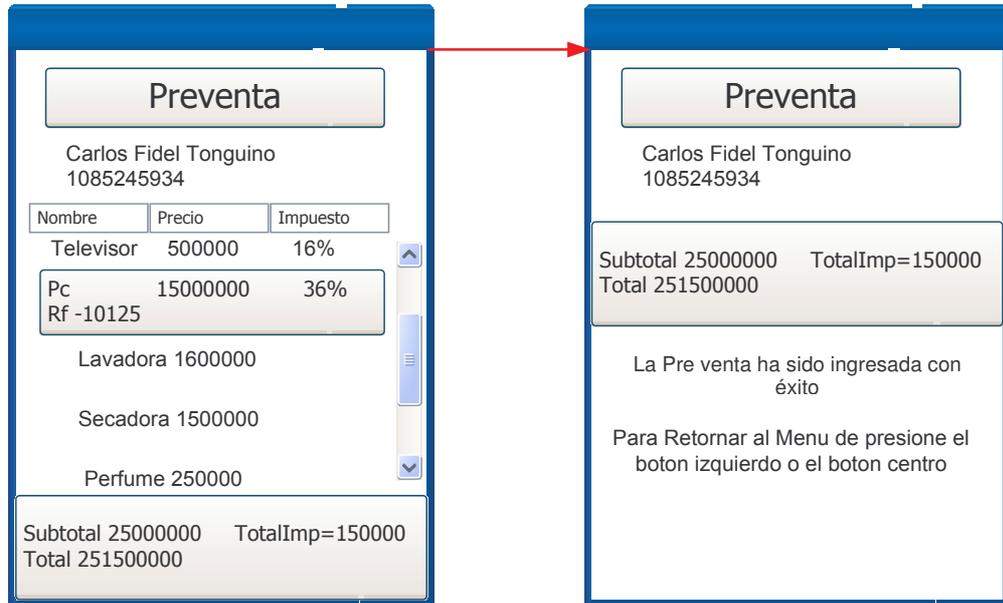
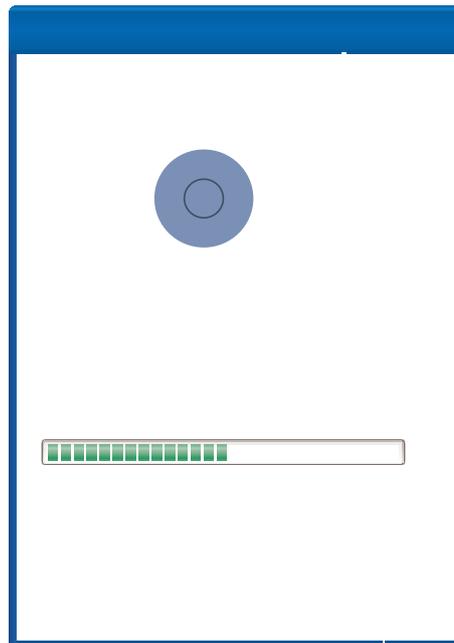


Figura 71 Diseño de pantalla – Pantalla de Carga



## 5. ARQUITECTURA DEL SISTEMA

VSHOP CMS está desarrollado el paradigma de programación orientada a objetos, se divide en tres módulos principales así:

- Administrador de la tienda
- Tienda generada
- Aplicación Móvil

En la siguiente figura 47 se muestra el modelo de arquitectura del sistema.

### 5.1. Administrador de la tienda.

Mediante este módulo el usuario podrá controlar la información de su tienda, tanto la parte comercial como los contenidos adicionales y el aspecto gráfico de la misma. Este módulo esta desarrollado mediante PHP 5 como lenguaje de programación y MySQL como sistema gestor de base de datos.



Figura 72 Arquitectura del sistema

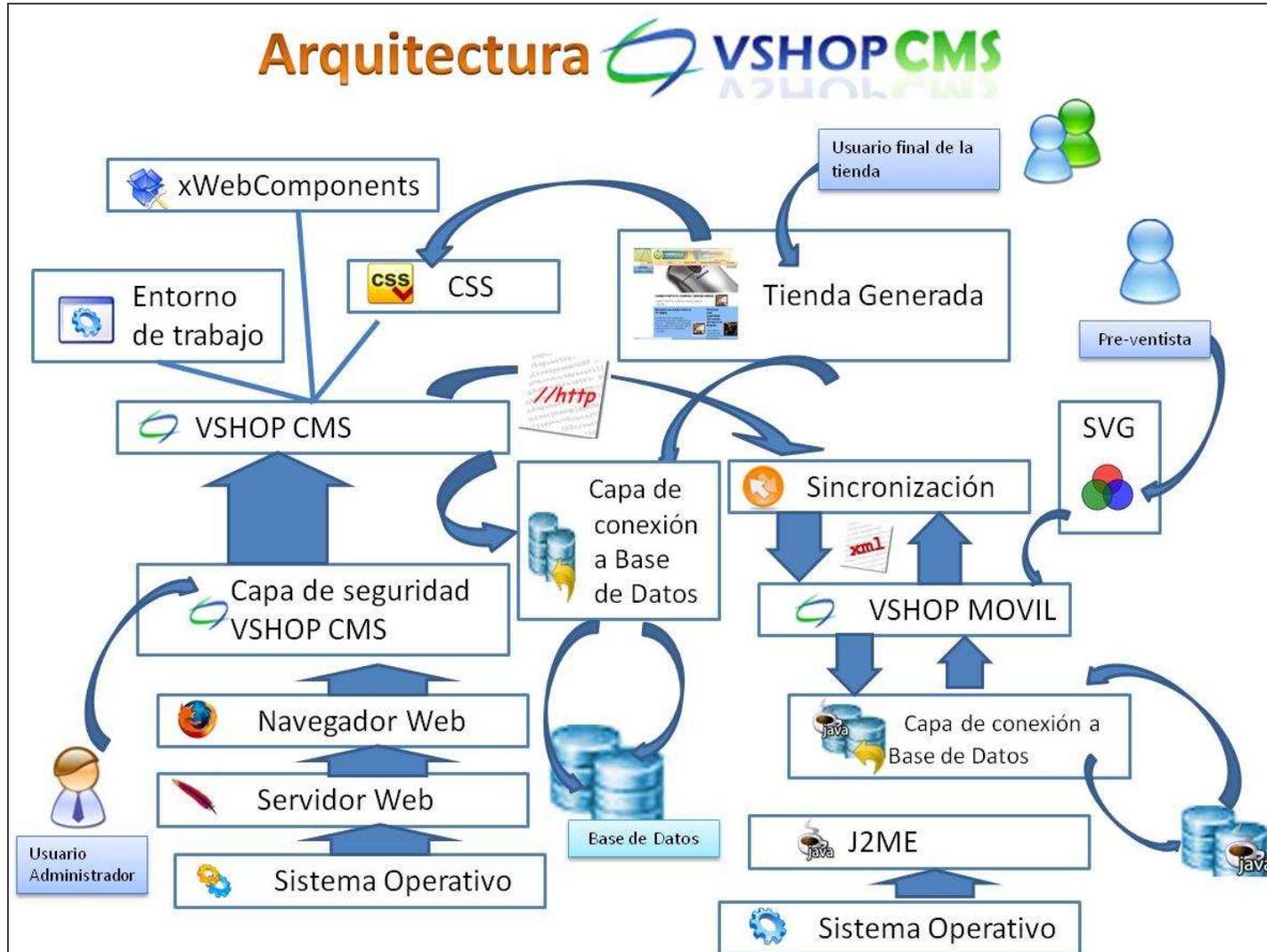
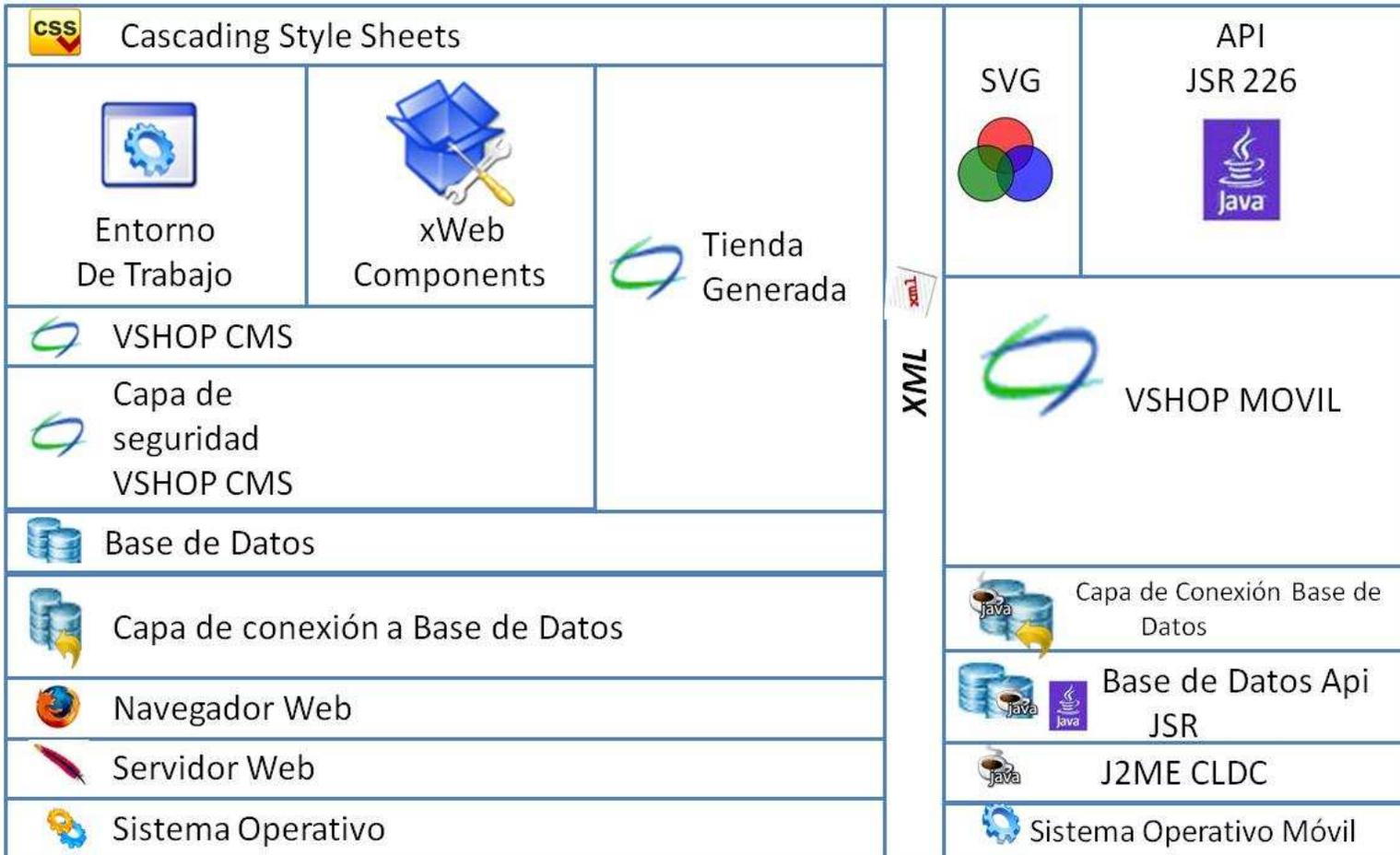


