



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE
HIDALGO

Instituto de Ciencias Básicas e Ingeniería

Tienda de Abarrotes Virtual
TAV-Hidalgo

TESIS

Que para obtener el título de:
Lic. en Sistemas Computacionales

PRESENTAN:

María Isabel Carbajal Amador
Miriam Vargas Cruz

Asesor:

M. C. Arturo Curiel Anaya

Coasesor:

M. C. C. Luís Heriberto García Islas

Pachuca de Soto Hgo., Julio 2006



DEDICATORIA

A Dios;

A nuestros Padres;

A nuestros Hermanos (a);

A nuestros Amigos (as).

“La humildad trae gracia y felicidad a la vida; permite acomodarse a las situaciones difíciles sin pensar en lo que se está dejando ó renunciando, nos vuelve más sencillos y naturales, permite que nos concentremos en lo que estamos haciendo, y que lo hagamos correctamente.

La humildad hace que podamos ver los beneficios en cada escena de la vida, haciendo que nuestras interacciones giren en un ambiente más agradable, así logramos un lugar en el corazón de todos, eliminando en un segundo aquello que nos hiere y no nos deja crecer.

Siendo humildes comprenderemos que aún tenemos mucho por mejorar, mucho que aprender y que podemos ocuparnos en la tarea de crecer”.

Andrea Moreno de Buitrago

Isabel y Miriam



AGRADECIMIENTOS

Cuando uno empieza un camino nunca sabe qué largo va a ser y cuántas personas se cruzarán en él. En esta tesis muchas son las personas que directamente o indirectamente han hecho posible que llegara a su fin.

Al M. C. Arturo Curiel Anaya:

Por todo el tiempo que nos ha dado, por sus sugerencias e ideas de las que tanto provecho hemos obtenido, por su respaldo y amistad.

Al I. S. C. Jorge Hernández Camacho:

Por tener la paciencia ante nuestras dudas de novatas y por escuchar atentamente los problemas que a lo largo de esta Tesis surgieron, por tu amistad.

A nuestros Sinodales:

Queremos dar las gracias a la gente que revisó con paciencia este trabajo, por sus aportaciones y sugerencias: Dra. Aurora Pérez Rojas, Dr. Omar Arturo Domínguez Ramírez, M.C. Arturo Curiel Anaya, M.C.C. Luís Heriberto García Islas, L.C. Norma Laura Salazar Viveros, L.I. Theira Irasema Samperio Monroy y L.S.C. Luís Ramón Macías Pulido

Al M. C. C. Luís Heriberto García Islas:

Por tu disposición permanente e incondicional en aclarar nuestras dudas y por tus sugerencias durante la realización de la Tesis, por tu amistad.

Al Arq. Alejandro Rivera Muñoz:

Gracias por la amistad brindada, las sugerencias y contribuciones que para este trabajo realizaste, que resultaron de gran utilidad.

A nuestros Catedráticos:

Gracias a todos los que contribuyeron realmente en nuestra formación.

A la P.L.S.C. Marlen Luna Vera:

Por su amistad, apoyo y por compartir agradables instantes.

Isabel y Miriam



Para ser exitoso no tienes que hacer cosas extraordinarias.
Haz cosas ordinarias, extraordinariamente bien.

A Dios:

Por permitirme cumplir esta
meta de mi vida.

A mi Papá:

Por ser mi guía, por su amor, apoyo, consejos y confianza que depositó en mí. Y por que sé que donde quiera que estés, compartes conmigo este logro, que es de los dos.

A mi Hermano:

Por estar siempre conmigo, apoyarme e impulsarme en todo lo que hago.

A mi Familia:

Por que siempre me ha apoyado y alentado durante el desarrollo de mi carrera y de mi vida. En especial a mi tío Antonio porque siempre me ha impulsado a luchar por lo que quiero para conseguirlo, y mi prima María de Jesús por alentarme a terminar la tesis y por su apoyo.

A Alejandro:

Por que sin tu apoyo, paciencia y ayuda no habría sido o posible terminar la tesis.

A mi Mamá:

Por su amor, paciencia, desvelos, consejos y sobre todo por tu apoyo y oraciones, con las que siempre me acompañó a lo largo de mis estudios.

A mi Abuelita Leonor:

Por ser mi inspiración, por tu fortaleza y cariño. Y por que sé que donde quiera que estés, te sientes orgullosa de mí.

A mi Amiga Miriam:

Por que sin tu amistad, paciencia, apoyo y confianza que depositaste en mí para que juntas hiciéramos la tesis, no hubiera sido posible llegar a esta meta.

A mis Amigas (os):

Gracias por todo el apoyo que me brindaron a lo largo de mi carrera y vida, tanto en las buenas como en las malas. En especial a Karis.

Isabel



Hay veces en que miro atrás y me doy cuenta de que todas las personas que han pasado y pasan por mi vida, se merecen mi agradecimiento por componerme como persona.

A Dios:

Por permitirme llegar hasta este momento tan importante de mi vida y lograr otra meta más en mi carrera.

A mi Bisabuela:

Tomasita. Aunque ya no estas conmigo quiero darte las gracias por tu cariño, por preocuparte por mí y por confiar en mí,

A mi Abuela:

María. Por tu cariño, apoyo, por preocuparte por mí y consentirme.

A mi Tío:

Ernesto. Aunque físicamente ya no estás conmigo quiero darte las gracias por tu cariño, apoyo y consejos.

A mi papi:

Alejandro. Gracias por tu sincera amistad y por compartir tu tiempo conmigo.

A Isabel y a su Familia:

Gracias por su cariño y amistad.

A ti *Isa* gracias por apoyarme y haber confiado en mí en iniciar la aventura de hacer la tesis juntas, creo que hacemos un gran equipo.

A mis Padres:

A ti *mamá.* Por tu comprensión, apoyo confianza, y tu amor. Gracias por dedicarme gran parte de tu tiempo y por enseñarme a caminar hacia la perspectiva de un nuevo día.

A ti *papá.* Por tu cariño y apoyo que me has dado a lo largo de mi vida.

A mis Tías:

Isi, Vero, Anselma, Pachis, Lore, Claudia, Faty, Nancy. Gracias por su cariño, apoyo, consejos y por creer en mí.

A mis Tíos:

Flavio, Jesús, Alberto, Luís, Cesar, Emilio y Tomas. Gracias por su cariño, apoyo, consejos y por confiar en mí.

A mis Primos:

Luís, Cesar, Nataneel, Rafael, Francisco, Gabriela, Roberto e Isaac. Gracias por regalarme una sonrisa y por recordarme que todos tenemos un niño en nuestro interior.

A mi amigo:

Antonio. Gracias por tu amistad y por tu ayuda que siempre me brindas.

A mi amor Alejandro:

Por tu apoyo, comprensión y amor que me permite sentir poder lograr lo que me proponga. Gracias por escucharme y por tus consejos. Gracias por ser parte de mi vida; eres lo mejor que me ha pasado.

A mi Madrina:

Catalina. Por tu cariño, confianza y apoyo. Gracias por ser siempre mi cómplice cuando más necesito de tu ayuda.

A mis Hermanos:

Lolbee y Martín, aunque somos primos siempre serán mis hermanos. Gracias por confiar en mí, por su apoyo y por creer en mí.

A la Familia de Alejandro:

A la cual le debo agradecer muchas cosas, pero principalmente el haberme recibido en su hogar.

A mis Amigos y Amigas:

Gracias por su amistad, apoyo y por compartir sus experiencias, desveladas y triunfos durante toda la carrera.

Miriam



ÍNDICE

INTRODUCCIÓN	i
I. Objetivo General	ii
II. Objetivos Específicos	ii
III. Justificación	iii
IV. Alcances	iii
V. Limitaciones	iv
CAPÍTULO 1: ESTADO DEL ARTE	
1.1 Comercio Electrónico	1
1.2 Concepto de Tienda Virtual	3
1.3 Características de una Tienda Virtual	3
1.3.1 El Local	3
1.3.2 El Escaparate	4
1.3.3 Los Productos	4
1.3.4 El Proceso de Venta	4
1.3.5 El Carrito de Compra	5
1.3.6 Los Medios de Pago	5
1.3.7 El Servicio en la Tienda	5
1.3.8 La Entrega del Material	6
1.3.9 La Tras-tienda Virtual	6
1.3.10 Procesos Internos	6
1.3.11 Información de Marketing	7
1.4 Elementos que Conforman la Tienda Virtual	7
1.5 Aplicaciones Existentes	8
1.5.1 Trico	9



1.5.2	La Playa	11
1.5.3	Sumercado.com	12
1.5.4	My Brands	14
1.5.5	El Corte Inglés Sumercado	16
1.6	Software para el Desarrollo de Tiendas Virtuales	17
1.6.1	Cube Stores	17
1.6.2	AB-Shop	19
1.6.3	e-Shop	22
1.6.4	Neostore	24
1.6.5	SACE (Sistema de Administración de Comercio Electrónico)	26
 CAPÍTULO 2: METODOLOGÍA DE DESARROLLO		
2.1	Metodologías para el Diseño de Sitios Web	28
2.1.1	OOWS (Object-Oriented Approach for Web Solutions Modeling)	31
2.1.1.1	Notación Utilizada por OOWS	33
 CAPÍTULO 3: MARCO TECNOLÓGICO		
3.1	Lenguajes de Programación Web	35
3.1.1	PHP	35
3.1.2	HTML	36
3.1.3	FLASH	37
3.1.4	JavaScript	38
3.2	Herramienta para el Diseño de Páginas Web	39
3.2.1	Dreamweaver	39
3.3	Lenguaje de Desarrollo de Ambientes Virtuales	39
3.3.1	VRML	39



3.4	Herramienta de Desarrollo de Ambientes Virtuales	40
3.4.1	3D Studio Max	40
3.5	Servidor Web	41
3.5.1	Apache	41
3.6	Sistema Gestor de Bases de Datos	42
3.6.1	MySQL	42
CAPÍTULO 4: DESARROLLO DEL SISTEMA		
4.1	Introducción	44
4.2	Desarrollo de la Aplicación	44
4.3	Requerimientos del Sistema	45
4.4	Metodología de Análisis y Diseño del Sistema	45
4.4.1	Especificación del Problema	46
4.4.2	Desarrollo de la Solución	75
4.5	Funcionamiento de TAV-Hidalgo	85
CAPÍTULO 5: CONCLUSIONES Y TRABAJOS FUTUROS		
5.1	Conclusiones	87
5.2	Trabajos Futuros	88
GLOSARIO		90
REFERENCIAS BIBLIOGRÁFICAS		95
REFERENCIAS ELECTRÓNICAS		97
ANEXO 1. CÓDIGO FUENTE		101
ANEXO 2. PONENCIA TAV-Hidalgo		134



ÍNDICE DE FIGURAS

CAPÍTULO 1: ESTADO DEL ARTE

Figura 1.	Pantalla Principal de Trico	9
Figura 2.	Pantalla Principal de La Playa	11
Figura 3.	Pantalla Principal de Sumercado.com	13
Figura 4.	Pantalla Principal de My Brands	14
Figura 5.	Pantalla Principal de El Corte Inglés Supermercado	16
Figura 6.	Pantalla Principal de AB-Shop	20
Figura 7.	Pantalla Principal de e-Shop	23
Figura 8.	Pantallas de la Sección Pública y del Panel de Control de Neostore	25
Figura 9.	Empaque del Software SACE	27

CAPÍTULO 4: DESARROLLO DEL SISTEMA

Figura 10.	Croquis de Aurrera	48
Figura 11.	Croquis del Mercado Soriana	50
Figura 12.	Caso de Uso del Agente Usuario Navegante	52
Figura 13.	Caso de Uso del Agente Administrador	53
Figura 14.	Modelo de Objetos	56
Figura 15.	Modelo Dinámico del Pedido	57
Figura 16.	Parte del Modelo Funcional de la Clase Paquete	57
Figura 17.	Mapa de Navegación del Usuario Navegante	58
Figura 18.	Mapa de Navegación del Administrador	59
Figura 19.	Contexto Productos para el Usuario Navegante	60
Figura 20.	Contexto Altas para el Administrador	60
Figura 21.	Contexto Bajas para el Administrador	61



Figura 22.	Contexto Modificaciones para el Administrador	61
Figura 23.	Contexto Visualizar para el Administrador	62
Figura 24.	Contexto Inventarios para el Administrador	62
Figura 25.	Contexto Visualizar para el Administrador	63
Figura 26.	Modelo de Presentación del Contexto Producto para el Usuario Navegante	63
Figura 27.	Modelo de Presentación del Contexto Altas para el Administrador	64
Figura 28.	Modelo de Presentación del Contexto Bajas para el Administrador	64
Figura 29.	Modelo de Presentación del Contexto Modificaciones para el Administrador	65
Figura 30.	Modelo de Presentación del Contexto Inventarios para el Administrador	65
Figura 31.	Modelo de Presentación del Contexto Pedido para el Administrador	66
Figura 32.	Diagrama Entidad-Relación	67
Figura 33.	Modelo Relacional	68
Figura 34.	Plano de Planta de TAV-Hidalgo	74
Figura 35.	Fachada de TAV-Hidalgo	74
Figura 36.	Perspectivas de TAV-Hidalgo	75
Figura 37.	Pantalla para exportar el Mundo Virtual a formato VRML	76
Figura 38.	Pantalla del Exportador de VRML	76
Figura 39.	Función Anchor en VRML	77
Figura 40.	Código SQL, PHP y HTML para Obtener y Visualizar los Datos del Producto	78
Figura 41.	Modelo de la Base de Datos	79