Figura 73 Arquitectura del sistema - Diagrama en bloques

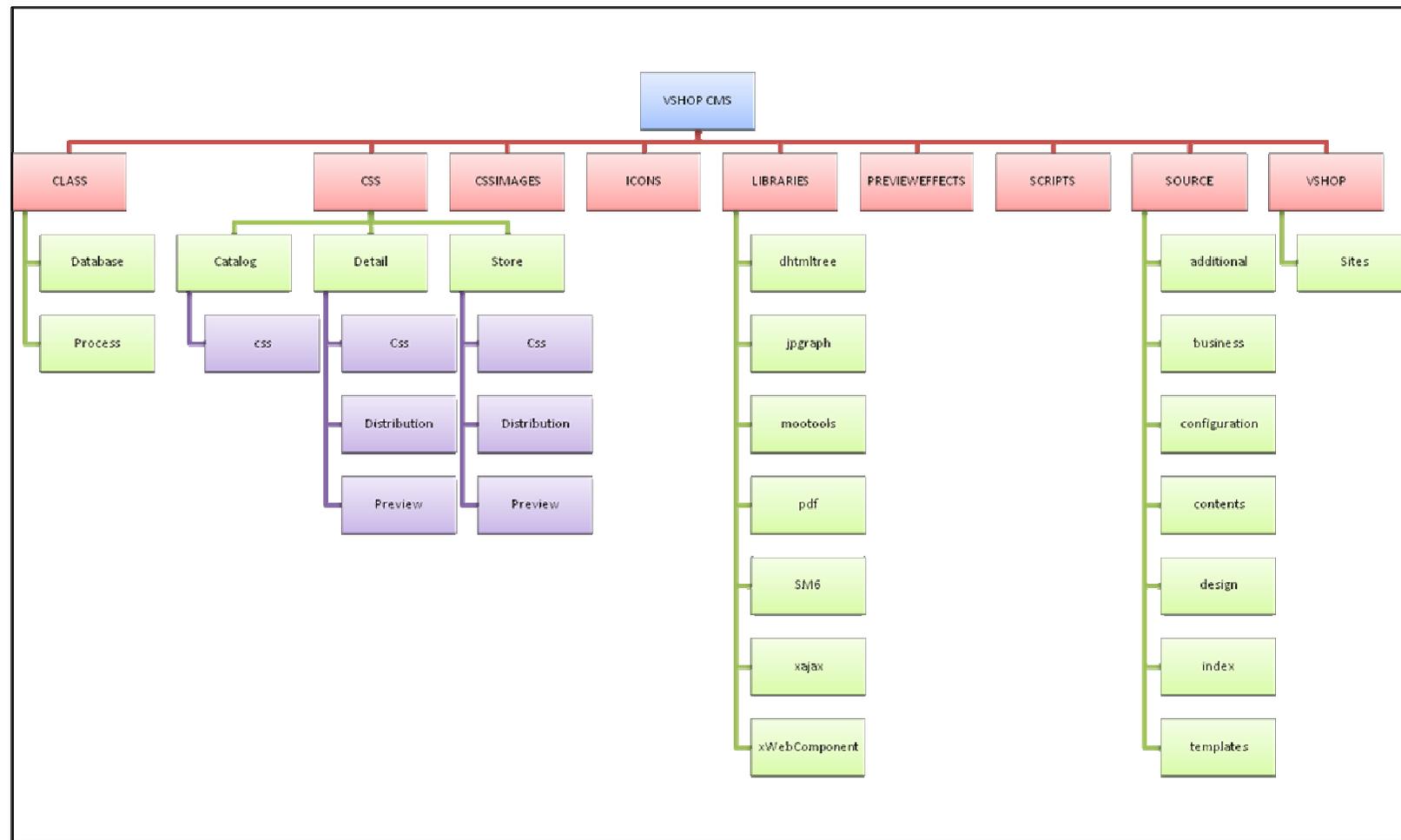




El sistema de archivos está estructurado como se muestra en la figura 55, a continuación se describe la funcionalidad de cada una de las carpetas:

- **CLASS:** clases del sistema, dentro de esta carpeta se encuentran las carpetas *process*, en la que se guardan las clases que intervienen en cada uno de los procesos del sistema, en estas clases se guardan las instrucciones SQL del sistema, y la carpeta *Database*, en donde están los archivos que se encargan de la capa de conexión con la Base de Datos y ejecución de instrucciones SQL.
- **CSS:** en esta carpeta se guardan los archivos CSS propios del sistema y los generados para las plantillas de tienda, catalogo y detalle producto.
- **CSSIMAGES:** En esta carpeta se guardan las imágenes relacionadas a los contenidos que se incluyen en cada una de las tiendas.
- **ICONS:** iconos que aparecen en la aplicación.
- **LIBRARIES:** aquí se encuentran los archivos correspondientes a las librerías utilizadas en el desarrollo del software, todas las librerías utilizadas son de uso libre, en esta carpeta se incluye la librería ***xWebComponents*** que contiene los componentes reutilizables desarrollados por el equipo de trabajo.
- **PREVIEWEFFECTS:** imágenes de vista previa de efectos utilizados por la aplicación.
- **SCRIPTS:** archivos Javascript que contienen funciones utilizados en el sistema.

Figura 74 Esquema de directorios VSHOP CMS



- **SOURCE:** en esta carpeta se encuentran los archivos de código fuente del sistema, cada uno de los archivos contienen funciones *xajax* que se incluyen en el archivo `source/index/index_i.php` el cual es el archivo principal del sistema. Cada una de las opciones cuenta con un archivo de interfaz el cual maneja los contenidos HTML y un archivo que maneja los procesos de lógica de funcionamiento de la opción.
- **VSHOP:** dentro de esta carpeta se encuentra la carpeta *Sites* en la cual se ubican los archivos generados de la tienda y los que intervienen en su funcionamiento.

**5.1.1. Sistema operativo.** [24] Un sistema operativo es un software de sistema, es decir, un conjunto de programas de computadora destinado a permitir una administración eficaz de sus recursos. Comienza a trabajar cuando se enciende el computador, y gestiona el hardware de la máquina desde los niveles más básicos, permitiendo también la interacción con el usuario. El proyecto VSHOP CMS funciona independiente del sistema operativo ya que esta desarrollado en entorno web, la cual es una gran ventaja que permite el trabajar en esta plataforma.

**5.1.2. Servidor web.** [25] Programa que utiliza el modelo cliente-servidor y el protocolo de transferencia de hipertexto para poner páginas web a disposición de los internautas. VSHOP CMS utiliza como servidor web Apache, pero se puede utilizar el servidor que el usuario seleccione.

**5.1.3. Seguridad.** El módulo cuenta con un esquema de seguridad inicia con autenticación de usuarios con nombre y contraseña, además permite que se creen nuevos usuarios por parte del administrador y que sea él quien defina el nivel de acceso que pueda tener entre: administrador, gestión comercial, administrador de contenidos, nivel intermedio. Cada uno de los niveles permite al usuario acceder a ciertas opciones lo cual permite al administrador dar acceso a otras personas a su sistema sin que eso signifique que tengan total acceso a la información de la tienda.

Siempre el usuario tendrá que pasar por la autenticación antes de poder acceder al sistema, esto se controla mediante manejo de variables de sesión. Si el usuario intenta acceder a cualquiera de los archivos que maneja el sistema mediante su URL inmediatamente será re direccionado a una página que indique el acceso restringido y lo vincule a la página de autenticación.

**5.1.4. Entorno de trabajo.** VSHOP CMS cuenta con un entorno de trabajo desarrollado para ayudar a la navegación del usuario y permitir que acceda fácilmente a las opciones que desee, cabe relacionar que el entorno de trabajo no fue desarrollado específicamente para este proyecto sino con propósitos de reutilización.

Figura 75 Entorno de trabajo VSHOP CMS



El sistema cuenta con un menú plegable, las opciones que aquí aparecen se extraen de base de datos y se presentan según el tipo de usuario que esté registrado en el sistema. Las opciones activas se manejan mediante pestañas, esto permite que el usuario pueda tener activa más de una tarea mientras trabaja en el sistema. Cuando el usuario ingresa a una opción la información respectiva se mostrará en el área de trabajo ubicada en la parte central del sitio.

Para formularios adicionales se usan cuadros modales como el de la siguiente figura.

Figura 76 Cuadro modal para formularios

The image shows a modal window titled 'Nueva Tienda' with a close button in the top right corner. The form contains the following fields:

- Nombre:
- Responsable:
- País:  (dropdown menu)
- Dirección:
- Teléfono 2:
- Servicio al cliente:
- Logo:  No se h... archivo
- Palabras clave:
- Empresa:
- E-mail:
- Ciudad:  (dropdown menu)
- Teléfono Ppal:
- Teléfono 3:
- Prefijo Base de Datos:

At the bottom center of the modal is a button labeled 'Crear tienda'.

Para mostrar mensajes de confirmación de procesos el sistema utiliza un cuadro modal como el de la siguiente figura.

Figura 77 Modal para alertas

The image shows a modal window titled 'Seleccionar Tienda' with a close button in the top right corner. The text inside the modal reads:

Se ha modificado la tienda del sistema.  
La tienda actual del sistema es **Patch Corp**

At the bottom center of the modal is a button labeled 'Aceptar'.

Para mostrar acciones que necesitan confirmación del usuario se usan cuadros modales como el siguiente:

Figura 78 Modal de confirmación

The image shows a modal window titled 'Editar información Tienda.' with a close button in the top right corner. The text inside the modal reads:

Seguro desea Desactiva la tienda **MiTienda** ?

At the bottom center of the modal are two buttons: 'Aceptar' (highlighted in green) and 'Cancelar'.

El uso de estas herramientas permite que la aplicación sea mucho más familiar para el usuario ya que encontrará similitud en todos los procesos que realicen en

el sistema, y a nivel de programación el uso de funciones genéricas agiliza en gran medida los procesos de desarrollo.

**5.1.5. xWebComponentes.** En VSHOP CMS se desarrollaron componentes que permiten simplificar el desarrollo de procesos que se utilizan generalmente, fueron diseñados no solo para esta aplicación sino para el uso en aplicaciones php.

**5.1.5.1. XDataGridComponent.** Componente que permite generar listados en pantalla a partir de consultas MySQL, cuenta con características como:

- Ordenamiento del listado por columnas.
- Paginado de resultados.
- Búsqueda de registros según criterios del usuario.
- Adición de columnas de opciones adicionales e imágenes.
- Generación de reportes en formato PDF, DOC y XLS.

A continuación se describen las características de la clase xDataGridComponent mediante la cual se genera el componente:

**Tabla 3. Atributos principales de xDataGridComponents**

| <b>Atributo</b>                       | <b>Descripción</b>   |
|---------------------------------------|--|
| Título                                | Título del informe   |
| Columnas                              | Columnas que aparecerán en el informe, son objetos de tipo xDataColumn que se describe a continuación. |
| Consulta inicial                      | Consulta SQL que genera los registros del informe.   |
| Índice del campo de orden de la tabla | Campo mediante el cual se ordenará inicialmente el informe.  |
| Orientación                           | Ascendente o descendente.  |
| Número de registros                   | Numero de registros que se muestran por página.  |
| Documentos                            | Permite elegir si se muestra las opciones de generación de documentos en formato PDF, DOC y XLS.       |
| Búsqueda                              | Permite elegir si se muestra las opciones de búsqueda.   |

Características de la clase xDataColumn mediante la cual se elige las características de cada una de las columnas.