Figura 42.	Pantalla para Exportar la Base de Datos a Código SQL	79
Figura 43.	Parte del Código SQL Generado por Desing/IDEF	80
Figura 44.	Pantalla para Crear una Base de Datos en phpMyAdmin	80
Figura 45.	Pantalla para Generar la Base de Datos en phpMyAdmin	81
Figura 46.	Pantalla que Visualiza la Base de Datos en phpMyAdmin	81
Figura 47.	Interfaz Principal de TAV-Hidalgo	83
Figura 48.	Vista de la Tienda TAV-Hidalgo	84
Figura 49.	Interfaz Administrativa de TAV-Hidalgo	84
Figura 50.	Funcionamiento de TAV-Hidalgo para el Usuario Navegante	85
Figura 51.	Funcionamiento de TAV-Hidalgo para el Administrador	86



ÍNDICE DE TABLAS

CAPÍTULO 2: ESTADO DEL ARTE

Tabla 1.	Comparación entre las Metodologías para el Diseño de Páginas Web	30
----------	--	----

CAPÍTULO 4: DESARROLLO DEL SISTEMA

Tabla 2.	Registrar Usuario Navegante	54
Tabla 3.	Cancelar / Confirmar Pedido	54
Tabla 4.	Surtir Pedido	55
Tabla 5.	Calcular Costo Promedio	55
Tabla 6.	Tabla de Calle	69
Tabla 7.	Tabla de Clasificación	69
Tabla 8.	Tabla de Cliente	69
Tabla 9.	Tabla de Colonia	69
Tabla 10.	Tabla de Dirección	70
Tabla 11.	Tabla de Empleado	70
Tabla 12.	Tabla de la Empresa	70
Tabla 13.	Tabla del Estado	70
Tabla 14.	Tabla del Estado Civil	71
Tabla 15.	Tabla del Estado del Pedido	71
Tabla 16.	Tabla del Estatus	71
Tabla 17.	Tabla del Inventario	71
Tabla 18.	Tabla de la Marca	71
Tabla 19.	Tabla del Municipio	72
Tabla 20.	Tabla del Nivel de Estudios	72
Tabla 21.	Tabla del Paquete	72



Tabla 22.	Tabla del Pedido	72
Tabla 23.	Tabla del Producto	72
Tabla 24.	Tabla del Proveedor	73





INTRODUCCIÓN

El presente trabajo es una aplicación web que muestra una tienda de abarrotes virtual en 3D¹ (TAV-Hidalgo), la cual permite al usuario realizar sus compras vía Internet.

Los usuarios podrán interactuar con TAV-Hidalgo de dos formas, siendo *usuarios generales o clientes registrados*.

El usuario al realizar un recorrido por el mundo virtual y elegir un producto, visualizará sus características y lo podrá agregar al carrito de compra, esto se logrará mediante la conexión del mundo virtual con la base de datos. A través de esta interfaz el usuario podrá realizar un pedido, y dependiendo si es un *cliente registrado* tendrá que *autenticarse* para posteriormente *confirmar* su *pedido* e *imprimir* su *nota de compra*, que se generará de forma dinámica; en caso de ser un *usuario general* tendrá que *registrarse* y *autenticarse* para posteriormente *confirmar* su *pedido* e *imprimir* su *nota de compra*.

Esta aplicación contará con una interfaz controlada por la *administración*, en la cual se podrán dar de *alta*, *baja*, *modificar* y *visualizar los empleados, productos, proveedores y clientes*; de igual forma se tendrá un *control de inventarios* en el que se podrá calcular el costo promedio de los productos, y permitirá tener un control sobre las *compras y ventas* que se van realizando; además permitirá administrar los *pedidos*.

En el capítulo 1, se presenta la definición de *comercio electrónico*, así como sus características, además de algunas aplicaciones existentes de tiendas virtuales de abarrotes y ciertos *software's* que permiten generar tiendas virtuales.

¹ 3D. Tercera Dimensión.



En el capítulo 2, se muestra una comparación entre las diferentes metodologías para el desarrollo de sitios web.

En el capítulo 3, se describen las tecnologías utilizadas en el desarrollo de TAV-Hidalgo.

En el capítulo 4, se detallan las características, diseño, desarrollo e implementación de TAV-Hidalgo.

En el capítulo 5, se describen las conclusiones y los trabajos futuros de TAV-Hidalgo.

I. Objetivo General

Elaborar un sistema para una tienda de abarrotes virtual en 3D vía web en donde un cliente pueda solicitar sus productos y efectuar sus compras, además de tener un control administrativo de la tienda.

II. Objetivos Específicos

- ◆ Instalar un servidor web y de correo electrónico.
- ◆ Construir un mundo virtual mediante el cual interactuará el cliente, para que solicite y/o adquiera los productos.
- ◆ Desarrollar una base de datos para el control de clientes, empleados, proveedores, productos, pedidos e inventarios de la tienda de abarrotes virtual (Sistema de Punto de Venta).
- ◆ Desarrollar la interfaz para conectar la base de datos con el mundo virtual.



- ◆ Implementar la tienda de abarrotes virtual en 3D en una página web.

III. Justificación

En la actualidad las tiendas virtuales están tomando auge y los investigadores en la materia consideran que el comercio electrónico día con día tendrá una mejor aceptación e importancia, motivo por el cual los empresarios deben buscar la manera de alojarse en la red.

Esta investigación da a conocer diferentes aspectos sobre el nuevo software que se está utilizando para el desarrollo de Tiendas Virtuales, Mundos Virtuales, Bases de Datos y los Lenguajes de Programación para manipular esta última.

Igualmente, guiará a las personas interesadas en la creación de su propia tienda virtual en 3D, suministrándoles los procedimientos necesarios de acuerdo a las expectativas de cada quien.

IV. Alcances

El propósito de la tienda de abarrotes virtual en 3D, es dar a conocer esta idea innovadora, además de implementarla como una incubadora en la UAEH, con la idea de que posteriormente pueda ir creciendo y que después existan otras de diferentes géneros.

De cierta manera, el sistema de pagos con tarjeta de crédito presenta ciertas limitaciones o inconvenientes debido a que muchos usuarios de este modo de pago desconfían, con sobrada razón, del mal uso que pudiera hacerse de su información confidencial al ser capturada por los crackers que pululan en el ciberespacio. Sin



embargo, hoy en día se están implementando nuevas tecnologías para salvaguardar estos datos.

Desde el punto de vista computacional se pretende conectar un mundo virtual escrito en VRML con una base de datos desarrollada en MySQL controlada con PHP.

La aplicación debe contar con un carrito de compra que permita ir incrementando los productos con la posibilidad de dar de baja algunos de estos que ya no se quieran comprar; así como la posibilidad de imprimir el pedido o guardarlo en forma de archivo PDF.

V. Limitaciones

TAV-Hidalgo hasta este momento sólo cuenta con algunos productos virtualizados, los cuales son suficientes para mostrar la funcionalidad de la aplicación; no es un centro comercial completo.

En esta etapa no se incorpora ningún método de encriptación de datos, además de que no se tiene un sistema que permita realizar pagos con tarjeta de crédito.

Por el momento sólo se podrá optar por el sistema de pago en efectivo en el momento de la entrega de la mercancía a domicilio.

Para hacer uso del sistema es necesario que el usuario tenga conocimientos mínimos en el uso de la computadora, acceso a Internet, y manejo de ventanas

Por lo que se refiere al acceso a la tienda de abarrotes virtual en 3D, podría ser de dos tipos: los clientes que cuenten con una conexión directa a Internet y los que



conociendo o sabiendo entrar a la red pero no disponen de ella, acudirían a los llamados café Internet para hacer sus pedidos.

De la misma forma la entrega de la mercancía estará limitada a una ciudad en específico, Pachuca. Además de que la mercancía deberá ser entregada en un máximo de dos horas.



ESTADO DEL ARTE

Internet, en la actualidad, es el medio por excelencia utilizado en las transacciones comerciales. Por tanto, las empresas que deseen incrementar su potencial de comercialización, forzosamente tendrán que buscar la posibilidad de acceder a la red, puesto que de lo contrario, paulatinamente se irán rezagando hasta quedar fuera de la preferencia de los consumidores y, al mismo tiempo, fuera del mercado por no tener capacidad para luchar comercialmente contra la competencia [27], [36].

El *comercio electrónico*, en la actualidad, se ha difundido a través de *tiendas virtuales*, que tienen la capacidad de ser tiendas físicas pero sin estructura local, personal de ventas, cuentas de luz, agua, etc.

En definitiva, el *comercio electrónico* no es más que una consecuencia de la evolución actual en materia de comunicación y transferencia electrónica de datos. Su éxito dependerá estrechamente de la confianza que le dispensen los usuarios, así como de la seguridad que se logre imbuir a estos en el uso de dicho servicio [29], [31].

1.1 Comercio Electrónico

Se puede decir que el *comercio electrónico* es una actividad comercial que involucra todas aquellas transacciones realizadas sobre una red mediada por computadoras en la que se intercambian bienes y servicios a través de un convenio celebrado entre comprador y vendedor.

Algunos autores definen al *comercio electrónico* como la actividad de hacer negocios electrónicamente, de tal forma que incluye transacciones de bienes físicos y



de bienes intangibles como información; esto involucra diversas actividades como la negociación electrónica de bienes y servicios, el envío de componentes digitales en línea, la transferencia electrónica de fondos, la negociación de segmentos electrónicos, los conocimientos de embarque electrónicos, las acciones comerciales, el diseño e ingeniería cooperativa, fuentes en línea, compras, mercado directo al consumidor y servicios post-venta [27], [36].

El *comercio electrónico* se caracteriza por que:

- ◆ Las operaciones se realizan por vía electrónica o digital.
- ◆ Se prescinde del lugar donde se encuentran las partes o agentes.
- ◆ Se reducen drásticamente los intermediarios.
- ◆ Las transacciones se hacen efectivas con mayor rapidez.

Internet se ha convertido para el *comercio electrónico* en el instrumento más versátil debido a su gran capacidad basada en la posibilidad de transmitir en forma simultánea sonido, imágenes y textos alrededor del mundo. Sin embargo, para llevar a cabo esta actividad comercial por Internet, es necesario contar con una infraestructura de telecomunicaciones capaz de proveer la suficiente capacidad de transporte para el flujo normal de información que el sistema requiere, y su acceso sólo será posible con los recursos específicos como equipo de cómputo, modems, software de acceso y un proveedor del servicio de Internet [29], [31].

Por otra parte, el desarrollo del *comercio electrónico* se ha visto frenado por la falta de experiencias previas de los consumidores, en algunos casos con reservas a realizar su primera operación. Adicionalmente, existen otros obstáculos importantes al comercio electrónico en el mundo, entre los que cabe citar el temor de los



consumidores a la revelación de datos confidenciales o la indefinición de la legislación aplicable [29].

1.2 Concepto de Tienda Virtual

Es un sistema que permite a los clientes de un comercio realizar pedidos en línea desde cualquier lugar del planeta a través de Internet. Las tiendas virtuales cuentan con ciertos elementos comunes como el *catálogo de productos*, el *carrito de compra* y el *registro de clientes*. Las tiendas virtuales, además, permiten que el cliente realice el pago por su compra mediante una tarjeta de crédito y el comercio reciba el dinero directamente en su cuenta bancaria a través de sistemas de pago electrónico [28], [38], [47].

1.3 Características de una Tienda Virtual

1.3.1 El Local

El equivalente del local físico es ahora un conjunto de páginas web al alcance de todos los consumidores. La ubicación es determinante para el éxito. La tienda virtual puede ubicarse en un local propio o dominio en Internet, por ejemplo, <http://www.mitienda.com/>, o bien ubicarse dentro de alguna galería o centro comercial, por ejemplo: <http://digitaldreams.mujeresdeempresa.com>.

Las iniciativas a este respecto han evolucionado tanto, que incluso existen proveedores actualmente que ofrecen herramientas que permiten construir la tienda virtual desde la computadora del cliente. Y eso sin mencionar las nuevas estrategias de mercadotecnia y marketing para asegurar el éxito [38], [53].



1.3.2 El Escaparate

Las páginas web que sirven a los visitantes son el escaparate y las estanterías de los productos. 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 [38], [53].

1.3.3 Los Productos

Son la parte esencial de una tienda virtual, sobre los que giran el desarrollo, la estructura y el éxito del negocio. Los cuales se busca estén siempre al alcance del cliente, cumpliendo con la calidad, el precio e incluso los gustos que para el consumidor son tan importantes.

La forma más sencilla de exhibir los productos en una tienda virtual, es mediante el uso de páginas web estáticas que se modifican conforme se altera alguna información sobre el producto. Sin embargo, para una tienda más profesional y más ambiciosa, es mejor confiar la gestión del catálogo de productos a algún paquete de software especializado, aunque también existe la posibilidad de desarrollar internamente el software [38], [53].