**Tabla 4. Atributos principales de xDataGridColumn**

| Atributo   | Descripción   |
|------------|---|
| Tipo       | Tipo de columna, puede ser uno de los siguientes tipos: <ul style="list-style-type: none"> <li>• Show: columna que muestra solo un dato</li> <li>• Action: columna que permite realizar una acción.</li> <li>• Empty: Columna vacía.</li> <li>• Image: Columna que tiene una imagen.</li> </ul> |
| Título     | Título de la columna  |
| Tamaño     | Ancho de la columna   |
| Alineación | Alineación de la columna: izquierda, derecha, centro.   |

A continuación se muestra un código ejemplo que permite el uso de este componente (este código esta dentro de una función xajax).

```

// CREAMOS EL OBJETO, PARA CREARLO LE ENVIAMOS UN OBJETO DE CONEXION A BASE DE DATOS
$xDataGrid=new xDataGrid($DBC);
// DEFINIMOS LAS COLUMNAS
$columns[0]=new xDataColumn('image',utf8_decode('Logo'),'170px','center'); // TIPO IMAGEN
$columns[1]=new xDataColumn('show',utf8_decode('Tienda'),'','center'); // TIPO SHOW
$columns[2]=new xDataColumn('show',utf8_decode('Empresa'),'','center'); // TIPO SHOW
$columns[3]=new
xDataColumn('action',utf8_decode('Modificar'),'','center','','../icons/modify.png',"xajax_cst_modifyStore()"); // TIPO ACCION
$columns[4]=new xDataColumn('action',utf8_decode('Desactivar'),'','center','','../icons/delete.png',"xajax_cst_offStore()"); //
TIPO ACCION
// CONSULTA INICIAL
$xDataGrid->initQuery="SELECT t.id_tie, IF(t.log_tie!=",t.log_tie, '../icons/nologo2.png'), t.nom_tie, t.emp_tie FROM tienda
t WHERE t.est_tie="a";
// INFORMACIÓN DE CADA CAMPO
$columns[0]->field="IF(t.log_tie!=",t.log_tie, '../icons/nologo2.png)";
$columns[1]->field="t.nom_tie";
$columns[2]->field="t.emp_tie";
// MUSTRAMOS LA OPCION DE BUSQUEDA
$xDataGrid->display_search='true';
// TIPO DE GRID : show, unique, multiple
$xDataGrid->type='show';
// ALINEACIÓN
$xDataGrid->align='center';
// ASIGNAMOS LAS COLUMNAS AL OBJETO
$xDataGrid->columns=$columns;
// ANCHO DE LA TABLA
$xDataGrid->size="800px";
// CAMPO QUE ESPECIFICA EL ORDEN
$xDataGrid->fieldOrder=3;
// ORIENTACIÓN DEL ORDENAMIENTO ASCENDENTE
$xDataGrid->orientation='ASC';

```

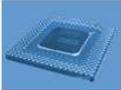
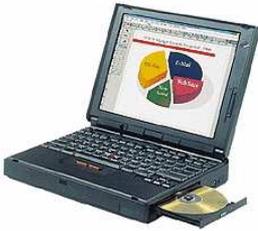
```

// FILAS POR PAGINA
$xDataGrid->sizeRows=10;
$xDataGrid->initLimit=0;
// IDENTIFICADOR DEL GRID
$xDataGrid->id='cst_xgrid';
// NO MOSTRAR OPCION DE GENERACION DE DOCUMENTOS
$xDataGrid->docs='hidden';
// SUBIMOS EL OBJETO A SESION
$_SESSION['cst_xgrid']=serialize($xDataGrid);
// ASIGNAMOS LA GENERACIÓN DEL CONTENIDO A UN DIV SELECCIONADO
$res->addAssign("cst_xgrid","innerHTML",$xDataGrid->getHTML());

```

El resultado se muestra en la siguiente figura:

**Figura 79 Ejemplo funcionamiento de xDataGridComponent**

| Buscar por: Tienda  |            |                          |   |   |
|---|------------|--------------------------|---|---|
| Logo  | Tienda     | Empresa                  | Modificar   | Desactivar  |
|    | CDG VSHOP  | Cronos Development Group |    |    |
|   | MITienda   | CDG SYSTEMS              |  |  |
|  | Patch Corp |                          |  |  |

Total de registros: 3

**5.1.5.2. xTabularDataAdmin.** Componente que extiende de xDataGridComponent, este componente permite hacer una administración tabular de la información de una tabla. Permite crear nuevos registros, editar o desactivar registros existentes.

A continuación se muestra un código ejemplo que permite el uso de este componente (este código esta dentro de una función xajax).

```

// CREAMOS EL OBJETO, ENVÍO CONEXION A BASE DE DATOS, NOMBRE DE LA TABLA A ADMINISTRAR Y LLAVE PRIMARIA
$xTabular=new xTabularDataAdmin($DBC, 'proveedor', 'id_pro');
// SE DEFINE LA CONSULTA INICIAL

```

```

$хTabular->initQuery="SELECT p.id_pro, p.des_pro, p.ase_pro, p.dir_pro, p.tel_pro FROM proveedor p WHERE
p.est_pro='a'";
// FILAS POR PAGINA
$хTabular->sizeRows=10;
// ANCHO DE LA TABLA
$хTabular->size=600;
// INDICE DEL CAMPO DE ORDENAMIENTO INICIAL
$хTabular->fieldOrder=2;
// TITULO DE LA TABLA
$хTabular->title='Administraci&oacute;n de Proveedores de la tienda.';
// DEFINIMOS LA PRIMERA COLUMNA DE TIPO TEXTO, CON TITULO NOMBRE, NOMBRE DEL CAMPO EN BASE DE
DATOS DES_PRO
$columns[0]=new хTabularColumnAdmin("text",'Nombre','des_pro','','texto');
$columns[0]->field="p.des_pro";
// DEFINIMOS LA SEGUNDA COLUMNA
$columns[1]=new хTabularColumnAdmin("text",'Asesor','ase_pro','','texto');
$columns[1]->field="p.ase_pro";
// DEFINIMOS LA TERCERA COLUMNA
$columns[2]=new хTabularColumnAdmin("text",'Direcci&oacute;n','dir_pro','','texto');
$columns[2]->field="p.dir_pro";
// DEFINIMOS LA CUARTA COLUMNA
$columns[3]=new хTabularColumnAdmin("text",'Tel&eacute;fono','tel_pro','','texto');
$columns[3]->field="p.tel_pro";
// ASIGNAMOS LAS COLUMNAS AL OBJETO
$хTabular->columns=$columns;
// CAMPO IDENTIFICADOR DEL OBJETO
$id='apr_xTabularData';
$хTabular->id=$id;
// INDICAMOS QUE NO SE MUESTRE DOCUMENTOS
$хTabular->docs='none';
// INDICAMOS QUE SE MUESTRE LAS OPCIONES DE BUSQUEDA
$хTabular->display_search='true';
// SUBIMOS EL OBJETO A SESION
$_SESSION["$id"]=serialize($хTabular); // SERIALIZO EL OBJETO
// ASIGNAMOS EL CONTENIDO GENERADO A UN DIV ELEGIDO
$хres->addAssign("apr_xTabularData", "innerHTML", $хTabular->getHTML());

```

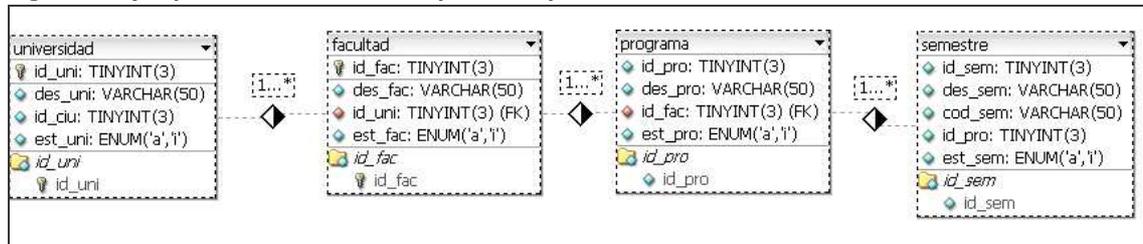
El resultado se muestra en la siguiente figura.

**Figura 80** Ejemplo funcionamiento xTabularDataAdmin

| Administración de Proveedores de la tienda. |                      |                      |                      |       |
|---|----------------------|----------------------|----------------------|-------|
| Buscar por: Nombre <input type="text"/>     |                      |                      |                      |       |
| Nombre                                      | Asesor               | Dirección            | Teléfono             | Elim. |
| ALMACENES EXITO                             | JUAN PEREZ           | CALLE 20 25-87       | 7215888              |       |
| <input type="text"/>                        | <input type="text"/> | <input type="text"/> | <input type="text"/> |       |

**5.1.5.3. xMasterDetail.** Componente que extiende de xTabularDataAdmin, permite realizar administración de tablas relacionales, como las del siguiente ejemplo:

**Figura 81 Ejemplo Tablas relacionales para la implementación de xMasterDetail**



En el ejemplo anterior se muestra una estructura relacional en donde la primera tabla es **Universidad**, cada universidad tiene 1 o varias **Facultades**, una facultad puede tener 1 o varios **Programas**, un programa puede tener 1 o varios **Semestres**.

A continuación se muestra un código ejemplo que permite el uso de este componente (este código está dentro de una función xajax).

```

/* DEFINO OBJETO: SEMESTRE */
// Se envía el nombre de la tabla y la llave primaria.
$xSemestreGrid=new xMasterDetail($DBC, 'semestre', 'id_sem');
// Defino la consulta inicial
$xSemestreGrid->initQuery="SELECT s.id_sem, s.des_sem FROM semestre s WHERE s.est_sem='a'";

// Cantidad de registros por pagina
$xSemestreGrid->sizeRows=10;
// Ancho de la tabla
$xSemestreGrid->size=600;
// Índice del campo de ordenamiento inicial
$xSemestreGrid->fieldOrder=2;
// Creamos las columnas
$columns[0]=new xTabularColumnAdmin("text",'Semestre','des_sem','','texto');
$columns[0]->field="s.des_sem";
// Asignamos las columnas al objeto
$xSemestreGrid->columns=$columns;
$id='xSemestreGrid';
// Definimos el identificador
$xSemestreGrid->id=$id;
// Definimos quien es el objeto principal o maestro
$xSemestreGrid->master='xMasterDetail';
// Definimos cual es el objeto padre
$xSemestreGrid->xparent='xProgramaGrid';
// Se serializa el objeto y se sube a sesión
$_SESSION["$id"]=serialize($xSemestreGrid); // SERIALIZO EL OBJETO

/* DEFINO OBJETO: PROGRAMA */
$xProgramaGrid=new xMasterDetail($DBC, 'programa', 'id_pro');
$xProgramaGrid->initQuery="SELECT p.id_pro, p.des_pro FROM programa p WHERE p.est_pro='a'";