1.3.4 El Proceso de Venta

El estudio del proceso de venta en la tienda, el recorrido libre o guiado por el local, la presencia y actitud de los vendedores, etc., son aspectos fundamentales en cualquier comercio. Por lo que se deben conocer y dominar a la perfección [38], [53].



1.3.5 El Carrito de Compra

Es un software que asiste al cliente en su recorrido virtual por la tienda, anotando los productos que él va indicando, calculando su importe final e incluso permitiéndole devolver productos a las estanterías.

La integración del carrito de compra y el software encargado de presentar los productos es un factor clave para conseguir una buena experiencia de compra por parte del cliente. Mientras mayor sea la flexibilidad con la que el cliente pueda interactuar y la facilidad de uso de estas utilidades, mayor será la satisfacción de los compradores [53].

1.3.6 Los Medios de Pago

Los medios de pago en Internet suscitan discusiones acerca de su fiabilidad y su seguridad. Por lo que es conveniente detallar la forma en cómo se cobrará al cliente y sobretodo garantizarle en todo momento la seguridad del proceso. Las políticas de seguridad y de confidencialidad de los datos y transacciones de los clientes deben ser claras y coherentes.

El *comercio electrónico*, como el tradicional, se basa en la confianza entre cliente y vendedor. Siendo el cobro por tarjeta de crédito el medio principal de transacción en la mayoría de las tiendas virtuales. Aunque el ofrecer diversas posibilidades de pago a los clientes para que decidan cuál prefieren es una buena idea [53].

1.3.7 El Servicio en la Tienda

La necesidad de prestar ayuda, al instante, es fundamental para asistir al cliente en su proceso de compra. Incluso existen tiendas virtuales en las que es posible obtener la asistencia de un vendedor humano por voz o video. Comprender las distintas



necesidades de los clientes es fundamental para el éxito de la tienda virtual. Incluido el papel que juegan los sistemas de software implementados en la tienda.

El análisis detallado del recorrido de los clientes por la tienda, el saber porque productos se inclinan, en qué orden, por cuánto tiempo y qué información parecen buscar, constituye un elemento decisivo a la hora de mejorar y adaptar la tienda a las necesidades de los clientes [53].

1.3.8 La Entrega del Material

Otro de los problemas que enfrentan las tiendas virtuales es el de la entrega del material. En el caso de las tiendas virtuales de alimentos o productos perecederos, se hace evidente porque un retraso en la entrega puede convertirse en algo desagradable para el cliente. Para el caso de entregas de pequeñas mercancías con un costo de transporte superior al mismo producto, provoca de inmediato que los clientes desistan. El servicio de entrega por lo regular estará en las manos de empresas terceras. La facilidad de compra por parte de clientes desde el extranjero siempre estará anulada por la dificultad de entrega del producto [53].

1.3.9 La Tras-tienda Virtual

Es necesario desenvolverse con facilidad por la trastienda, gestionando fácil y correctamente el catálogo de productos [53].

1.3.10 Procesos Internos

El software de la tienda virtual proporciona capacidades de gestión de mercancías, almacenes, proceso de pedidos, facturación, etc. Sin dejar de interactuar con el resto del sistema de información de la empresa. El cual mediante la conexión con



los proveedores de mercancías y los servicios de logística que proporcionan información puntual y exacta a los clientes acerca de sus pedidos, lo hace ser más competitivo. Es recomendable entonces que el paso de datos a contabilidad, control de almacén, facturación y cuentas por cobrar esté automatizado en la medida de lo posible [53].

1.3.11 Información de Marketing

La información obtenida desde la tienda virtual es una fuente necesaria para conocer los deseos y necesidades de los clientes y adaptar o crear productos y servicios que atiendan esos requerimientos. Una buena base de datos generada a partir de una buena tienda virtual puede obtener verdaderos patrones de comportamiento y de compra de individuos y grupos.

La posibilidad de ofrecer productos o servicios adicionales aumenta el potencial del negocio, pero también la complejidad de la tienda virtual. Los mailings personalizados, las ofertas puntuales, etc., son algunas de las posibilidades de una buena gestión de la información generada por la tienda virtual [53].

1.4 Elementos que Conforman una Tienda Virtual

La estructura básica de las tiendas virtuales no presenta gran complejidad y la mayoría de ellas coinciden con un esquema que incluye los siguientes aspectos: *Inicio*, *Instrucciones*, *Catálogo de Productos*, *Términos y Condiciones*, *Cobertura*, *Sugerencias*.

- ◆ **Inicio.** En este apartado de la tienda virtual se muestra a los usuarios la misión y la visión de la empresa, además de una breve reseña de cómo es que surge la tienda virtual.



- ◆ **Instrucciones.** En esta parte se muestra al usuario los procedimientos que debe seguir para realizar una compra en la tienda virtual.
- ◆ **Catálogo de Productos.** En esta sección se agrupan los productos de acuerdo a la categoría a la que pertenecen; esto se realiza con el fin de que el cliente encuentre de una manera más sencilla los productos que está buscando. Por lo general los catálogos de todas las tiendas virtuales se muestran en la barra de la izquierda, al navegarlo los clientes podrán encontrar los productos y precios.
- ◆ **Términos y Condiciones.** En este apartado se da a conocer a los clientes los términos y las condiciones sobre la forma en que opera la tienda virtual.
- ◆ **Cobertura.** En este apartado se muestran todos los lugares en donde se puede realizar la entrega de los productos a domicilio.
- ◆ **Sugerencias.** En esta sección el cliente podrá enviar sus sugerencias sobre el servicio y sobre los productos que quisiera que se pusieran a la venta en la tienda virtual.

Por lo general las tiendas virtuales muestran una interfaz gráfica de la variedad de productos que están ofreciendo a la venta; una diversidad de botones para ir navegando en la tienda virtual y algunas ofrecen ciertos sonidos relacionados con los de una tienda; en cuanto a las animaciones casi no presentan [53].

1.5 Aplicaciones Existentes

Después de realizar una visita por diferentes páginas de tiendas virtuales, en el ámbito Nacional e Internacional, se encontró que existe un gran auge en el comercio electrónico.



Las principales herramientas que se están utilizando para el desarrollo de las tiendas virtuales son **ASP²**, **JSP³**, **PHP⁴**, **MySQL**, **SQL⁵ Server**, **HTML⁶**, **Front Page**, **Flash**, entre otras; todas enfocadas a la búsqueda de la eficiencia y competitividad. Así, se tiene que muchas de estas tiendas virtuales se enfocan más a la interfaz gráfica, otras en la agilidad para atender al cliente y gran parte de ellas ofrecen cierta seguridad en las transacciones del cliente, pero siempre buscando integrar la mayoría de los aspectos anteriores.

1.5.1 Trico

<http://www.trico.com.mx/asp/> (ver figura 1).

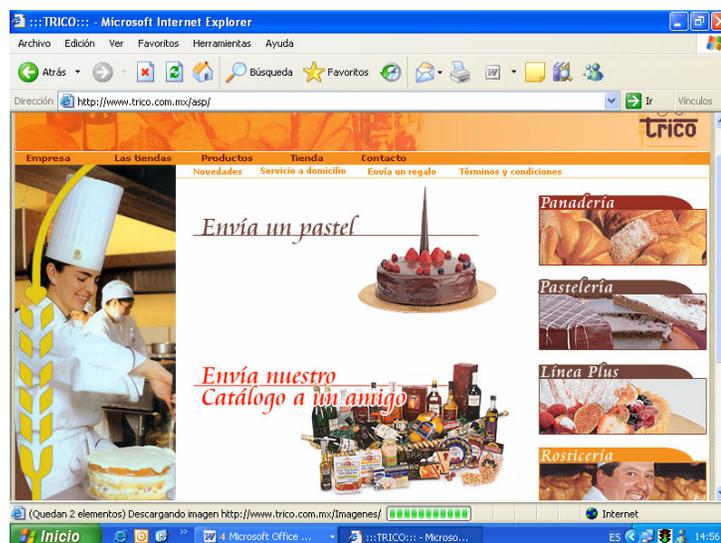


Figura 1. Pantalla Principal de Trico.

Descripción:

Tienda de abarrotes virtual, especializada en pastelería, panadería, línea plus, rosticería, baguettería, canastas, charcutería, vinos, latería, licores y regalos. Hace la

² ASP. Active Page Server, (Página Activa en el Servidor).

³ JSP. Java Server Page, (Página Java en el Servidor).

⁴ PHP. Acrónimo de Hyper Text Preprocessor.

⁵ SQL. Structured Query Language, (Lenguaje de Consulta Estructurado).

⁶ HTML. Hiper Text Markup Language, (Lenguaje de Marcación de Hipertexto).



diferencia al dar a sus clientes servicio, atención personalizada y la más alta calidad; manejando diferentes y exclusivas líneas de productos, estableciendo de forma vanguardista en el mercado nacional el concepto de gourmet y delicatessen.

Forma de compra:

En la página principal existe un menú con cinco categorías: *Empresa*, las *Tiendas*, *Productos*, *Tienda* y *Contactos*. Al posicionarse en él se despliega un listado con diferentes opciones.

Al posicionarse en la categoría de "*Productos*", muestra los productos que ofrece. Al dar click en alguna especialidad, se despliega una subdivisión de los productos. Al elegir una subdivisión se muestran los productos con una fotografía, la presentación y su clave.

Al dar click en la opción de "*Seleccionar*" se muestran los datos anteriores y el precio del producto. Al volver a dar click en la opción de "*Seleccionar*" aparece el carrito de compra, en el cual se establece la cantidad, el producto, la clave, la presentación, el precio y el subtotal. Y se muestran cuatro opciones: "*Seguir Comprando*", "*Vaciar Pedido*", "*Actualizar Pedido*" y "*Continuar*".

Para salir y pasar a la caja se oprime el botón "*Continuar*". Al dar click en el botón "*Continuar*" se muestra un formulario para ingresar el *correo electrónico* y la *contraseña*. Si no se está registrado aun, se oprime el botón de "*Nuevo registro*" para llenar la forma de registro de nuevo cliente. Si ya se ingresó el *nombre de usuario* y el *password* se oprime el botón de "*Continuar*" para confirmar el registro.

Para terminar se oprime el botón "*Continuar*" que muestra el monto total del pedido y la cuenta de banco en la cual se debe hacer el depósito.

**Tecnologías utilizadas:**

ASP, HTML y Flash.

Seguridad:

En ninguna parte de la tienda se especifica cómo se protege al usuario.

Cobertura:

Distrito Federal, Michoacán, Querétaro, Hidalgo, Veracruz [48].

1.5.2 La Playa

<http://www.superlaplaya.com.mx/> (ver figura 2).



Figura 2. Pantalla Principal de La Playa.

Descripción:

Tienda virtual elaborada con una buena interfaz gráfica, especializada totalmente en vinos y licores, incluye un catálogo por producto con fotografías, detalles de productos, imagen corporativa, historia, etc.

**Forma de compra:**

La página inicial de entrada se vincula a un índice de categorías, al acceder a cada categoría se tiene la oportunidad de navegar de diferentes maneras, ya sea con una imagen estática o con un slider. Una vez seleccionados los productos se accesa al apartado de pedidos. Si se es *usuario registrado* se introduce el *nombre de usuario* y la *contraseña*, de otro modo se debe registrar como nuevo usuario, sino se cuenta con datos reales es imposible acceder al área de pedidos.

Tecnologías utilizadas:

Flash y ASP.

Seguridad:

Desconocida.

Cobertura:

Guadalajara, México [37].

1.5.3 Sumercado.com

<http://www.sumercado.com/Lima/abarrotos.htm> (ver figura 3).



Figura 3. Pantalla Principal de Sumercado.com.

Descripción:

Sumercado.com, es una tienda virtual, donde se puede adquirir una gama de productos y servicios de primera calidad, los mismos que serán entregados a las personas que desee, en la comodidad de su hogar o en el lugar que se designe.

Forma de compra:

En la parte izquierda de la página principal se encuentra un panel con las categorías y subcategorías, estas contienen los productos disponibles en la tienda. Al seleccionar un producto se muestra el precio, la fotografía del producto y un botón para comprarlo, al oprimirlo aparece la orden de confirmación y el cálculo del monto de los productos. Sino es usuario registrado, se requiere llenar un formulario, de lo contrario se accesa con la contraseña de usuario registrado, dentro de los datos personales se requiere el estado y la provincia de Perú, si ambos son incorrectos es imposible continuar.

Tecnologías utilizadas:

HTML, JavaScript y ASP.



Seguridad:

Cuenta con un Certificado Digital, que garantiza la confidencialidad e integridad en la transacción realizada a través de la tienda.

Cobertura:

Lima Metropolitana o Provincias [45].

1.5.4 My Brands

<http://www.mybrandsinc.com/ShopOnline/catalog.asp> (ver figura 4).



Figura 4. Pantalla Principal de My Brands.

Descripción:

Tienda virtual especializada en productos difíciles de encontrar, y productos que por lo regular se encuentran en tiendas de abarrotes y almacenes de especialidad. Única en su tipo por ofrecer una relación directa con los fabricantes, lo cual permite ofrecer grandes promociones y ofertas especiales y a su vez cuando el producto no



está disponible, da la oportunidad a los clientes de opinar sobre qué productos son de su preferencia.