```

```

$xProgramaGrid->sizeRows=10;
$xProgramaGrid->size=600;
$xProgramaGrid->fieldOrder=2;
$columns[0]=new xTabularColumnAdmin("text",'Programa','des_pro','','texto');
$columns[0]->field="p.des_pro";
$xProgramaGrid->columns=$columns;
$id='xProgramaGrid';
$xProgramaGrid->detail=$xSemestreGrid;
$xProgramaGrid->id=$id;
$xProgramaGrid->docs='none';
$xProgramaGrid->master='xMasterDetail';
$xProgramaGrid->xparent='xFacultadGrid';
// $xProgramaGrid->display_search='true';
$_SESSION["$id"]=serialize($xProgramaGrid); // SERIALIZO EL OBJETO

/* DEFINO OBJETO: FACULTAD */
$xFacultadGrid=new xMasterDetail($DBC, 'facultad', 'id_fac');
$xFacultadGrid->initQuery="SELECT f.id_fac, f.des_fac FROM facultad f WHERE f.est_fac='a'";
$xFacultadGrid->initLimit=0;
$xFacultadGrid->sizeRows=10;
$xFacultadGrid->size=600;
$xFacultadGrid->fieldOrder=2;
$columns[0]=new xTabularColumnAdmin("text",'Facultad','des_fac','','texto');
$columns[0]->field="f.des_fac";
$xFacultadGrid->columns=$columns;
$id='xFacultadGrid';
$xFacultadGrid->detail=$xProgramaGrid;
$xFacultadGrid->id=$id;
$xFacultadGrid->docs='none';
$xFacultadGrid->master='xMasterDetail';
$xFacultadGrid->xparent='xMasterDetail';
// $xFacultadGrid->display_search='true';
$_SESSION["$id"]=serialize($xFacultadGrid); // SERIALIZO EL OBJETO

/** DEFINO OBJETO: UNIVERSIDAD */
$xMasterGrid=new xMasterDetail($DBC, 'universidad', 'id_uni');
$xMasterGrid->initQuery="SELECT u.id_uni, u.des_uni FROM universidad u WHERE u.est_uni='a'";
$xMasterGrid->initLimit=0;
$xMasterGrid->sizeRows=10;
$xMasterGrid->size=600;
$xMasterGrid->fieldOrder=2;
$xMasterGrid->title='Administraci&oacute;n de Universidades';
$columns[0]=new xTabularColumnAdmin("text",'Descripci&oacute;n','des_uni','','texto');
$columns[0]->field="u.des_uni";
$xMasterGrid->columns=$columns;
$id='xMasterDetail';
$xMasterGrid->detail=$xFacultadGrid;
$xMasterGrid->id=$id;
$xMasterGrid->docs='none';
$xMasterGrid->display_search='true';
$xMasterGrid->master='xMasterDetail';
$_SESSION["$id"]=serialize($xMasterGrid); // SERIALIZO EL OBJETO
// Retorno el codigo html correspondiente
$res->addAssign("xMasterDetail", "innerHTML", $xMasterGrid->getHTML());

```

Figura 82 Ejemplo funcionamiento xMasterDetail

| Administración de Universidades              |       |
|--|-------|
| Buscar por: Descripción <input type="text"/> |       |
| Descripción                                  | Elim. |
| + Universidad del Cauca                      |       |
| - Universidad de Nariño                      |       |
| Facultad                                     | Elim. |
| + Artes                                      |       |
| + Derecho                                    |       |
| + Economía                                   |       |
| - Ingeniería                                 |       |
| Programa                                     | Elim. |
| + Civil                                      |       |
| + Industrial                                 |       |
| - Sistemas                                   |       |
| Semestre                                     | Elim. |
| 1  |       |
| 3  |       |
| <input type="text"/>                         |       |

**5.1.5.4. xCreateCSS.** El objetivo de este componente es construir formularios que permitan establecer características CSS y generar el código CSS correspondiente a las características establecidas.

Un ejemplo del código utilizado para generar el formulario es el siguiente (este código esta dentro de una función xajax):

```
// Creamos el objeto
$xml1=new xCreateCSS('vshop_encabezado');
// Asignamos el codigo que genera el objeto a un div
$xml->addAssign("cssEncabezado","innerHTML", $xml1 -> getHTML(true,true,true,true,true,true,false));
```

El formulario generado es el siguiente:

Figura 83 Ejemplo funcionamiento de xCreateCSS

|                    |                              |   |   |                          |
|--------------------|------------------------------|---|---|--------------------------|
| <b>Posición:</b>   | X:                           | <input type="text"/>                              | Y:  | <input type="text"/>     |
| <b>Tamaño:</b>     | Ancho:                       | <input type="text"/>                              | Alto:   | <input type="text"/>     |
| <b>Fondo:</b>      | Color:                       | <input type="text"/>                              |  |                          |
|                    | Imagen:                      | <input type="text"/>                              |   |                          |
|                    | Repetir en X:                | <input type="checkbox"/>                          | Repetir en Y:   | <input type="checkbox"/> |
|                    | Ancho Mínimo:                | <input type="text"/>                              | Alto Mínimo:  | <input type="text"/>     |
| <b>Fuente:</b>     | Familia:                     | Arial, Helvetica, sans-serif <input type="text"/> |   |                          |
|                    | Tamaño:                      | 8 <input type="text"/>                            |   |                          |
|                    | Negrita:                     | <input type="checkbox"/>                          | Cursiva:  | <input type="checkbox"/> |
|                    | Color:                       | <input type="text"/>                              |  |                          |
| <b>Alineación:</b> | Derecha <input type="text"/> |   |   |                          |
| <b>Margenes:</b>   | lzq:                         | <input type="text"/>                              | Der:  | <input type="text"/>     |
|                    | Sup:                         | <input type="text"/>                              | Inf:  | <input type="text"/>     |
| <b>Relleno:</b>    | lzq:                         | <input type="text"/>                              | Der:  | <input type="text"/>     |
|                    | Sup:                         | <input type="text"/>                              | Inf:  | <input type="text"/>     |

Un código ejemplo que genera el archivo CSS:

```
// $array es el array de nombres de objetos xCreateCSS
if($array==""){
$array=$_POST['ete_x_array'];
}
$array=explode(',',$array);
// Recorro el array
foreach($array as $object){
unset($xcss);
// Creo cada objeto
$xcss=new xCreateCSS($object);
if($_POST['ete_x_array']!=""){
$xcss->abr=substr($object,4);
}else{
$xcss->abr=$object;
}
// Asigno las propiedades que me envia el formulario
$xcss->orientation=$_POST[$object.'_xcr_orientation'];
$xcss->align=$_POST[$object.'_xcr_align'];
$xcss->x=$_POST[$object.'_xcr_x'];
$xcss->y=$_POST[$object.'_xcr_y'];
$xcss->width=$_POST[$object.'_xcr_width'];
$xcss->height=$_POST[$object.'_xcr_height'];
$xcss->fontFamily=$_POST[$object.'_xcr_fontFamily'];
$xcss->fontFamily=str_replace("\",\"", $xcss->fontFamily);
$xcss->fontFamily=str_replace("\'", "'", $xcss->fontFamily);
$xcss->sizeFont=$_POST[$object.'_xcr_sizeFont'];
$xcss->colorFont=$_POST[$object.'_xcr_colorFont'];
$xcss->bold=$_POST[$object.'_xcr_bold'];
```

```

$xcss->italic=$_POST[$object.'_xcr_italic'];
$xcss->marginLeft=$_POST[$object.'_xcr_marginLeft'];
$xcss->marginRight=$_POST[$object.'_xcr_marginRight'];
$xcss->marginTop=$_POST[$object.'_xcr_marginTop'];
$xcss->marginBottom=$_POST[$object.'_xcr_marginBottom'];
$xcss->paddingLeft=$_POST[$object.'_xcr_paddingLeft'];
$xcss->paddingRight=$_POST[$object.'_xcr_paddingRight'];
$xcss->paddingTop=$_POST[$object.'_xcr_paddingTop'];
$xcss->paddingBottom=$_POST[$object.'_xcr_paddingBottom'];
$xcss->bgImage="../../../cssimages/"$_POST[$object.'_xcr_bgImage'];
$xcss->bgColor=$_POST[$object.'_xcr_bgColor'];
$xcss->bgRepeat=$_POST[$object.'_xcr_bgRepeat'];
$xcss->bgWidth=$_POST[$object.'_xcr_bgWidth'];
$xcss->bgHeight=$_POST[$object.'_xcr_bgHeight'];
// Genero el codigo de tipo class
$code.=$xcss->getCode($type='class');
}
// nombre del archivo
$nombre=$_POST['nts_nom_pla'];
if($nombre==""){
$nombre=$_POST['ete_nom_pla2'];
}
// Creo el archivox
file_put_contents ("../../../css/store/css/$nombre.css", $code);

```

El archivo generado tiene el siguiente contenido:

```

.vshop_encabezado{
    left:0;
    top:0;
    position:absolute;
    width:800;
    height:100;
    font-family:Arial, Helvetica, sans-serif;
    font-size:8;
    text-align:right;
    background-image:url(../../../cssimages/fondo.jpg);
    background-color:#4b74a3;
}
.vshop_menuCatalogos{
    left:0;
    top:130;
    position:absolute;
    width:150;
    background-image:url(../../../cssimages/);
}
.vshop_btn_catalogo{
    width:145;
    height:25;
    font-family:Arial, Helvetica, sans-serif;
    font-size:12;
    font-weight:bold;
    text-align:center;
    padding-right:5px;
    padding-top:5px;
}

```

```

        background-image:url(../../cssimages/tmp001_btnCatalogo.png);
    }
    .vshop_btn_catalogo_hover{
        width:145;
        height:25;
        font-family:Arial, Helvetica, sans-serif;
        font-size:12;
        font-weight:bold;
        text-align:right;
        padding-right:5px;
        padding-top:5px;
        background-image:url(../../cssimages/tmp001_btnCatalogo_h.png);
    }
    .vshop_menuAdicionales{
        left:50;
        top:101;
        position:absolute;
        width:750;
        height:28;
        background-image:url(../../cssimages/);
        background-color:#fcf9dc;
    }
    .vshop_btn_adicional{
        float:left;
        width:150;
        height:27;
        font-family:Arial, Helvetica, sans-serif;
        font-size:12;
        font-weight:bold;
        text-align:center;
        background-image:url(../../cssimages/tmp001_btnOpciones.png);
    }
    .vshop_btn_adicional_hover{
        float:left;
        width:150;
        height:27;
        font-family:Arial, Helvetica, sans-serif;
        font-size:14;
        font-weight:bold;
        text-align:center;
        background-image:url(../../cssimages/tmp001_btnOpciones_h.png);
    }
    .vshop_principales{
        left:150;
        top:130;
        position:absolute;
        width:650;
        height:200;
        font-family:Arial, Helvetica, sans-serif;
        font-size:14;
        text-align:center;
        background-image:url(../../cssimages/);
        background-color:#FFFFFF;
    }
    .vshop_ofertas{
        left:150;
        top:331;
        position:absolute;
        width:650;
        height:100;
        font-family:Arial, Helvetica, sans-serif;
        font-size:12;
        text-align:center;

```

```
background-image:url(../cssimages/);
background-color:#FFFFFF;
}
.vshop_compras{
left:150;
top:581;
position:absolute;
width:650;
height:50;
font-family:Arial, Helvetica, sans-serif;
font-size:12;
text-align:center;
background-image:url(../cssimages/);
background-color:#FFFFFF;
}
.vshop_noticias{
left:150;
top:431;
position:absolute;
width:650;
height:300;
font-family:Arial, Helvetica, sans-serif;
font-size:12;
text-align:center;
background-image:url(../cssimages/);
background-color:#8cbffa;
}
}
```

**5.1.6. CSS.** El control de apariencia de la tienda y del sistema como tal se maneja mediante CSS, mediante el cual se permite establecer las características de fondo, pestañas, botones, cuadros modales, color de texto, imágenes de fondo, etc. [26] Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación. Por ejemplo, el elemento de HTML <H1> indica que un bloque de texto es un encabezamiento y que es más importante que un bloque etiquetado como <H2>. Versiones más antiguas de HTML permitían atributos extra dentro de la etiqueta abierta para darle formato (como el color o el tamaño de fuente). No obstante, cada etiqueta <H1> debía disponer de la información si se deseaba un diseño consistente para una página, y además, una persona que lea esa página con un navegador pierde totalmente el control sobre la visualización del texto.

**5.1.7. Capa de acceso a datos.** Para el acceso a datos el sistema cuenta con un conjunto de clases que permiten que esta capa se separe de las otras

características del sistema, mediante la clases DBConnection el sistema ejecuta todas las instrucciones SQL, haciendo que la conexión y contacto con la base de datos sea transparente para el usuario, esto además permite si se desea, modificar la base de datos que actualmente en VSHOP CMS es MySQL sin que el sistema sufra mayores traumatismos.

## **5.2. TIENDA GENERADA**

En el momento en que se crea una tienda en el sistema se generan las siguientes acciones:

- Creación de tablas en base de datos: los productos, marcas, proveedores, ofertas, noticias, entre otros, son propios de cada tienda, para mayor comodidad en acceso y rapidez el sistema crea tablas propias de cada tienda en la base de datos.
- Creación de Página inicial de la tienda: el sistema genera un archivo con el nombre de la tienda dentro de la carpeta /vshop/sites que será la página inicial de la tienda, a esta página se deberá re direccionar el dominio de la página de la tienda.
- Creación de carpetas para imágenes: se crean carpetas para guardar las imágenes de Noticias, Secciones y Ofertas de cada tienda dentro de la carpeta CSSImages.

## **5.3. APLICACIÓN MOVIL**

La aplicación de móvil que integra VSHOP–CMS. es una aplicación desarrollada para los Pre-ventista pertenecientes a la empresa pueden realizar pre ventas a los clientes, es una aplicación desarrollada sobre la plataforma JME (Java Micro Edition) sobre la Configuración CLDC y el perfil MIDP. Es una aplicación generada para que interactúe con la aplicación de VSHOP-CMS web para realizar operaciones conjuntas de administración de inventarios, administración de clientes, administración de ventas, así la aplicación móvil es una herramienta de soporte para la aplicación web, para realizar pre ventas sobre dispositivos móviles, y luego lograr que estas pre ventas sean registradas en la tienda.

El equipo en el que se aloje la aplicación móvil debe soportar como mínimo:

- Java Micro Edition.
- Estándar SVG.

- Conexión WIFI.

**5.3.1. Interfaz SVG.** El estándar SVG desarrollado para representar imágenes de forma vectorial utilizando XML como lenguaje de interpretación, es el estándar recomendado para desarrollar interfaces graficas ricas y con apariencia amigable para el usuario, para desarrollar las interfaces de VSHOP MÓVIL.

Se utilizo este estándar debido a que el desarrollo sobre los nuevos dispositivos móviles, permite mediante el api 226, a la maquina virtual de java entender este estándar, además el api posee clases e interfaces capaces de manipular en tiempo de ejecución las imágenes SVG; la gran ventaja de SVG, para desarrollo de interfaces graficas sobre los móviles es que los tamaños de las imágenes son muy pequeños, comparados con otros formatos más comunes como JPG ,GIF o PNG, además son muy ricas como interfaz debido a que el archivo SVG no solo contiene la imagen como tal además pueden desarrollarse animaciones, que hacen a un mas interactiva la imagen sin necesidad de aumentar en gran cantidad el tamaño de la imagen.

El paquete SVG en la aplicación VSHOP-MÓVIL, es un conjunto de clases que gracias al api 226 de java, interpreta e interactúa con la aplicación, permitiendo que el usuario vea de forma amigable interfaces mucho más atractivas sobre el móvil, y haga que la interacción con la aplicación sea más efectiva, sin necesidad que incrementar el número de pantallas que tiene que ver el usuario. El paquete está conformado por las siguientes clases:

**5.3.1.1. Paquete SVG.** El paquete SVG se compone de las siguientes clases:

- MenuSource: Clase que permite la interacción con el menú principal de VSHOP-MÓVIL, esta clase se encarga de la administración de la apariencia del menú, la carga inicial de la imagen SVG de la aplicación, y los eventos del móvil sobre el menú.
- NewClientSource: Clase que se encarga de la carga de la imagen SVG de el ingreso de nuevos clientes, la interacción con la clase Cliente y ListaCliente del paquete proceso para el ingreso de los nuevos clientes sobre la base de datos.
- ContactListSource: Clase encargada de realizar la interacción entre base de datos y la clase SVGList

- SVGList: Clase encargada de almacenar todos los elementos de la imagen SVG y realizar el ingreso de los datos de ContactListSoucer, en la imagen SVG.
- ContactListScreen: Clase encargada de los eventos del móvil sobre la imagen SVG.
- NewPrevent: Clase encargada de la imagen SVG del ingreso de pre ventas, ingreso de datos de la pre venta. La interacción con la clase Pre venta para el ingreso de los datos en base de datos.

**5.3.2. Persistente de datos.** Debido las prestaciones de los dispositivos móviles, no es muy conveniente manejar un lenguaje como SQL para guardar los datos en la memoria no volátil del móvil, además los móviles no tienen la capacidad para llevar a cabo procesos de interpretación e interacción con un servidor de base de datos.

Para llevar a cabo la tarea de lograr persistencia de datos, existe el api js86 (Rms) que es un api que maneja una forma mucha más conveniente para guardar los datos sin necesidad de crear relaciones entre tablas. Para la aplicación VSHOP-MÓVIL se desarrollo un paquete de Conexión que permite a la aplicación interactuar con el api js86 (Rms) sin necesidad de hacer procesos tediosos de programación y compatible con el paquete SVG para el manejo de los eventos del móvil sobre la interfaz.

**5.3.2.1. Paquete conexión.** El paquete SVG se compone de las siguientes clases:

- ConexiónRms: Clase encargada de abrir la conexión con la base de datos del móvil, además interpreta los eventos del móvil para guardar los datos, y las clases del paquete Proceso; Como parámetros la clase recibe el nombre del RecordStore y un valor booleano donde especifica si el recordStore no existe debe crearse.
- Filtroid: Clase encargada de realizar filtros sobre los datos de la base de datos para realizar búsquedas. Como parámetros recibe la clase de tipo proceso donde se define la búsqueda y el id para realizar la búsqueda. Esta clase implementa la interfaz RecordFilter del api Js86.
- OrdenarClientes: Clase encargada de ordenar los clientes de forma ascendente, implementa la interfaz RecordComparator del api js86.

**5.3.2.2. Paquete proceso** El paquete proceso contiene las clases que permite realizar el proceso de consulta y la interpretación de los datos de la imagen SVG sobre la aplicación. Es el paquete encargado de la capa gestión de Negocio, se compone de las siguientes clases:

- Cliente: Clase encargada de guardar los nuevos clientes en el móvil.
- ListaCliente: Clase encargada de guardar los datos de los cliente y la sincronización de estos con la aplicación web.
- Producto: Clase encargada de guardar los productos y la sincronización de los datos con la aplicación web
- Pre venta: Clase encargada de realizar las operación de pre venta y la sincronización de los datos con la aplicación web.
- Persistencia: Para realizar la operaciones con la capa de conexión, es necesario que los datos de todas las clases de proceso, estén guardados en un único flujo de datos. La configuración CLDC no implementa la característica de serialización de los datos así que la interfaz persistencia permite que los datos de las clases del paquete proceso sean convertidos en único flujo para que sean interpretados por el paquete de Conexión.
- PersistenciaXML: Esta interfaz permite que los datos que se encuentra en las clases del paquete proceso sean convertidos en formato XML para realizar la sincronización

**5.3.3. Sincronización.** La tarea de sincronización permite que los datos que se encuentra guardados en el móvil sean integrados a la aplicación VSHOP-CMS web y viceversa.

Para realizar la tarea de sincronización se tomo como base el lenguaje XML, la principal razón de elegir XML como lenguaje es que XML es un lenguaje multiplataforma, un lenguaje interpretativo de muy buenas características para guardar grandes cantidades de datos y con capacidad de ser entendido por cualquier lenguaje que cuente con un interpretador, definir nuevos campos y formas para adaptarlo a las necesidades de la aplicación y los tamaños de los archivos son mínimos así que es muy conveniente para aplicaciones móviles.

El árbol XML para enviar datos del móvil a la aplicación web es:

```
<cliente id=" iden=">
    <nombre></nombre>
    <movil></movil>
    <telefono></telefono>
    <direccion></direccion>
```

```

    <pre venta id=" total=" imp=">
      <producto id=">
        (#cantidad)
      </producto>
    </pre venta>
  </cliente>

```

El árbol XML para enviar datos de la aplicación web al móvil es:

```

<cliente id="#">(#identificacion)</cliente>
  <producto id="" >
    <cant></cant>
    <imp></imp>
    <val></val>
  </producto>

```

Para la realización de la conexión del móvil con la aplicación web se uso una conexión HTTP que permita realizar traspaso de datos entre el móvil y una página web así que se tomo como base las clases de Connection Framework de la configuración CLDC, este grupo de clases permite realizar conexión http con una capa de conexión que evita que el equipo móvil tenga que hacer comprobación de errores manualmente la capa se encarga de este problema.

La interfaz HttpURLConnection permite que una clase defina métodos para realizar una conexión pero para que la conexión sea exitosa se necesita que los datos sean un solo flujo por tal motivo se creó la clase sincronización. Dentro del paquete Sincronización se encuentra la siguiente clase:

- Sincronización: Es la clase encargada de la capa de conexión y sincronización, permite que las clases de paquete proceso al implementar la interfaz PersistenciaXML retornen un único flujo XML que permite que la clase envíe los datos. La clase además es sensible a los eventos del móvil y permite que el usuario pueda ver la ejecución de la sincronización mediante la interfaz SVG.

## CONCLUSIONES

### **Respecto a la aplicación web**

El desarrollo de herramientas que utilicen la web como plataforma permite que el usuario o cliente vea el potencial del negocio que tienen internet para aumentar sus ingresos, la aplicación web VSHOP-CMS es un herramienta capaz de lograr que los procesos de las empresas en la parte comercial alcancen gran eficiencia y permite reducir los costos de mercadeo y establecimientos comerciales. Usando como plataforma de propagación de los productos y como medio de venta internet.

El desarrollo de componentes permite que se agilice en gran medida los procesos de fabricación de software, dándoles un valor agregado y permitiendo la reutilización de dichos componentes no solo en el proyecto sino en desarrollos futuros, convirtiéndose en una gran ventaja competitiva en el mercado.

El servicio web de internet ya no es más informativo es un servicio que sea ha ido convirtiendo en una plataforma para las aplicaciones convirtiendo las simples páginas web de información en herramientas capaces de competir con aplicaciones de escritorio y con la ventaja de que tienen como base la globalización de la información para todos los usuarios. Tecnologías como Javascript, Ajax, SVG, WI-FI, WI-MAX, y las grandes prestación de ancho de banda que tienen los proveedores de conexión hacen que la red sea un mundo para que sus usuarios no solo vean información si no que creen sus propios espacios , como los blogs, las páginas personales o las comunidades virtuales, Los sistemas gestores de contenidos son las principales herramientas para que usuarios del común puedan lograr sin conocimiento alguno de lenguajes de programación como HTML ,PHP, JSP, ASP, crear sus espacios con sus gustos su forma de ver el mundo y su propia forma de mercadear sus ideas productos o servicios sin necesidad de grandes costos y logrando que sus aplicaciones alcancen prestaciones iguales o mucho mejores que aplicaciones hechas a la medida. VSHOP-CMS es una herramienta capaz de lograr que usuarios con muy poco conocimiento o usuarios profesionales logren generar su propio espacio, su tienda virtual, utilizando como base tecnologías AJAX y JAVASCRIPT, que permiten generar tiendas amigables, confiables, y de gran riqueza visual para sus clientes.