Forma de compra:

La página principal de My Brands cuenta con un menú en la parte superior con los diferentes apartados, cada uno de los cuales se divide en categorías, en las que se puede seleccionar el producto, la marca y presentación deseada, al elegir el producto aparece su fotografía, sus detalles y el botón para agregarlo al carrito; una vez agregado en el carrito, el siguiente paso es confirmar el producto en el carrito para que sea enviado a la página de pedidos, es necesario ser usuario registrado, de lo contrario se debe llenar un formulario con datos personales, que incluyen un estado de Estados Unidos y su código postal, si son incorrectos no se puede continuar con el registro.

Tecnologías utilizadas:

ASP y JavaScript.

Seguridad:

Esta tienda virtual usa un certificado de seguridad que encripta la información provista por el usuario. La información no puede ser leída en el trayecto y es guardada en una base de datos segura.

Cobertura:

Estados Unidos [39].



1.5.5 El Corte Inglés Supermercado

<http://supermercado.elcorteingles.es/secciones/entrada/noidentificado.asp> (ver figura 5).



Figura 5. Pantalla Principal de El Corte Inglés Supermercado.

Descripción:

Tienda virtual con más de 25,000 productos de alimentación, limpieza e higiene personal que entrega a domicilio las 24 horas en toda España. Con una interfaz sencilla y agradable que permite a los usuarios realizar sus compras en línea en tres sencillos pasos.

Forma de compra:

En la página principal se eligen los artículos por sección, familia y variedad, se indica la marca y el tipo de producto y aparece un listado con todos los artículos disponibles. Para añadirlos al carrito de compra, se tendrá que indicar el número de unidades que se desea, para confirmar el producto, se indica como se desea pagarlo y se elige una fecha de entrega. Para poder realizar la compra es necesario estar registrado como cliente. Y si no es así, es necesario registrar los datos solicitados.

**Tecnologías utilizadas:**

JavaScript y ASP.

Seguridad:

El envío de datos se realiza de forma segura mediante el empleo de un sistema avanzado de seguridad SSL (Secure Socket Layers).

Cobertura:

España [32].

1.6 Software para el Desarrollo de Tiendas Virtuales

1.6.1 Cube Stores

Es un sistema avanzado para la creación de catálogos y tiendas en línea, desarrollado en Java. Cube Stores integra en un sistema distribuido la funcionalidad de diferentes servidores: HTTP⁷, de aplicaciones, de bases de datos.

La arquitectura del sistema le permite interconectarse con sitios web de forma independiente a la plataforma en que se encuentren (WNT, UNIX, LINUX, MS W2000, etc.)

Las tiendas virtuales incluyen todos los elementos básicos de un sistema de comercio electrónico: el *catálogo de productos*, *carrito de compras* y *registro de usuarios*. Cuenta con vínculos para cada producto, así como para la página del catálogo principal, permitiendo opciones de compra directa así como vistas generales de líneas

⁷ HTTP. Hyper Text Transfer Protocol, (Protocolo de Transferencia de Hipertexto).



completas de productos. Permite la búsqueda de productos por nombre o descripción. Calcula el costo de envío de acuerdo al área geográfica.

Cada tienda cuenta con su propio sistema de administración, el cual permite al administrador actualizar los detalles de productos y servicios de la tienda vía Internet. El sistema de administración permite:

- ◆ Insertar nuevos productos en el catálogo de la tienda.
- ◆ Organizar productos o servicios en categorías o subcategorías.
- ◆ Resaltar productos en oferta.
- ◆ Configurar nuevas zonas de entrega y asignar diferentes costos a cada una.
- ◆ Manejar el concepto de productos temporalmente sin stock, incluyendo la capacidad de sacarlo de circulación y luego reinsertarlo en el catálogo de productos disponibles.
- ◆ Manejar el concepto de productos complementarios, incluyendo la capacidad de que el cliente vea tanto el producto principal como los complementarios que pueden ser añadidos a la compra.
- ◆ Enviar al administrador un e-mail informándole que una transacción ha sido realizada en su tienda.

La consola de administración de Cube Stores ofrece un amplio rango de reportes para el seguimiento de las ventas.



La seguridad de las transacciones es un punto crucial en el comercio electrónico. La encriptación SSL⁸ mantiene la información del cliente segura a lo largo de todo el proceso. Para el procesamiento de tarjetas de crédito Cube Stores trabaja con los principales proveedores de soluciones de pago del mundo [30].

1.2.2 AB-Shop

Es un software de creación de tiendas virtuales independiente del sistema operativo y del servidor de Internet.

AB-Shop no necesita ningún software adicional, debido a que permite crear en menos de una hora un completo sistema de comercio virtual, sin programación ni herramientas adicionales.

AB-Shop permite una compra segura bajo el protocolo SSL, situándose al nivel de grandes desarrollos de software para comercio electrónico y elimina la necesidad de realizar peticiones constantes al servidor, aumentando drásticamente la velocidad de carga de las páginas. Es 100% configurable con la posibilidad de crear e importar plantillas de terceros y permite realizar el mantenimiento en línea, ahorrando tiempo y elevados costos de teléfono (ver figura 6).

Funcionamiento:

AB-Shop dispone de 2 módulos principales. El modulo AB-Shop y el Diseñador de tiendas.

⁸ SSL. Secure Socket Layers.



El modulo AB-Shop es utilizado para las labores de configuración y mantenimiento de la tienda virtual; desde él, los usuarios pueden definir cómo va a funcionar la tienda virtual y realizar las labores de mantenimiento de la tienda virtual, altas, bajas, modificaciones, etc.

El diseñador, es el módulo encargado de realizar la presentación gráfica de la tienda virtual. Desde este módulo el usuario puede, en un entorno 100% visual, diseñar el aspecto que tendrá su tienda virtual. Su funcionamiento es similar a editores avanzados de páginas HTML, pero especializado en la creación de componentes para la tienda virtual, sin ser necesarios conocimientos gráficos ni de programación HTML.



Figura 6. Pantalla Principal de AB-Shop.

AB-Shop incluye, además, la edición en modo HTML para usuarios avanzados que deseen trabajar a fondo en el diseño gráfico de la tienda virtual sin limitaciones. Pudiendo incluso importar páginas HTML desarrolladas en cualquier tipo de editor de páginas HTML.

**Características:**

- ◆ Ilimitados artículos, agrupaciones de artículos, campos para cada artículo, opciones y subopciones para cada artículo.
- ◆ Ilimitadas formas de pago, de envío, monedas y conversión de cambio automático.
- ◆ Generación de gastos de envío según país destino y/o forma de envío.
- ◆ Cargo de impuestos adicionales según país de destino.
- ◆ Opción de validación a través de Banco Online.
- ◆ Buscador de productos.
- ◆ Validación de pago a través de páginas seguras SSL y Banco On-line.
- ◆ Asistentes para la creación de botones, banners, imágenes, etc.
- ◆ Ficha de artículo configurable.
- ◆ Formulario de compra configurable.
- ◆ Generación de tiendas en múltiples idiomas.
- ◆ Importación de artículos desde otras bases de datos.
- ◆ Copia de seguridad.
- ◆ Plantillas de diseños incluidas.
- ◆ Mantenimiento Off-line.
- ◆ Publicación automática de la tienda.



◆ Cliente FTP⁹ incorporado.

Todas estas características hacen de AB-shop una de las herramientas más potentes del mercado para la creación de tiendas virtuales, con una relación calidad-precio inmejorable [24].

1.6.3 e-Shop

e-Shop es una pasarela que va a permitir mostrar y vender productos a los clientes en Internet, aumentando las posibilidades de venta. Es una solución para la creación de una forma sencilla y rápida de catálogos y tiendas en Internet convirtiéndole así en la plataforma de gestión de webs transaccionales más sólida, fiable y escalable del mercado (ver figura 7).

e-Shop ofrece un producto para catálogos y tiendas virtuales fácil de instalar que funciona en dos horas.

e-Shop no sólo le ofrece una solución para una tienda o catálogo virtual, e-Shop es una herramienta de desarrollo para profesionales que deseen crear una tienda o catálogo virtual, debido a que incluye el lenguaje para el desarrollo orientado a profesionales, “shoptalk”, que permite cambiar la imagen y la conceptualización de la página web de los clientes. El profesional diseña y e-Shop hace el resto.

⁹ FTP. File Transfer Protocol, (Protocolo de Transferencia de Archivos).

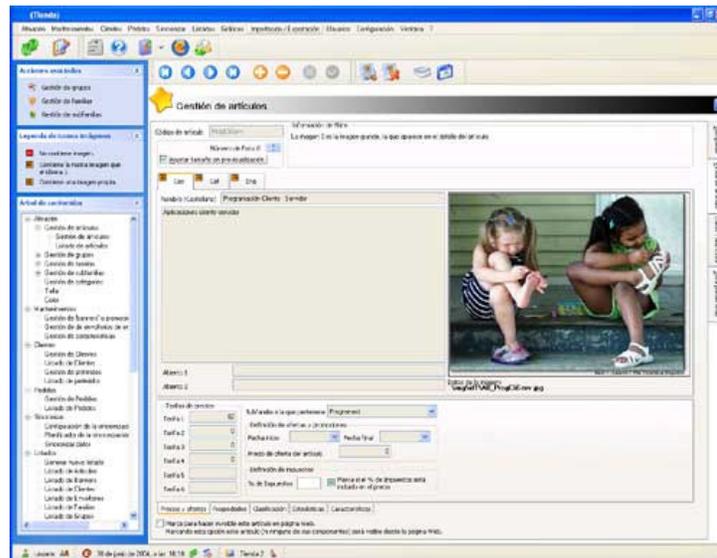


Figura 7. Pantalla Principal de e-Shop.

Funcionamiento:

El funcionamiento de e-Shop es muy sencillo. Simplemente deberá instalar el software e-Shop Cliente, que se encontrará en la zona de descarga de productos de servicios web, indicar su licencia de uso y empezar a funcionar.

e-Shop cliente permitirá administrar y gestionar todos los artículos y clientes de la tienda o catálogo virtual de una forma muy fácil e intuitiva.

Características:

- ◆ Gestión de la tienda o catálogo virtual.
- ◆ Sistema integrado de plataforma de servidores.
- ◆ Administración remota de todos los componentes de la plataforma.
- ◆ Integración con 42 entidades bancarias.
- ◆ Sistema multidioma, hasta 9 idiomas.



- ◆ Sistema multidivisa, tantas monedas como desee.

- ◆ Gestión de usuario, en entornos distribuidos o empresariales [34].

1.6.4 Neostore

Es una solución para e-commerce que ofrece la habilidad de construir un negocio online exitoso sin tener experiencia en diseño web o conocimientos de programación. Neostore ofrece todas las características y funcionalidades que son necesarias para desarrollar y administrar una tienda online, sin necesidad de programación. Entre las características se incluye la administración a través de la web, procesamiento automático de imágenes, manejo de usuarios registrados, etc.

El sitio web puede ser de dos tipos:

- a) El sitio *corporativo ordinario* que funciona sólo como una presentación virtual de una compañía y productos, que en la mayoría de casos nadie realmente usa y básicamente sólo cuesta dinero tenerlo.

- b) Un sitio web de *e-commerce* que es usado activamente por los clientes y por el administrador para trabajar más eficientemente, para incrementar sus negocios y ganancias por completo.

La versión de Neostore tiene todas las funciones necesarias para poner una tienda online en marcha y además incluirá todas las nuevas funciones que se vayan creando.

Todo lo que se necesita hacer es solicitar el servicio, configurar la apariencia de la tienda e insertar los productos que se van a vender (tantos como se desee).



Automáticamente estará listo para tomar pedidos en línea. Los resultados serán profesionales.

Neostore está pensado para pequeñas y medianas empresas. Es simple de implementar a través del un manual, con los videos explicativos o con la ayuda online.

Funcionamiento:

Sólo se necesita contratar Neostore y tan pronto como la cuenta esté lista, puede comenzar a crear una tienda en línea. La creación consiste de 2 pasos. Se debe comenzar eligiendo el diseño y luego se debe insertar la información de los productos, categorías y otras páginas. Antes de lanzar la tienda se debe revisar todo, pequeños detalles y realizar pruebas. Siempre se tiene acceso al soporte técnico que es gratis por email y teléfono (ver figura 8) [40].



Figura 8. Pantallas de la Sección Pública y del Panel de Control.

Características:

- ◆ Panel de control web.
- ◆ No se requieren conocimientos de HTML ni de diseño.
- ◆ Procesamiento de imágenes de productos automático.



- ◆ Control de inventario.
- ◆ Acepta diversas monedas y cotizaciones para cálculo de precios.
- ◆ Hosting confiable.
- ◆ Cuentas de e-mail.
- ◆ Acceso a cuentas de mail vía web.
- ◆ Backup periódicos.
- ◆ Categorías y subcategorías ilimitadas.
- ◆ Alta de banners promocionales.
- ◆ Manejo de clientes y pedidos.
- ◆ Altamente flexible y personalizable.
- ◆ Actualizaciones de software gratis.

1.6.5 SACE (Sistema de Administración de Comercio Electrónico)

Es una herramienta que permite la implantación del comercio electrónico de una manera sencilla, cómoda y personalizada. Su ventaja radica en que no se necesitan conocimientos de programación, protocolos e implantación de líneas seguras para la creación de un sitio web con el comercio electrónico implantado en sus variantes de carrito de compra con pago online y otras variantes de pago.