Las tecnologías Ajax y Javascript, logran que simples construcciones de páginas web se conviertan en ricos ambientes virtuales capaces de atraer y mantener al

usuario interesado en la información, productos y servicios presentes en el sitio, VSHOP-CMS no solo aplica estas tecnologías en la presentación final de la tienda, sino que también muestra la usuario interfaces graficas amigables y atractivas en su administrador permitiendo que el usuario encuentre con facilidad los puntos a cambiar en su tienda, y además reduce el tiempo de administración de campos y contenidos.

### **Respecto a la aplicación Móvil**

Las aplicaciones móviles son en perspectiva con otras aplicaciones relativamente nuevas, son aplicaciones que por las características de los dispositivos no logran grandes prestaciones, pero gracias a las nuevas tecnologías el mercado de estas aplicaciones será en muy pocos años el mercado con mayor perspectiva y crecimiento en el desarrollo de software, la plataforma JME es la principal fuerza en este desarrollo, gracias a esta plataforma las aplicaciones móviles han logrado enriquecerse en los últimos años, de encontrar dispositivos que solo podían hacer llamadas o dispositivos con aplicaciones sin conexión a internet o redes personales, dispositivos con editores de texto poco amigables o interfaces de usuario muy corrientes recordando el tiempo del “DOS” en los equipos PC, las aplicaciones y los dispositivos presentan interfaces muy ricas simulando ya los gráficos de PC actuales y con muy fácil interacción con el usuario. La aplicación VSHOP-MÓVIL logra que dispositivos que usan la aplicación tengan con el usuario una interacción amigable, con gran confiabilidad y que el usuario tenga en sus manos una aplicación que utilice la menor cantidad de pasos posibles para lograr su cometido.

El principal inconveniente para la plataforma JME, es que esta plataforma no tiene un estándar definido para los productores de equipos móviles así que en muchos casos las aplicaciones no son compatibles con todos los existentes, además tienen capacidades diferentes en su hardware. La plataforma se dividió en partes para que sea compatible con todos los equipos así que se depende en gran medida del equipo como tal, este es un inconveniente menor ya que la tecnología del hardware para dispositivos móviles tiene un gran avance, como ejemplo podemos tomar el desarrollo de la aplicación VSHOP-CMS, al comenzar el desarrollo de esta aplicación no existían equipos capaces de entender e interpretar el estándar SVG, ni siquiera existían emuladores para realizar las pruebas para esta especificación y para el API como tal, pero en este instante ya existen equipos capaces de utilizar este estándar, además es el más difundido y confiable para el desarrollo de interfaces para equipos móviles.

## RECOMENDACIONES

Como punto de partida para trabajos futuros se han establecido una serie de recomendaciones.

Desarrollar aplicaciones web que tengan como base la ideología WEB 2.0 es el futuro del desarrollo de software, internet ya no es más una red de información es una red de interacción, las aplicaciones deben permitir que los productos y servicios se globalicen, así que el estudio y desarrollo de aplicaciones web que utilicen la red como plataforma debe ser el paso a seguir para todos los desarrolladores.

Integrar los procesos de desarrollo de componentes para proyectos futuros, ya que permiten la reutilización y optimizan en gran medida el proceso de desarrollo de cualquier tipo de aplicación.

Incentivar el desarrollo de aplicaciones móviles, el mundo móvil está creciendo a pasos agigantados, los siguientes sistemas operativos de compañías como Nokia y Google, son sistemas operativos orientados a que el mundo móvil sea el siguiente paso en el mercadeo de la información de productos y servicios, la plataforma JME es la principal herramienta para estas compañías para el desarrollo de estos sistemas, además estas compañías generan ya nuevos Framework para desarrollo de aplicaciones para sus sistemas operativos, así que el siguiente paso de la ciencia de la informática y el desarrollo de sistemas es la movilidad, los usuarios ya no quieren su información quieta, la quieren donde estén en cualquier punto en el mundo, gracias a internet y a los nuevos avances de hardware en el campo móvil, la mayor demanda de aplicaciones será para este tipo de dispositivos.

Implementar nuevos tipos de pagos en línea para la aplicación web de VSHOP CMS, de acuerdo a los requerimientos del mercado en la tienda virtual.

Implementar la conexión persistente entre la aplicación móvil y la aplicación web de VSHOP-CMS según las nuevas perspectivas de las tecnologías de conexión y la demanda del mercado.

Desarrollar una nueva aplicación móvil para el cliente final de VSHOP-CMS, que le permita realizar compras sobre la tienda virtual, sin necesidad de conexiones persistentes para reducir los costos de la prestación del servicio de conexión.

Finalmente, el desarrollo de este trabajo y de esta herramienta ha permitido que apliquemos todo el conocimiento adquirido en el programa de Ingeniería de Sistemas, y generar nuevos conocimientos y fuente de trabajo con el desarrollo de un nuevo Framework para las aplicaciones web y conexión entre aplicaciones web y móvil.

## ANEXOS

### A1. GUIA DE INSTALACIÓN DE LA APLICACIÓN

El primer paso es instalar un servidor web y el servidor Mysql, para ello usaremos XAMPP aplicación que permite instalar fácilmente los dos. Descargamos el archivo de XAMPP en [www.apachefriends.org](http://www.apachefriends.org).

El archivo de XAMPP esta comprimido en formato SFX 7-Zip. El pack cuenta con su instalador (CL) una vez que los archivos exe sé hallan descomprimido en el directorio deseado.

Una vez ejecutado el instalador no deben mover los archivos que se encuentran dentro de /xampp eso implicaría tener que reconfigurar el pack. Cuenta con actualizaciones de solo los archivos a cambiar, facilitando el uso y el tamaño en descarga.

#### Instalación

Los archivos se encuentran en SFX lo que facilita la instalación. Al ejecutar el archivo xampp-win32-1.6.5-installer.exe, aparecerá la siguiente pantalla:

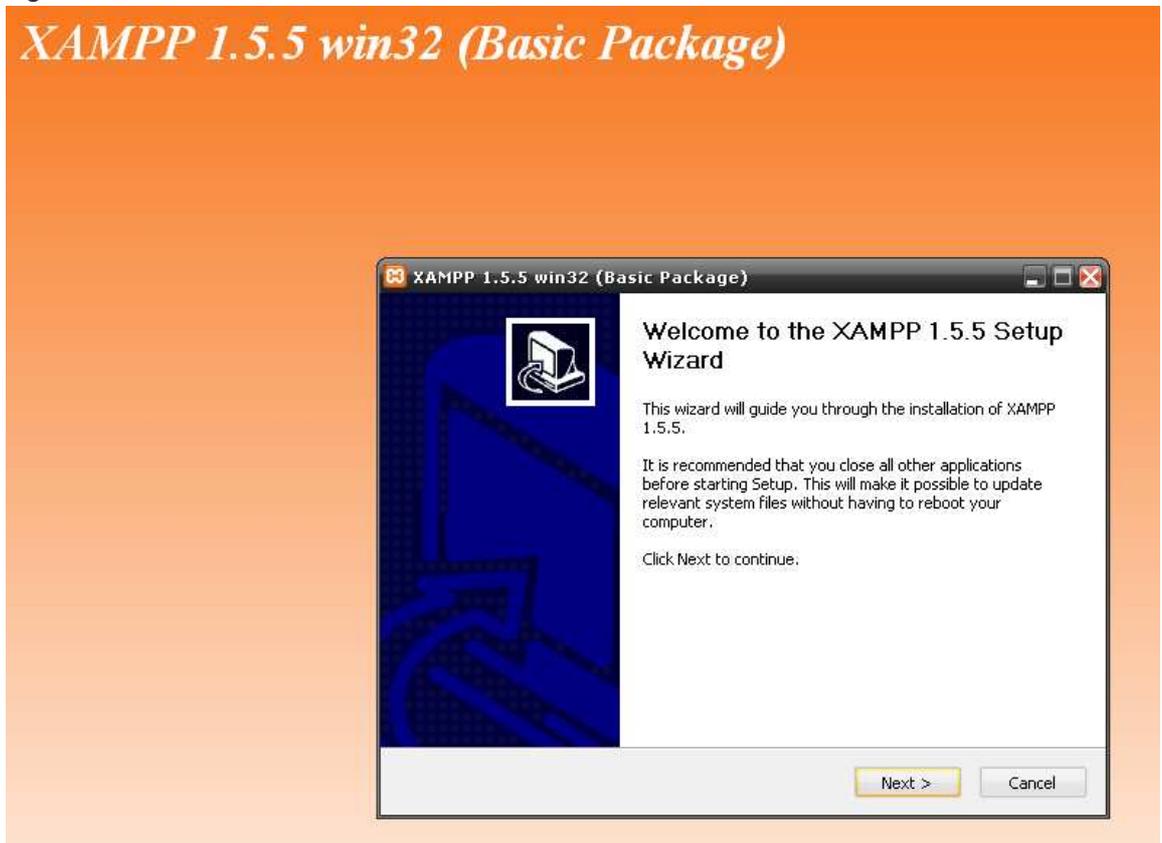
Figura 84 Pantalla inicial de instalación XAMPP



Seleccione English, y luego oprima Ok.

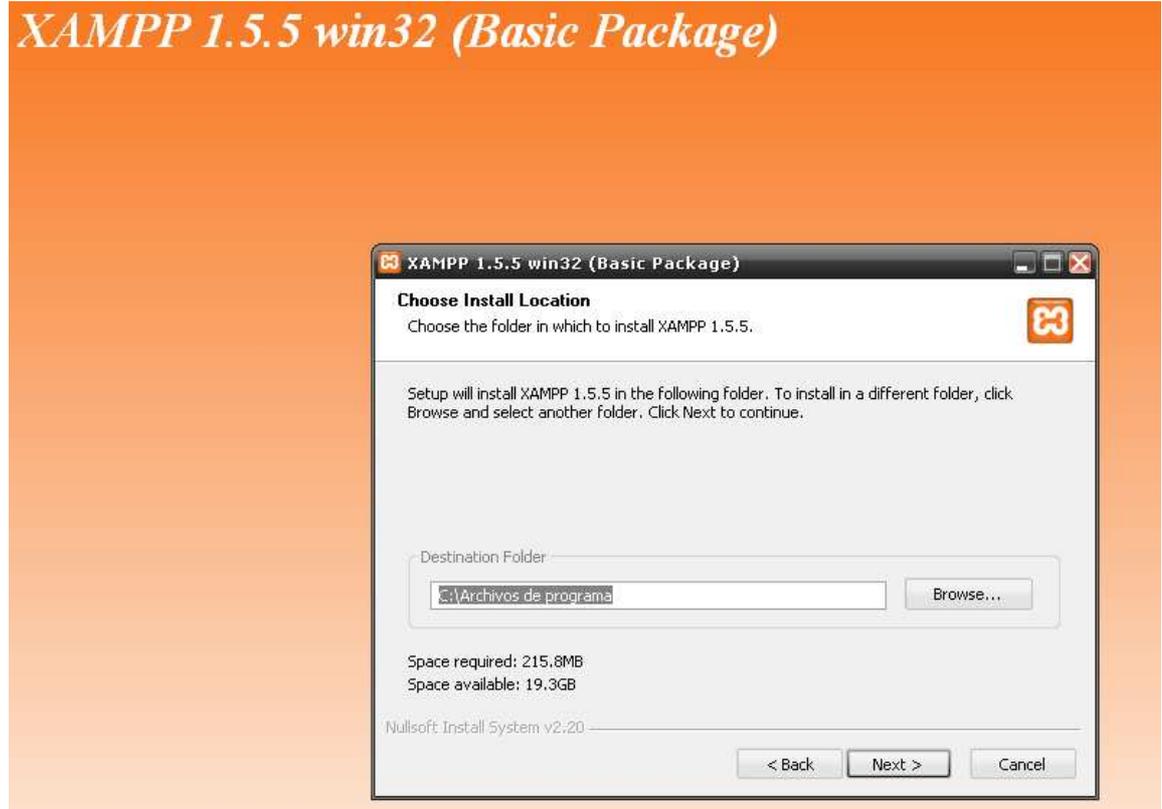
La pantalla que se presenta a continuación, es la de bienvenida. Luego de leerla oprima Next para continuar la instalación.