Esta herramienta es un complemento ideal para digitalizar de manera sencilla y segura los negocios tradicionales, incluye: TPV¹⁰ Virtual de Bancos, Gestión de la apariencia de la tienda virtual, Administración de: Pedidos, Almacén, Familias, Clientes, Mailings, Entorno, Transporte y un potente sistema de ayuda (ver figura 9).

Características:

- ◆ SACE se puede configurar con cualquiera de estas dos monedas: \$ USD y Euros.
- ◆ SACE entrega por defecto la secuencia lógica de compra, pues están predeterminadas y programadas todas las plantillas a mostrar, desde el carrito de compra hasta las páginas de errores, siendo configurables a la imagen corporativa.
- ◆ El sistema es independiente de la plataforma, hosting, navegador o computadora desde el que se gestione.
- ◆ Sólo hace falta que se disponga de un equipo con la capacidad necesaria para navegar por Internet, para tener acceso al SACE por la Red [44].



Figura 9. Empaque del Software SACE.

¹⁰ TPV. Terminal Punto de Venta.



METODOLOGÍA DE DESARROLLO

2.1 Metodologías para el Diseño de Sitios Web

Se puede considerar una *metodología* como un planteamiento paso a paso completo para el desarrollo de sistemas, que guían en el proceso de desarrollo e influyen en la calidad del producto final, proporcionando detalle tanto del conjunto de actividades a llevar a cabo como de los resultados que se deben obtener al final de cada una de ellas (conjunto completo de conceptos, modelos, reglas y directrices). Dependiendo del producto que se esté desarrollando, habrá de elegirse una metodología que resulte la más apropiada de entre todas las posibles, por eso no se puede hablar de la mejor metodología. Es más, metodologías diferentes pueden utilizarse en las diferentes fases del proceso de desarrollo, debido a que puede haber alguna desarrollada para cada problema específico.

Las metodologías tradicionales utilizadas en ingeniería del software se muestran inadecuadas para las aplicaciones web dado que generalmente no hacen referencia a aspectos tanto de conocimiento como de apariencia que son de especial relevancia en el desarrollo de aplicaciones web y aplicaciones hipertexto en general y que son los siguientes:

- ◆ En los proyectos hipertexto suelen participar profesionales de diferentes disciplinas.
- ◆ El diseño de este tipo de aplicaciones implica capturar y organizar la estructura de dominios complejos y hacerla clara y accesible para los usuarios.
- ◆ Se debe tener en cuenta que en el proceso de desarrollo de este tipo de aplicaciones se implican también aspectos más relacionados con el arte que con las ciencias.



- ◆ La necesidad de realizar prototipos y test intensivos con los usuarios es mucho más atribuida en el desarrollo de hipermedia que en el desarrollo de software tradicional dado que la tolerancia a fallos del usuario en las aplicaciones hipermedia es muy bajo.
- ◆ La fase de mantenimiento juega un papel más importante en el ciclo de vida de una aplicación hipermedia que en el de otros sistemas.

Actualmente existe una gran cantidad de metodologías que intentan ayudar en la sistematización de la construcción de sitios web, proporcionando mecanismos de abstracción que faciliten el desarrollo de estos sistemas.

Existen dos tendencias claras. La primera se basa en extender iniciativas orientadas al diseño hipermedial (navegacional), introduciendo expresividad para dotar de dinamismo a los sistemas. Estas metodologías aparecieron con el objetivo de construir aplicaciones hipermediales donde se unía el concepto de navegación con la multimedia, en sistemas estáticos (sin funcionalidad). Es por esto que la mayoría de estas metodologías están basadas en el Modelo Relacional clásico, o bien en extensiones de éste. Algunos ejemplos destacables de estas metodologías son HDM (Hypertext Desing Model) [42], RMM (Relationship Management Methodology) [42], RNA (Realtionship Navigational Analisys) [33], VHDM (View-Based Hypermedia Design Methodology) [42].

El otro grupo de metodologías se basan en la idea de extender los métodos de desarrollo orientados a aplicaciones dinámicas (con funcionalidad), tratando de introducir la semántica de la hipermedia como característica inherente a este nuevo tipo de sistemas software. Este tipo de metodologías tratan de introducir características navegacionales al modelo OO¹¹. En este grupo se pueden encontrar las metodologías

¹¹ OO. Orientado a Objetos.



EORM (Enhanced Object Relationship Model) [42], OOHDM (Object Oriented Hypermedia Design Method) [42], SOHDM (Scenario-Based Object-Oriented Hypermedia Design Methodology) [42], WSDM (Web Site Design Method) [42], UWE (UML-Based Web Engineering) [33], OO-Method (Object-Oriented Method) [43], OOWS (Object-Oriented Approach for Web Solutions Modeling) [43], [52].

A continuación en la tabla 1 se presenta una comparación entre las metodologías antes mencionadas, basada en los pasos que componen el proceso, la técnica de modelado, la representación gráfica, la notación para los modelos y la herramienta CASE que brinda soporte para su desarrollo; es importante señalar que las metodologías que se comparan a continuación son las más comunes.

Tabla 1. Comparación entre las Metodologías para el Diseño de Páginas Web.

Metodología	Proceso	Técnica de modelado	Representación gráfica	Notación	Herramienta de soporte
HDM	1. Desarrollo a largo plazo. 2. Desarrollo a corto plazo.	E-R	Diagrama E-R	E-R	
RMM	1. Diseño entidad-relación. 2. Diseño slice. 3. Diseño de navegación. 4. Diseño de protocolo de conversión. 5. Diseño de interfaz de usuario. 6. Diseño de comportamiento en tiempo de ejecución. 7. Prueba y construcción.	E-R	1. Diagrama E-R. 2. Diagrama slice. 3. Diagrama RMDM.	E-R propio	RMCase
EORM	1. Clases del entorno de desarrollo. 2. Composición del entorno de desarrollo. 3. Entorno de desarrollo de la interfaz de usuario.	OO	1. Diagrama de clases. 2. Diseño GUI17.	OMT	ONTOS Studio
OOHDM	1. Diseño conceptual. 2. Diseño navegacional. 3. Diseño abstracto de la interfaz de usuario. 4. Implementación.	OO	1. Diagrama de escenarios de actividad. 2. Diagrama navegacional, clases + contexto. 3. Diagrama de configuración de ADV + diagrama ADV.	OMT / UML propio ADV's	OOHDM-Web
SOHDM	1. Análisis del dominio. 2. Modelo orientado a objetos. 3. Diseño de la vista. 4. Diseño navegacional. 5. Diseño de la implementación. 6. Construcción.	Escenarios Vistas OO	1. Diagrama de escenarios de actividad. 2. Diagrama de estructura de clases. 3. Vista OO. 4. Esquema de enlace navegacional. 5. Esquema de páginas.	Propio	



Metodología	Proceso	Técnica de modelado	Representación gráfica	Notación	Herramienta de soporte
WSDM	1. Modelado del usuario. 2. Diseño conceptual. 3. Diseño de la implementación. 4. Implementación. Especificación del problema.	E-R OO	1. Diagramas de E-R o clase. 2. Capas de navegación.	E-R / OMT propio	
OOWS	1. Modelado de objetos. 2. Modelado dinámico. 3. Modelado funcional. 4. Modelado navegacional. 5. Modelado de la presentación. Desarrollo de la solución.	OO	1. Diagrama de clases. 2. Modelos dinámicos. 3. Modelos funcionales. 4. Modelo de navegación. 5. Modelo de presentación.	UML	

Después de hacer un análisis de las metodologías antes mencionadas se eligió la OOWS como metodología para el diseño del sitio web TAV-Hidalgo debido a que es la que más se adecua a los requerimientos de la aplicación.

2.1.1 OOWS (Object-Oriented Approach for Web Solutions Modeling)

Este método es una extensión del OO-Method con capacidades navegacionales y de presentación. OO-Method capta las propiedades funcionales del sistema que se consideran relevantes para construir una especificación textual OO y formal. Esta especificación formal OO constituye un repositorio de información de alto nivel del sistema [43], [52].

La especificación de los requisitos de usuario debe realizarse en la fase de modelado conceptual. Para modelar la navegación asociada al sistema deseado se propone un proceso de desarrollo de soluciones web con dos pasos principales: *Especificación del Problema y Desarrollo de la Solución*.

En la fase de **Especificación del Problema** se deben capturar las peculiaridades y el comportamiento que debe ofrecer el sistema para satisfacer los requisitos de usuario identificados. En este paso se incluye el conjunto de requisitos usando una aproximación de Casos de Uso [7] y posteriormente las actividades de modelado conceptual del sistema. En el *modelado conceptual*, las abstracciones que se



derivan del problema son especificadas en términos de clases y de su estructura, comportamiento y funcionalidad, construyendo los siguientes modelos: de *Objetos*, *Dinámico*, *Funcional*, *Navegacional* y de *Presentación*. Éstos describen la sociedad de objetos desde cinco puntos de vista diferentes usando un marco de trabajo OO bien definido:

- ◆ El **Modelo de Objetos** define la estructura y las relaciones estáticas entre clases identificadas en el dominio del problema.
- ◆ En el **Modelo Dinámico** se describen las posibles secuencias de servicios y los aspectos relacionados con la interacción entre objetos.
- ◆ El **Modelo Funcional** captura la semántica asociada a los cambios de estado entre los objetos motivados por la ocurrencia de eventos o servicios.
- ◆ El **Modelo de Navegación** define la semántica navegacional asociada a las clases de los objetos del modelo. Es en este modelo donde se explicita la navegación permitida en la aplicación para cada agente del sistema.
- ◆ El **Modelo de Presentación** captura los requisitos básicos de presentación de información, orientado hacia ambientes web. Está fuertemente basado en el modelo de navegación y permite definir, de una manera abstracta la estructura lógica de presentación de los objetos navegacionales en la interfaz de usuario.

En esta fase se realiza un estudio de los tipos de usuarios que pueden interactuar con el sistema, indicando qué visibilidad sobre el sistema tendrán (qué atributos y qué operaciones podrán ver y/o activar), cómo se podrán conectar (requerirán o no identificación), y se organizarán en jerarquías de especialización para potenciar el reúso en la especificación del sistema, facilitando así la tarea de modelado.



En la fase de **Desarrollo de la Solución** se propone una estrategia de generación de código basada en componentes para integrar la solución propuesta en ambientes web. En esta etapa se obtendrá una aplicación web, con una funcionalidad equivalente a la especificación inicial según una visión operativa [43], [52].

2.1.1.1 Notación Utilizada por OOWS

UML (Unified Modeling Language)

El Lenguaje Unificado de Modelado es un lenguaje estándar para escribir planos de software. UML puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software.

Diagramas en UML

Un diagrama es la representación gráfica de un conjunto de elementos, visualizando la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones). Los diagramas se dibujan para visualizar un sistema desde diferentes perspectivas, de forma que un diagrama es una proyección de un sistema. Para todos los sistemas un diagrama representa una vista resumida de los elementos que constituyen un sistema. UML incluye nueve diagramas:

1. **Diagrama de Clases.** Muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones, cubre la vista de diseño estática de un sistema.
2. **Diagrama de Objetos.** Muestra un conjunto de objetos y sus relaciones, cubre la vista de diseño estática o la vista de procesos estática de un sistema pero desde la perspectiva de casos reales o prototípicos.



3. **Diagrama de Casos de Uso.** Muestra un conjunto de Casos de Uso y actores (un tipo especial de clases) y sus relaciones. Estos diagramas son importantes en el modelado y organización del comportamiento de un sistema.
4. **Diagrama de Secuencia.** Muestra una interacción, que consta de un conjunto de objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos, cubre la vista dinámica de un sistema y resalta la ordenación temporal de los mensajes.
5. **Diagrama de Colaboración.** Es un diagrama de interacción que resalta la organización estructural de los objetos que envían y reciben mensajes.
6. **Diagrama de Estados.** Muestra una máquina de estados, que consta de estados, transiciones, eventos y actividades, cubren la vista dinámica de un sistema. Es importante en el modelado del comportamiento de una interfaz, una clase o una colaboración y resalta el comportamiento dirigido por eventos de un objeto.
7. **Diagrama de Actividades.** Es un tipo especial de diagrama de estados que muestra el flujo de actividades dentro de un sistema, cubre la vista dinámica de un sistema. Es importante a la hora de modelar el funcionamiento de un sistema y resalta el flujo de control entre objetos.
8. **Diagrama de Componentes.** Muestra la organización y las dependencias entre un conjunto de componentes, cubre la vista de implementación estática de un sistema.
9. **Diagrama de Despliegue.** Muestra la configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos, cubre la vista de despliegue estática de una arquitectura [3], [4].



MARCO TECNOLÓGICO

En la actualidad existen diferentes tecnologías que permiten el desarrollo de aplicaciones web como son:

- ◆ **Lenguajes de Programación Web:** ASP, Java, JavaScript, PHP, HTML, Flash.
- ◆ **Herramientas para el Diseño de Páginas Web:** FrontPage, ColdFusion, Dreamweaver.
- ◆ **Lenguajes de Desarrollo de Ambientes Virtuales:** VRML, OpenGL, Java 3D.
- ◆ **Herramientas de Desarrollo de Ambientes Virtuales:** Maya, 3D Studio Max, TrueSpace.
- ◆ **Servidores Web:** Apache, ISS (Internet Information Services o Server), Lighttpd.
- ◆ **Sistemas Gestores de Bases de Datos:** Oracle, Informix, MySQL, SQL Server.