Figura 85 Inicio de instalación XAMPP



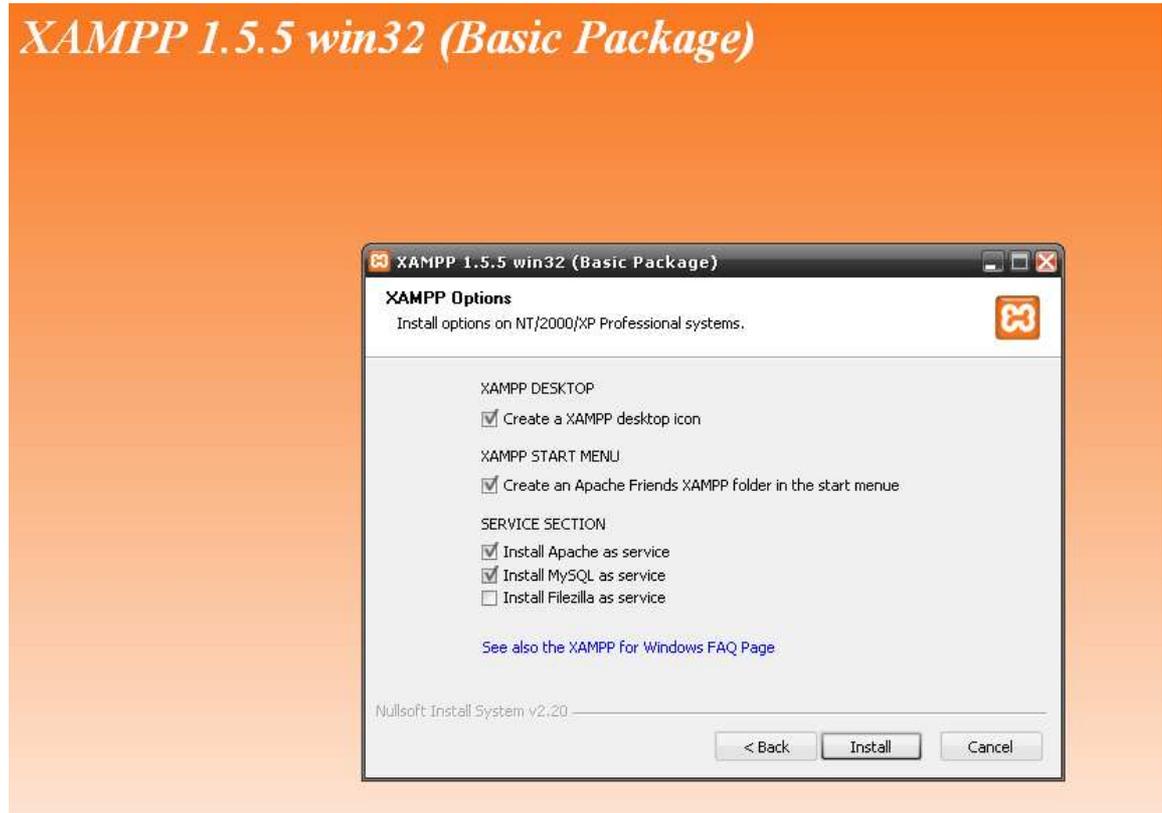
Se recomienda instalarlo en el directorio principal que se muestra en la siguiente imagen, luego de elegir la ubicación ("Destination Folder"), oprima Next:

Figura 86 Instalación XAMPP - Selección directorio destino



Seleccione las opciones que se muestran a continuación y presione Install.

Figura 87 Instalación XAMPP - Servicios



Luego de la instalación aparecerá una ventana del sistema, con el objetivo de hacerle saber que el XAMPP ha sido instalado con éxito.

Figura 88 Instalación XAMPP - Finalización

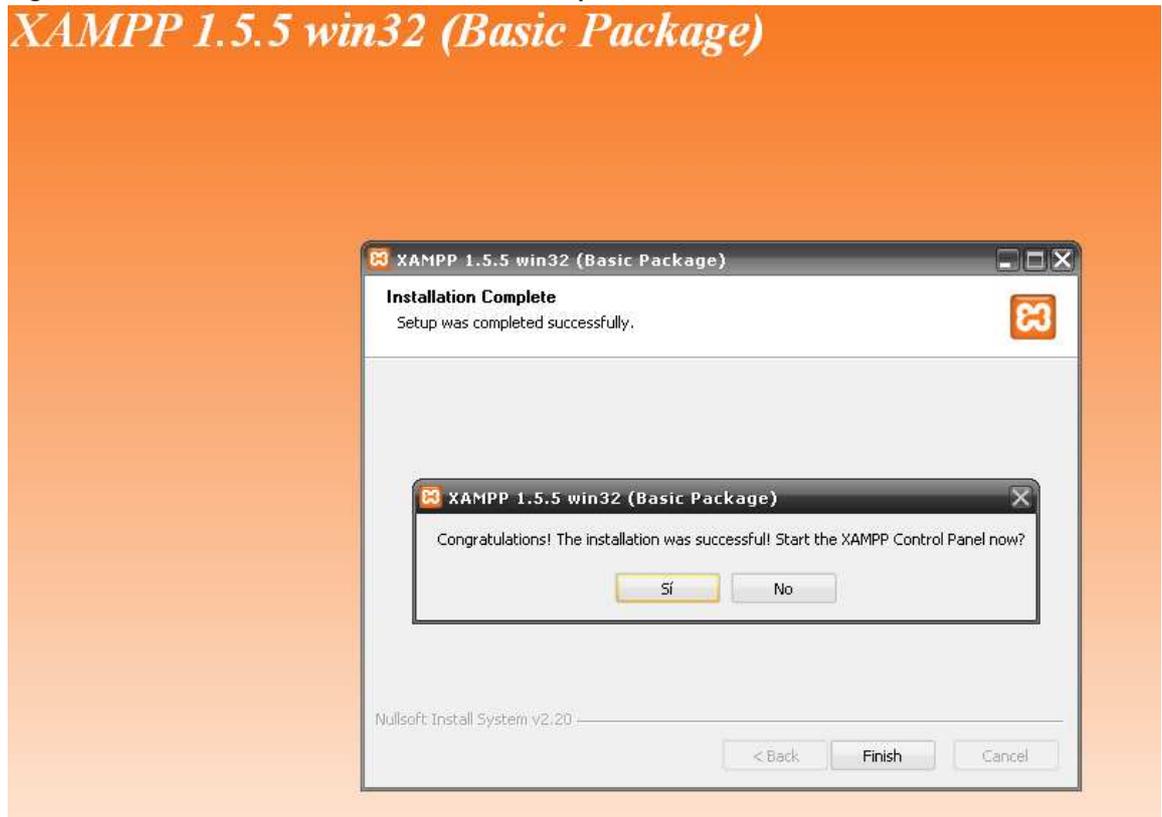
## *XAMPP 1.5.5 win32 (Basic Package)*



Luego pulse Finish.

Figura 89 Instalación XAMPP - Confirmación inicio panel de control

## *XAMPP 1.5.5 win32 (Basic Package)*



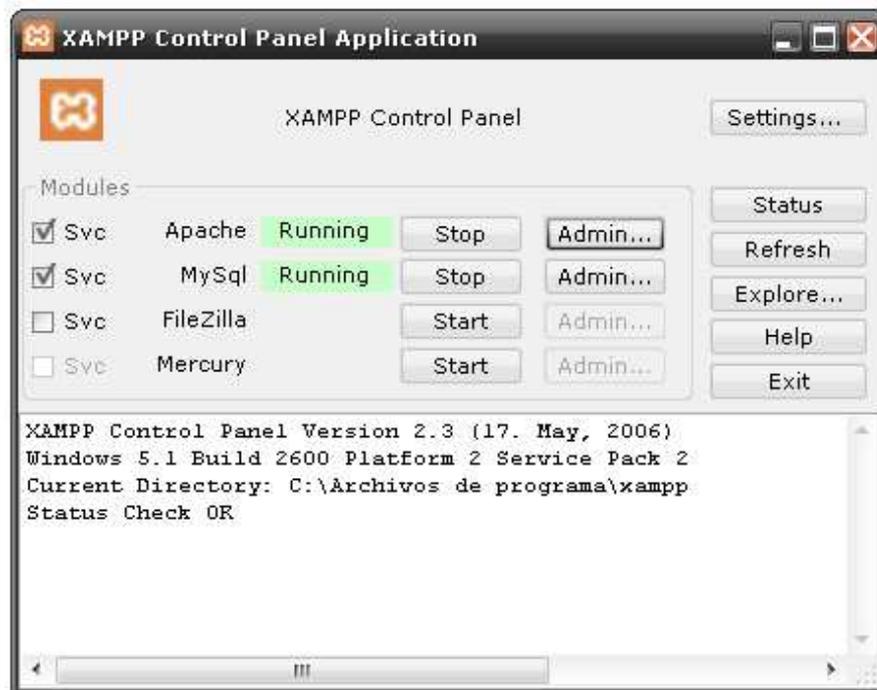
Una vez finalizado, el XAMPP se encargara de configurar los puertos del Apache. Puede suceder que se solicite que se desbloqueen los puertos, en ese caso desbloquéelos.

Luego de la configuración de los puertos, el instalador le preguntará si desea abrir el panel de control ahora. Si desea seguir la configuración presione Si, de lo contrario ingrese No. (Ver imagen superior).

### **Panel de Control de XAMPP**

Se presenta a continuación una imagen del panel de control.

Figura 90 Panel de control XAMPP



Si se desea configurar a mayor escala se debe hacer clic en admin el cual lo guiara a las configuraciones del servidor que ha seleccionado.

NOTA: Para el correcto funcionamiento de VSHOP CMS se debe tener Corriendo ("Running") las dos opciones presentada en la imagen superior.

Ya teniendo instalados Apache y MySQL, debemos hacer lo siguiente:

1. Desde <http://localhost/phpmyadmin> creamos la base de datos llamada VSHOP.
2. Damos clic en importar, seleccionamos el archivo vshopsql.sql que tenemos en el CD de instalación.

Figura 91 PHPMYADMIN - Importar archivo

Servidor: localhost ▶ Base de datos: vshop

Estructura SQL Buscar Generar una consulta Exportar **Importar** Diseñador Operaciones

Privilegios Eliminar

### Importar

**Archivo a importar**

Localización del archivo de texto:  No se ha seleccionado archivo (Tamaño máximo: 16,384 KB)

Juego de caracteres del archivo: utf8

La compresión escogida para el archivo a importar se detectará automáticamente de: Ninguna, gzip, zip

**Importación parcial**

Permita la interrupción de la importación en el caso de que el script detecte que se ha acercado a su límite de tiempo. Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.