Después de hacer un análisis de las tecnologías antes mencionadas se optó por elegir las que se describen a continuación para la construcción de TAV-Hidalgo.

3.1 Lenguajes de Programación Web

3.1.1 PHP

Es un lenguaje de programación utilizado mayormente para desarrollar servicios web; distribuido en forma gratuita, que permite interactuar con muchos sistemas de gestión de bases de datos [17].



Es un lenguaje de script incrustado dentro del HTML, la mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo [75].

PHP se ejecuta en el servidor, por lo cual puede acceder a los recursos del mismo, como puede ser el caso de una base de datos; por esta razón es independiente del navegador, no obstante para que las páginas PHP funcionen, el servidor en que se encuentran alojadas debe soportar PHP [22].

Características:

- ◆ Accede a gran número de gestores de bases de datos como son: Hyperwave, Informix, Microsoft SQL Server, mSQL, MySQL, PostgreSQL, entre otros.
- ◆ Brinda soporte para: acceso a servidores IMAP¹², envío de correo con SMTP¹³, acceso a servidores de FTP, acceso a SNMP¹⁴ para gestión de redes y equipos, generación dinámica de gráficos y documentos PDF, análisis de documentos XML¹⁵, corrector de ortografía, generación de datos en WDDX (Intercambio Web de Datos Distribuidos) [20].
- ◆ Permite desarrollar aplicaciones de escritorio tanto para los sistemas operativos basados en Unix, como para Windows y Mac OS X.

3.1.2 HTML

Acrónimo inglés de Hyper Text Markup Language (Lenguaje de Marcación de Hipertexto), es un lenguaje muy sencillo que permite crear hipertexto, es decir, texto presentado de forma estructurada y agradable, con enlaces (hyperlinks) que conducen

¹² IMAP. Internet Message Access Protocol, (Protocolo de Red de Acceso a Mensajes).

¹³ SMTP. Simple Mail Transfer Protocol, (Protocolo Simple de Transferencia de Correo).

¹⁴ SNMP. Simple Network Management Protocol, (Protocolo Simple de Administración de Redes).

¹⁵ XML. Extensible Markup Language, (Lenguaje de Marcado Extensible).



a otros documentos o fuentes de información relacionadas, y con inserciones multimedia (gráficos, video, audio). Este lenguaje es el que se utiliza para presentar información en el World Wide Web.

La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc.), así como los diferentes efectos que se quieren dar (cursiva, negrita o un gráfico determinado) y dejar que luego la presentación final de dicho hipertexto se realice por un programa especializado (como Mosaic o Netscape) [15], [18], [23], [46], [50].

3.1.3 FLASH

Es un programa de edición multimedia distribuido por Macromedia, que utiliza gráficos vectoriales e imágenes de mapa de bits, sonido y código de programa.

Los gráficos vectoriales permiten llevar a cabo animaciones de poco peso, es decir, que tardan poco tiempo en ser cargadas por el navegador, además de ser una de las más populares y completas.

Los archivos de Flash, que tienen generalmente la extensión de archivo SWF, pueden aparecer en una página web para ser vistos en un navegador o pueden ser reproducidos independientemente por un reproductor Flash. Los archivos de Flash aparecen muy a menudo como animaciones en páginas web y sitios web multimedia, y más recientemente en aplicaciones de Internet. Son también ampliamente utilizados en anuncios de la web, lo que permite aportar vistosidad a las páginas web al mismo tiempo que proporciona la posibilidad de interaccionar con el usuario. Para ello, Flash invoca un lenguaje de programación llamado Action Script, orientado a objetos, este lenguaje tiene claras influencias del JavaScript y permite, entre otras muchas cosas,



gestionar el relleno de formularios, ejecutar distintas partes de una animación en función de eventos producidos por el usuario, saltar a otras páginas, etc.

En versiones recientes, Macromedia ha ampliado Flash más allá de las animaciones simples, convirtiéndolo en una herramienta de desarrollo completa, para crear principalmente elementos multimedia e interactivos para Internet [9], [14].

3.1.4 JavaScript

Es un lenguaje interpretado orientado a las páginas web, con una sintaxis semejante a la del lenguaje Java, desarrollado por Netscape Communications [8].

Se trata de un lenguaje de tipo script compacto, basado en objetos y guiado por eventos, diseñado específicamente para el desarrollo de aplicaciones cliente-servidor dentro del ámbito de Internet.

Los programas JavaScript van incrustados en los documentos HTML, y se encargan de realizar acciones en el cliente, como pueden ser pedir datos, confirmaciones, mostrar mensajes, crear animaciones, comprobar campos, entre otras [26].

Con JavaScript se pueden crear páginas HTML dinámicas que procesen la entrada del usuario y que sean capaces de gestionar datos persistentes usando objetos especiales, archivos y bases de datos relacionales [51].



3.2 Herramienta para el Diseño de Páginas Web

3.2.1 Dreamweaver

Es la herramienta de diseño de páginas web más avanzada. Cumple el objetivo de diseñar páginas con aspecto profesional y soporta gran cantidad de tecnologías, además muy fáciles de usar:

- ◆ Hojas de estilo y capas.
- ◆ JavaScript para crear efectos e interactividades.
- ◆ Inserción de archivo multimedia, entre ellos.

Además es un programa que se puede actualizar con componentes, que fabrica tanto Macromedia como otras compañías, para realizar otras acciones más avanzadas [1], [19].

3.3 Lenguaje de Desarrollo de Ambientes Virtuales

3.3.1 VRML

VRML, acrónimo para Virtual Reality Modeling Language (Lenguaje para Modelado de Realidad Virtual), es un lenguaje para la descripción de objetos y mundos virtuales 3D, con los que el usuario puede interactuar [11].

Mediante VRML puede especificarse la geometría de los objetos de estos mundos y controlarse numerosos aspectos relativos a su visualización y comportamiento. Estos objetos pueden, al ser seleccionados, establecer hiperenlaces con otros mundos o con documentos HTML [41].



Este modo de visitar sitios en Internet es mucho más avanzado y posee grandes ventajas. Para empezar la navegación se desarrolla de una manera mucho más intuitiva, dado que la forma de actuar dentro del mundo virtual es similar a la de la vida real. Podemos movernos en todas las direcciones, no solo izquierda y derecha sino también adelante, atrás, arriba y abajo. Podemos tratar con los objetos como en la vida misma, tocarlos, arrastrarlos, etc. [12].

Entre sus principales características destaca la de ser un lenguaje estándar, y por consiguiente, universalmente utilizado en Internet como el lenguaje para simulaciones interactivas dentro de la web [11].

VRML no requiere de dispositivos externos a los equipos de cómputo domésticos para la navegación, ya que un simple ratón y el teclado son suficientes. El único requerimiento especial que precisa VRML es tener instalado en el navegador de Internet un plug-in que interprete el lenguaje. Estos plug-in son gratuitos y pueden cargarse desde las páginas de los fabricantes para visualizar los mundos en tres dimensiones [12].

3.4 Herramienta de Desarrollo de Ambientes Virtuales

3.4.1 3D Studio Max

3D Studio Max es una aplicación para la creación de gráficos y animación 3D desarrollado por Autodesk, basado en el entorno Windows (9x/NT) que permite crear tanto modelados como animaciones en tres dimensiones (3D) a partir de una serie de vistas o visores (planta y alzados). La utilización de 3D Studio Max permite al usuario la fácil visualización y representación de los modelos, así como su exportación y salvado en otros formatos distintos del que utiliza el propio programa [6], [16].



3D Studio Max es uno de los programas de animación 3D más utilizados. Dispone de una sólida capacidad de edición, una omnipresente arquitectura de plugins y una larga tradición en plataformas Microsoft Windows. 3D Studio Max es utilizado en mayor medida por los desarrolladores de videojuegos, aunque también en el desarrollo de proyectos de animación como películas o anuncios de televisión, efectos especiales y en arquitectura [2], [49].

3.5 Servidor Web

Un *servidor web* es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir hipertextos, páginas web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.

Un *servidor web* se encarga de mantenerse a la espera de peticiones HTTP llevadas a cabo por un cliente HTTP conocido como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita [13].

3.5.1 Apache

Es un servidor HTTP de código abierto para plataformas Unix (BSD¹⁶, Linux, etcétera), Windows y otras, que implementa el protocolo HTTP y la noción de sitio virtual. El servidor Apache se desarrolla dentro del proyecto HTTP Server de la Apache Software Foundation.

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido. En la

¹⁶ BSD. Berkeley Software Distribution, (Distribución de Software Berkeley).



actualidad, Apache es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios web en el mundo [13].

3.6 Sistema Gestor de Bases de Datos

3.6.1 MySQL

Es uno de los Sistemas Gestores de Bases de Datos (SGBD) más populares desarrollado bajo la filosofía de código abierto debido a que consume pocos recursos.

La desarrolla y mantiene la empresa MySQL AB, pero puede utilizarse gratuitamente y su código fuente está disponible.

Entre las ventajas que ofrece MySQL se encuentran las siguientes: mayor rendimiento, mejores utilidades de administración, integración perfecta con PHP, sin límites en los tamaños de los registros [21].

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan.

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar: amplio subconjunto del lenguaje SQL, disponibilidad en gran cantidad de plataformas y sistemas, diferentes opciones de almacenamiento según si se desea velocidad en las



operaciones o el mayor número de operaciones disponibles, transacciones y claves foráneas, conectividad segura, replicación, búsqueda e indexación de campos de texto.

Según las cifras del fabricante, existirían más de seis millones de copias de MySQL funcionando en la actualidad, lo que supera la base instalada de cualquier otra herramienta de bases de datos [10].



DESARROLLO DEL SISTEMA

4.1 Introducción

Hoy en día, con la rápida expansión de Internet y los avances en el área de las tecnologías web han aparecido un nuevo tipo de aplicaciones enfocadas a las actividades comerciales en la red.

TAV-Hidalgo es una tienda virtual de abarrotes en 3D que pone a disposición de los usuarios de Internet la posibilidad de hacer compras en línea de los productos que utiliza en la vida diaria, con la comodidad de no salir de su hogar o lugar de trabajo, sin pérdida de tiempo y con la oportunidad de emplearlo en otras actividades.

TAV-Hidalgo se compone de dos interfaces una dirigida a los usuarios navegantes (clientes) y la otra a la administración de la tienda. La segunda interfaz permite tener un control absoluto de la tienda, debido a que permite administrar a clientes, empleados, proveedores, productos e inventario.

4.2 Desarrollo de la Aplicación

TAV-Hidalgo ha sido desarrollada utilizando las siguientes herramientas tecnológicas:

- ◆ **Sistema Operativo:** *Microsoft Windows XP.*
- ◆ **Servidor Web:** *Apache 2.0.*
- ◆ **Servidor de Correo:** *ArGoSoft Mail Server.*
- ◆ **Sistema Gestor de Bases de Datos:** *MySQL 4.0.24.*



- ◆ **Lenguajes de Programación Web:** HTML, PHP, JavaScript.
- ◆ **Lenguajes de Desarrollo de Ambientes Virtuales:** VRML.
- ◆ **Herramientas de Desarrollo de Ambientes Virtuales:** 3D Studio Max 6.
- ◆ **Herramientas para el Diseño de Páginas Web:** Dreamweaver, Flash.
- ◆ **Herramientas para el Diseño de Bases de Datos:** Desing/IDEF 3.7.
- ◆ **Utilidad Web:** phpMyAdmin 2.6.1.

4.3 Requerimientos del Sistema

- ◆ Procesador compatible con Pentium II o superior.
- ◆ Sistema Operativo Windows 98, 2000, XP, Linux y Unix.
- ◆ Disco Duro con 5 GB mínimo de espacio libre.
- ◆ 128 MB de RAM como mínimo.
- ◆ Monitor VGA o superior con resolución de 800 x 600 píxeles.

4.4 Metodología de Análisis y Diseño del Sistema

TAV-Hidalgo es una aplicación web basada en la metodología de desarrollo de aplicaciones web OOWS (Object-Oriented Approach for Web Solutions Modeling), la cual se divide en dos fases:



1) Especificación del Problema que consiste en la obtención de los requisitos del sistema y se compone de cinco modelos:

- ◆ Modelo de Objetos.
- ◆ Modelo Dinámico.
- ◆ Modelo Funcional.
- ◆ Modelo de Navegación.
- ◆ Modelo de Presentación.

2) Desarrollo de la Solución donde se obtiene el prototipo software completo siguiendo la especificación realizada del sistema.

4.4.1 Especificación del Problema

Caso de Estudio: TAV-Hidalgo

Los requisitos de la aplicación TAV-Hidalgo se presentan a continuación:

La aplicación esta orientada a la venta en línea de abarrotes, organizados en una clasificación.