Número de registros (consultas) a saltarse desde el inicio:

**Formato del archivo importado**

DocSQL

SQL

**Opciones**

Modalidad compatible con SQL:

[Abrir nueva ventana de phpMyAdmin](#)

3. Por último copiamos en la carpeta c:/xampp/htdocs la carpeta VSHOP que viene en el CD de instalación.

## A2. GUIA DE INSTALACIÓN LAMPP EN LINUX

Paso a paso: El paquete se llama Xampp (Lampp para Linux).

1. Descargamos el paquete Xampp en la carpeta del directorio raíz /opt desde la siguiente dirección:

[http://sourceforge.net/project/showfiles.php?group\\_id=61776&package\\_id=60248](http://sourceforge.net/project/showfiles.php?group_id=61776&package_id=60248)

Se dirige a la sección descargas y muestra versiones actuales y antiguas del producto, debe elegir la más reciente de XAMPP Linux, que es un paquete de más o menos 51.mb

2. Abra una consola como administrador con el comando “su” y descomprima y desempaque el archivo con el comando:

```
tar xvfz xampp-linux-1.5.3a tar.gz -C /opt
```

Donde 1.5.3 es la versión que descargo

3. Eso es todo ahora Xampp ya está instalado en el PC en el directorio /opt/lampp
4. Para arrancar Xampp simplemente teclee este comando en la consola /opt/lampp/lampp start con esto se inicia los servicios de Mysql, Apache, PHP
5. Para probar xampp abra el navegador y escriba:

```
http://localhost
```

Aparecerá la pantalla de bienvenida de xampp con algunos links para visualizar el estado de los servicios e instalar o desinstalar aplicaciones que vienen con el paquete.

6. El paquete se instala por defecto sin configuraciones de seguridad, lo cual podría ser fatal para un ambiente de producción puesto que:
  - El administrador del mysql (root) no tiene password.

- El demonio del Mysql es accesible vía Internet.
- ProFtpd (el servicio para subir archivos al servidor) usa el password lampp para el usuario “nobody”.
- PhpMyAdmin es accesible vía Internet.
- MySQL Y Apache funcionan con el mismo usuario (nobody)
- Para agregar la seguridad simplemente escriba en la consola:

```
/opt/lampp/lampp security
```

- Con este comando el sistema de seguridad pedirá los password de los servicios.

7. Si las aplicaciones de PHP, funcionan con la versión PHP4 o PHP5, no hay ningún problema puesto que se eligió con que versión trabajar, con este comando se soluciona el problema:

```
/opt/lampp/lampp php4 o /opt/lampp/lampp php5
```

Si desconoce con que versión está trabajando, entonces escriba:

```
/opt/lampp/lampp phpstatus
```

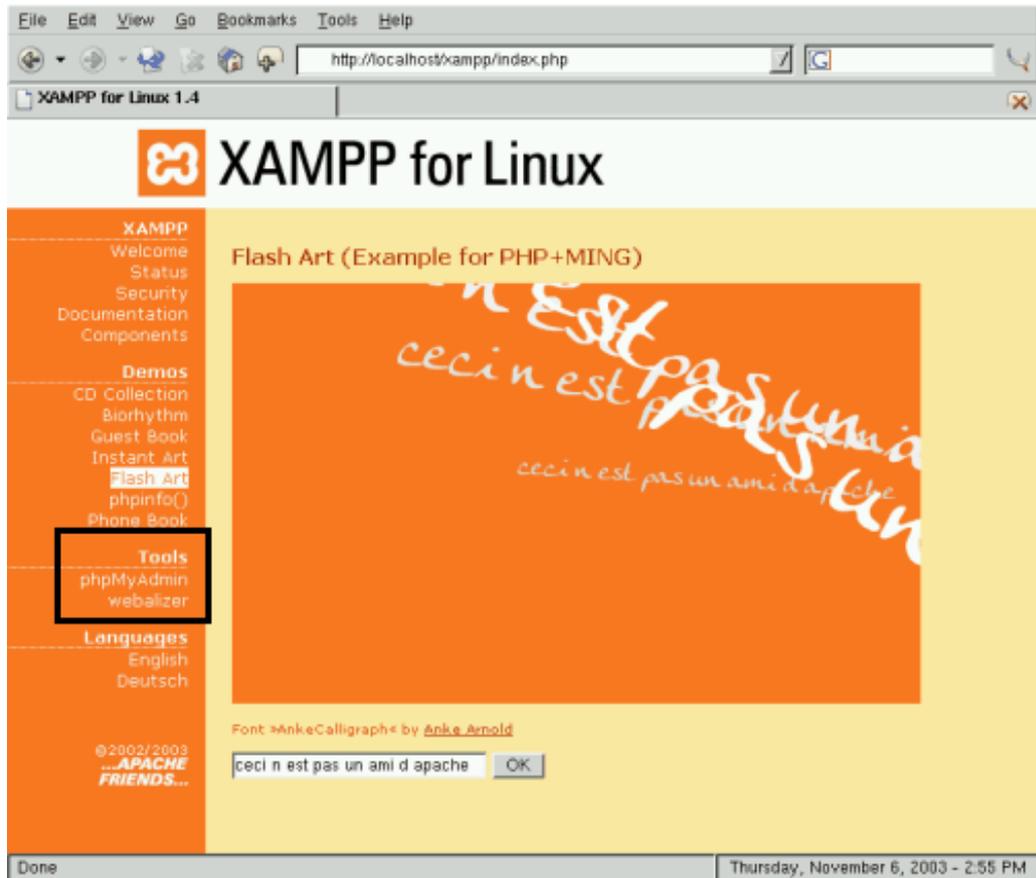
8. Aquí comandos básicos para manejar xampp, seguidos de /opt/lampp/lampp:

- start: Inicia XAMPP.
- stop: Detiene XAMPP.
- restart: Detiene e inicia XAMPP.
- startapache: Inicia solo Apache con el soporte SSL (Protocolo de seguridad).
- stopapache: Detiene el apache.
- stopssl: Detiene el apache con el soporte SSL
- startmysql: Inicia solo el servidor de bases de datos MySQL
- stopmysql: Detiene el servidor de bases de datos MySQL
- startftp: Inicia el servidor ProFTPD, para subir archivos desde el servidor web.
- stopftp Detiene el servidor ProFTPD
- security: Inicia un programa de revisión de la seguridad.

Para comprobar que todo ha ido bien abra el navegador y escriba la siguiente dirección:

http://localhost/

Figura 92 Instalación Lammpp en Linux



## REFERENCIAS BIBLIOGRÁFICAS

[1] BORJA SALAZAR, Comercio Electrónico: Empresario Tecnológico. Aspectos Estratégicos, Empresariales Y Tecnológicos De Los Negocios Electrónicos, Editorial Desclée de Brouwer.

[2] Delitosinformaticos.com, “Seguridad En Comercio Electrónico”, en <http://www.delitosinformaticos.com/ecommerce/cargos.shtml>, consultado en Noviembre 03 de 2007.

[3] WIKIPEDIA®, “Sistema de gestión de contenido”, en [http://es.wikipedia.org/wiki/iuyinet/Sistema\\_de\\_gestión\\_de\\_contenido](http://es.wikipedia.org/wiki/iuyinet/Sistema_de_gestión_de_contenido), consultado en Noviembre 13 de 2007.

[4] MOSAIC, “Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto”, en <http://mosaic.uoc.edu/articulos/cms1204.html>, consultado en Noviembre 14 de 2007.

[5] Microsoft Corporation ©, “Desarrollo de Software basado en Componentes”, en <http://msdn.microsoft.com/es-es/library/bb972268.aspx>, consultado en Diciembre 2 de 2007.

[6] WIKIPEDIA®, “XML”, en <http://es.wikipedia.org/wiki/XML>, consultado en Enero 20 de 2008.

[7] MAXIMAIL, “Curso: Desarrollo de aplicaciones para dispositivos inalámbricos (J2ME)”, en <http://www.mailxmail.com/curso/informatica/dispositivosinalambricos/>, consultado en Marzo 10 de 2008.

---

[8] ABSERVER, “Ab shop”, en <http://www.ab-shop.com/producto.asp> , consultado en Marzo 10 de 2008.

[9] WIKIPEDIA®, “Joomla!”, en <http://es.wikipedia.org/wiki/Joomla>, consultado en Marzo 11 de 2008.

[10] WORDPRESS, “Joomla, el todopoderoso”, consultado en sitio web <http://blogestores.wordpress.com/2006/10/30/joomla-el-todopoderoso/>, consultado en Junio 2 de 2008.

[11] WIKIPEDIA®, “Mambo (CMS)”, consultado en sitio web [http://es.wikipedia.org/wiki/Mambo\\_Open\\_Source](http://es.wikipedia.org/wiki/Mambo_Open_Source), consultado en Marzo 12 de 2008.

[12] DESARROLLOWEB.COM, “¿Qué es OsCommerce?”, consultado en <http://www.desarrolloweb.com/articulos/2132.php>, consultado en febrero 18 de 2008.

[13] CREATIVECOMMONS ©, “Mapa Visual de la Web 2.0”, en <http://www.internality.com/web20/>, consultado en Mayo 20 de 2008.

[14] MAESTROS DEL WEB, “¿Qué es la Web 2.0?”, en <http://www.maestrosdelweb.com/editorial/web2/>, consultado en Julio 16 de 2008.

[15] THE PHP GROUP, “Manual de PHP”, en <http://www.php.net/manual/es/>, consultado en Noviembre 14 de 2007.

[16] WIKIPEDIA®, “JavaScript”, en <http://es.wikipedia.org/wiki/JavaScript>, consultado en Diciembre 2 de 2007.

---

[17] WIKIPEDIA ®, “Ajax”, en <http://es.wikipedia.org/wiki/AJAX>, consultado en Diciembre 11 de 2007.

[18] WIKIPEDIA ®, “XAJAX” , en <http://es.wikipedia.org/wiki/Xajax>, consultado en Diciembre 11 de 2007.

[19] SUN MICROSYSTEMS, “Java ME”, en <http://java.sun.com/javame/index.jsp>, consultado en marzo 4 de 2008.

[20] WIKIPEDIA ®, “Servidor HTTP Apache”, consultado en [http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache), consultado en Febrero 21 de 2008.

[21] WIKIPEDIA ®, “MySQL”, en <http://es.wikipedia.org/wiki/MySQL>, consultado en Febrero 21 de 2008.

[22] APACHE FRIENDS, “XAMPP”, consultado en sitio web <http://www.apachefriends.org/en/xampp.html>, consultado en diciembre 14 de 2007.

[23] KENDALL Y KENDALL, Análisis Y Diseño Sistemas 6<sup>a</sup>, Prentice Hall.

[24] WIKIPEDIA®, “Sistema Operativo”, consultado en sitio web [http://es.wikipedia.org/wiki/Sistema\\_operativo](http://es.wikipedia.org/wiki/Sistema_operativo), consultado en Septiembre 17 de 2008.

[25] UNIVERSIDAD PÚBLICA DE TARRAGONA, “Glosario”, en <http://www.sre.urv.es/sredec/glosario/index08.htm>, consultado en Agosto 21 de 2008.

[26] WIKIPEDIA®, “Hojas de estilo en cascada”, en <http://es.wikipedia.org/wiki/CSS>, consultado en Enero 20 de 2008.