TAV-Hidalgo se construirá tomando como base la estructura física de diferentes tiendas como *Aurrera* (ver figura 10), *Mercado Soriana* (ver figura 11), por mencionar algunas, sin embargo TAV-Hidalgo se edificará de acuerdo a un criterio propio en cuanto a su apariencia y se tomaran ciertos aspectos en cuanto a la clasificación de los productos que utilizan las tiendas anteriormente mencionadas. Los productos deberán



responder a una acción al dar click con el ratón debido a que deben mostrar las características propias del producto además de una caja de texto en la cual se podrá introducir la cantidad es decir el número de productos que se desean comprar de dicho producto.

Existen dos tipos de usuarios de la aplicación: los *Administradores* y los *Usuarios Navegantes* (clientes). Los primeros son los encargados de la administración de los productos que se ponen en venta, así como de los usuarios, empleados, proveedores e inventario. Los segundos son los usuarios comunes (compradores) de esta aplicación.

La funcionalidad de la aplicación es:

- ◆ ***Usuarios Navegantes (clientes)***. Las compras que se realicen se deberán ir incluyendo simbólicamente en un carrito de compra; el usuario podrá consultar en cualquier momento el contenido de su carrito y realizar modificaciones sobre su contenido. Este carrito de compra se creará en el momento en el que se reciba la petición de entrada en el sistema y pertenecerá al usuario que está navegando en ese momento; todas las operaciones que el usuario realice sobre el sistema se harán de forma anónima, de modo que el usuario no deberá identificarse (registrarse) hasta que no vaya a confirmar su compra; para comprar un producto se deberá llegar a él a través de un recorrido por el mundo virtual y seleccionándolo; cuando se seleccione un producto en el mundo virtual, se mostrarán todos los datos de ese producto y se podrá comprar. Esto hará que el producto sea incluido en el carrito de compra de ese usuario y que se muestre su contenido actual; mientras se vea el contenido del carrito, se podrá cambiar la cantidad de productos que se desea adquirir de cada producto de los comprados hasta el momento o eliminar alguna de las compras del carrito; cuando se decida confirmar la compra se realizarán dos acciones: La primera consistirá en crear una nota de compra (para lo que el comprador debe haberse identificado) y la



segunda será reducir las existencias de los productos comprados; cuando se haya confirmado una compra, ya no se podrá modificar el contenido del carrito.

- ◆ **Administradores.** Gestionar y mantener la aplicación, así como el mundo virtual, los clientes, productos, empleados, proveedores, pedidos e inventario.

Croquis de la Distribución de Productos en Aurrera

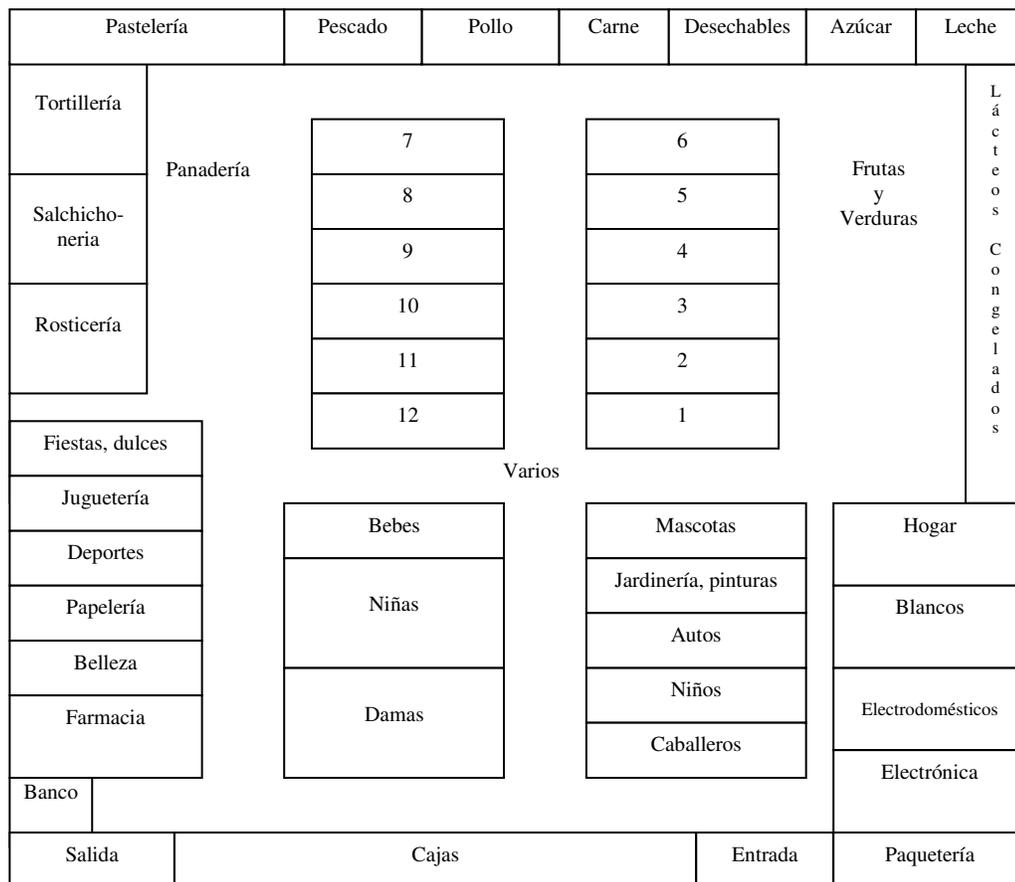


Figura 10. Croquis de Aurrera.



Pasillos

1. Atún, mayonesa, enlatados.
2. Sopas, pastas, caldo de pollo.
3. Aceite, fríjol, arroz, chiles secos.
4. Blanqueador, limpiadores, insecticidas.
5. Detergentes, suavizantes.
6. Papel higiénico, servilletas, pañuelos desechables.
7. Cerveza, vinos, licores.
8. Refrescos.
9. Galletas, néctares, jugos.
10. Avena, granola, cereales.
11. Pan de caja, gelatinas, mermeladas.
12. Chocolate, leche en polvo, café.



Croquis de la Distribución de Productos en Mercado Soriana

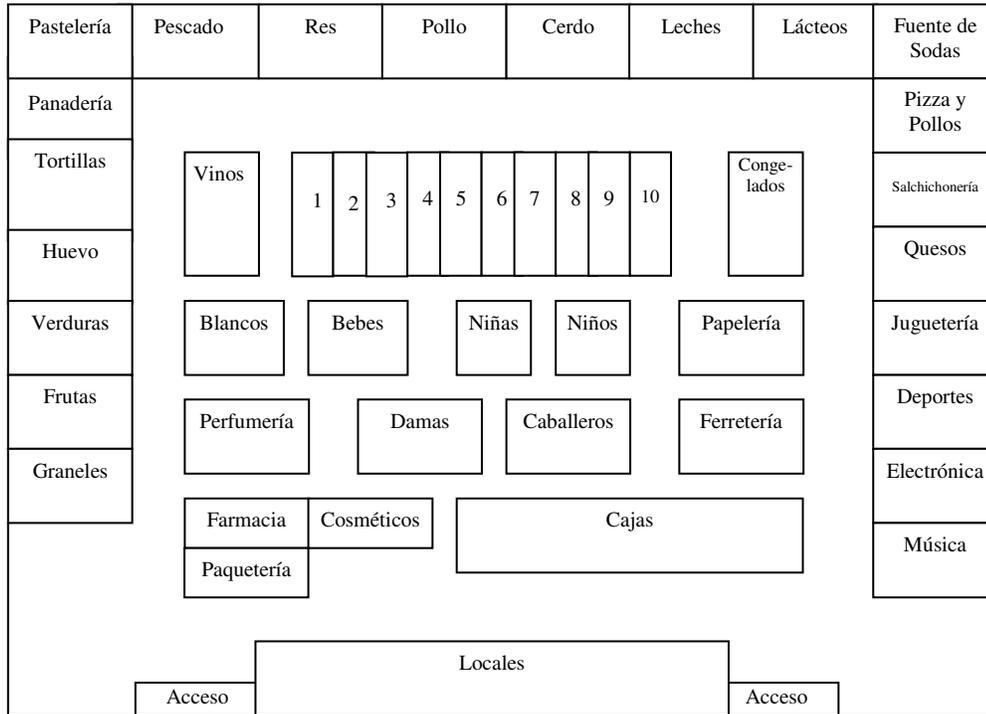


Figura 11. Croquis del Mercado Soriana.

Pasillos

1. Refrescos, agua, concentrados energéticos.
2. Tostadas, botanas, bebidas infantiles, jugos/néctares.
3. Bebidas de galón, café/té, azúcar.
4. Cereales/pan, frutas en lata, avena/postres, granola.
5. Galletas, harinas, miel, cajeta.
6. Fríjol/arroz, aceites/mantecas, moles/salsas, chiles.



7. Purés, consomés, sopas/pastas, condimentos/atún.
8. Papel higiénico, desechables, servitoallas/faciales, aluminio/carbón.
9. Detergentes, cloros/limpiadores, suavizantes, jabones.
10. Limpiadores, hogar, pinos/aromatizantes, insecticidas/velas, mascotas.
11. Cristalería, jarcería, plástico/limpieza, plástico/asador.
12. Utensilios/peltre, vajillas, loza, refractarios.
13. Licuadoras, batidoras, estufas, lavadoras.
14. Microondas, freidoras, tostadores, aspiradoras.

El sistema se divide en la funcionalidad relevante y estos requisitos son descritos mediante un diagrama de Casos de Uso. En la Figura 12 y 13 se muestran los diagramas de Casos de Uso para el agente Usuario Navegante y para el agente Administrador.

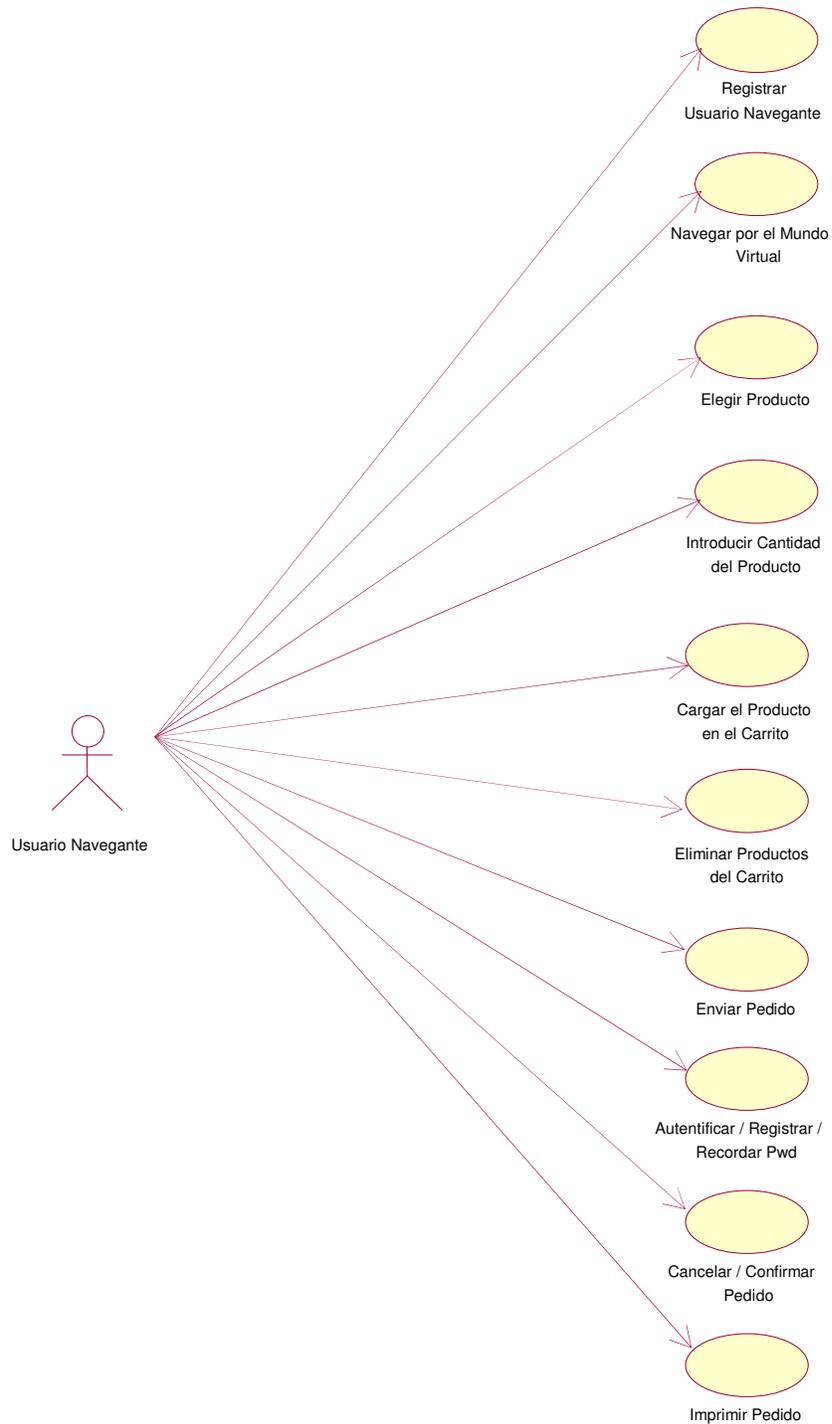


Figura 12. Caso de Uso del Agente Usuario Navegante.

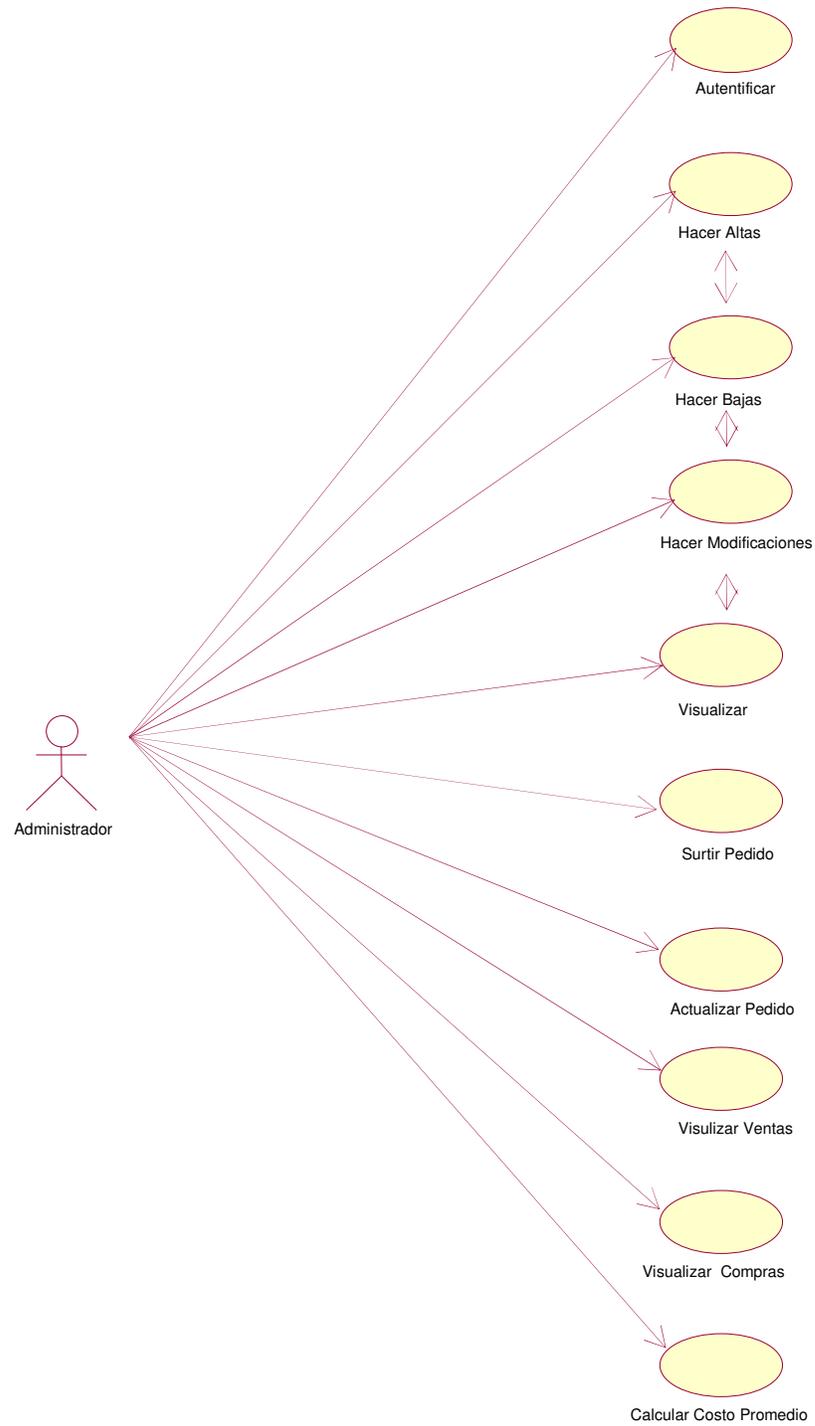


Figura 13. Caso de Uso del Agente Administrador.



Descripción de los Casos de Uso

Las tablas 2, 3, 4 y 5 representan la descripción de algunos Casos de Uso tanto del Usuario Navegante como del Administrador, cada tabla contiene el nombre del Caso de Uso, los Actores que intervienen, el Propósito del Caso de Uso, el Resumen de las actividades que se realizan, el Tipo y las Referencias Cruzadas.

Tabla 2. Registrar Usuario Navegante.

Caso de Uso:	Registrar Usuario Navegante.
Actores:	Usuario Navegante (Iniciador).
Propósito:	Dar de alta a un Usuario Navegante para su posterior ingreso al sistema.
Resumen:	<ul style="list-style-type: none">◆ El Usuario Navegante ingresa su información.◆ El sistema agrega al Usuario Navegante a la base de datos.◆ El sistema agrega el identificador y el password del Usuario Navegante a la base de datos.◆ El sistema envía los datos del identificador y el password del Usuario Navegante al correo electrónico especificado por el Usuario Navegante.
Tipo:	Primario o abstracto.
Referencias Cruzadas:	Control de acceso.

Tabla 3. Cancelar / Confirmar Pedido.

Caso de Uso:	Cancelar/Confirmar Pedido.
Actores:	Usuario Navegante (Iniciador).
Propósito:	Cancelar o Confirmar un pedido para su posterior registro en el sistema.
Resumen:	<ul style="list-style-type: none">◆ El Usuario Navegante Cancela el pedido.◆ El sistema elimina el pedido del carrito de compra.◆ El Usuario Navegante Confirma el Pedido.◆ El sistema agrega el Pedido a la base de datos.◆ El sistema genera la nota de compra de forma dinámica en formato PDF.◆ El Usuario Navegante imprime la nota de compra.
Tipo:	Primario o abstracto.
Referencias Cruzadas:	Cancelar o Confirmar Pedido.



Tabla 4. Surtir Pedido.

Caso de Uso:	Surtir Pedido.
Actores:	Administrador (Iniciador).
Propósito:	Surtir un Pedido para su posterior envío al domicilio del cliente que lo solicito.
Resumen:	<ul style="list-style-type: none">◆ El Administrador elige el Pedido que se va a surtir.◆ El Administrador modifica el Estado del Pedido y selecciona el empleado que lo va a surtir.◆ El Administrador actualiza el Pedido y lo imprime.
Tipo:	Primario o abstracto.
Referencias Cruzadas:	Surtir un Pedido.

Tabla 5. Calcular Costo Promedio.

Caso de Uso:	Calcular Costo Promedio.
Actores:	Administrador (Iniciador).
Propósito:	Calcular el Costo Promedio de un producto.
Resumen:	<ul style="list-style-type: none">◆ El Administrador elige el producto del cual se va a hacer el cálculo del Costo Promedio.◆ El sistema hace una consulta a diferentes tablas de la base de datos para extraer la información necesaria para realizar el cálculo del Costo Promedio.◆ El sistema muestra el Costo Promedio del producto.
Tipo:	Primario o abstracto.
Referencias Cruzadas:	Surtir un Pedido.

En la fase de **modelado conceptual** se construyen los siguientes modelos: *Modelo de Objetos*, *Modelo Dinámico*, *Modelo Funcional*, *Modelo Navegacional* y *Modelo de Presentación*.

La Figura 14 muestra el **Modelo de Objetos** del caso de estudio. El carrito de compra (paquete) participa directamente en el proceso de compra en línea del Usuario Navegante. Para su construcción se han estudiado los requisitos funcionales del sistema desde un punto de vista OO, y se ha asociado esta funcionalidad a cada usuario, según los Casos de Uso planteados, además de que se ha incorporado la parte administrativa del sistema.

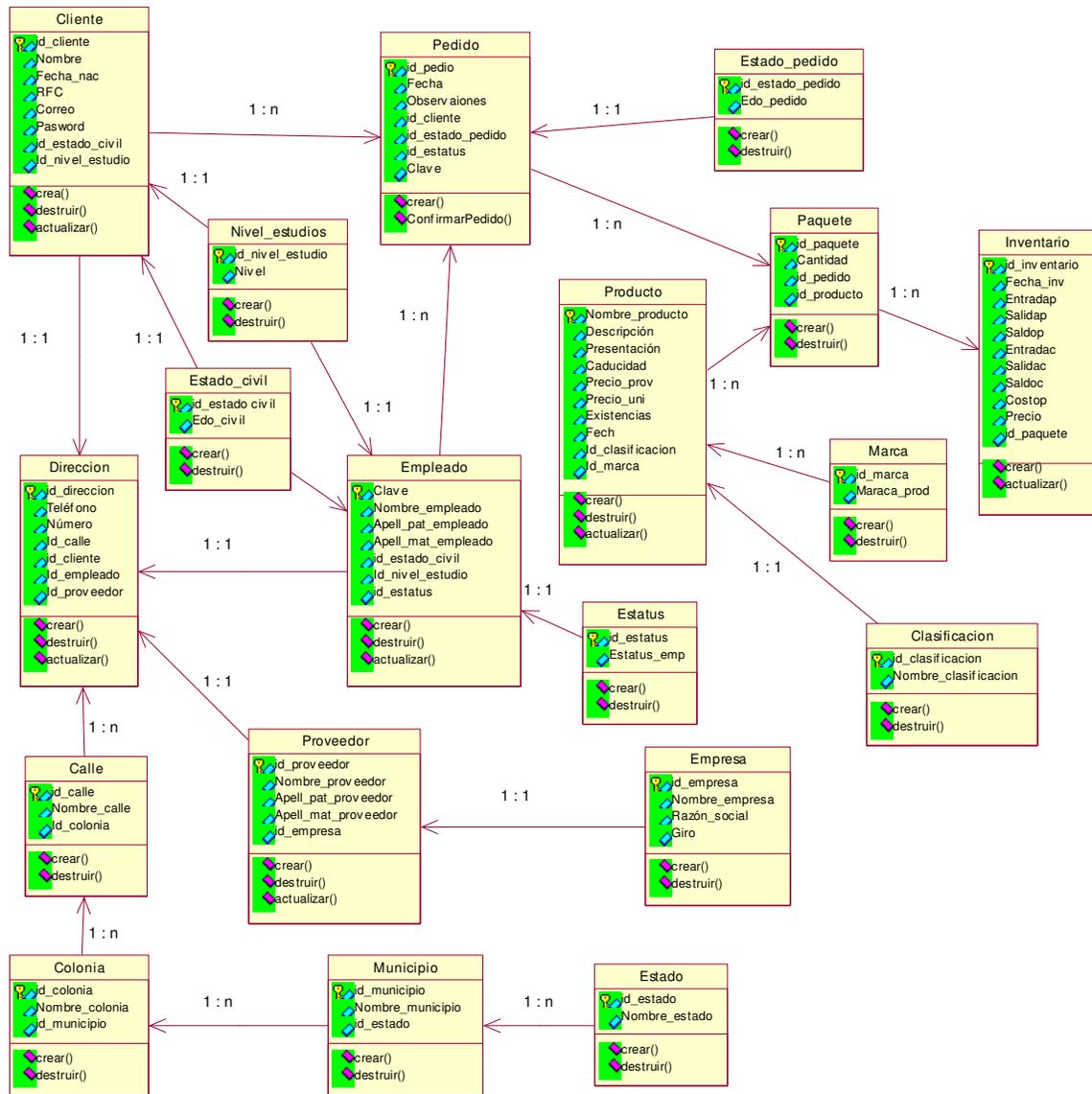


Figura 14. Modelo de Objetos.

Después del Modelo de Objetos, se construye el **Modelo Dinámico** donde se describen las vidas válidas de los objetos representando el comportamiento del sistema. La Figura 15 presenta el Modelo Dinámico de la clase Pedido.

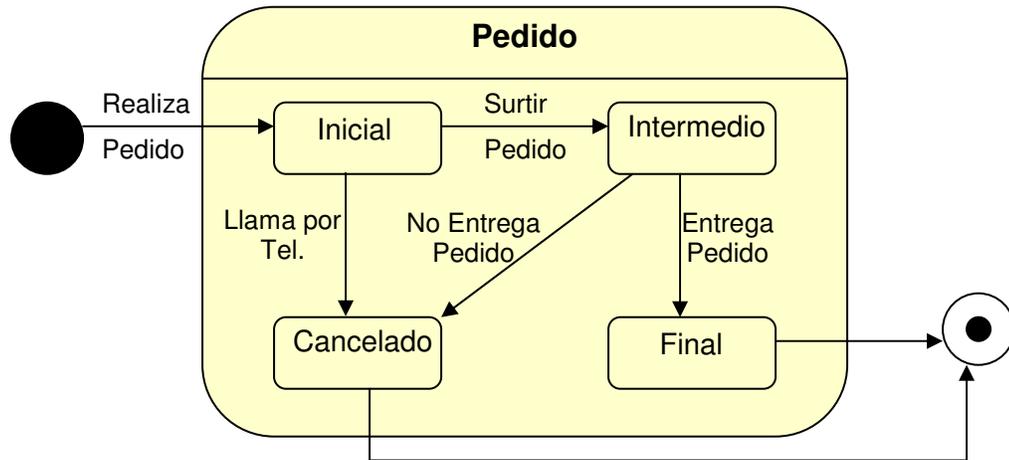


Figura 15. Modelo Dinámico del Pedido.

El **Modelo Funcional** captura la semántica asociada a los cambios de estado de los objetos. El valor de cada atributo es modificado dependiendo de la acción que activó el cambio de estado, de los argumentos de dicho evento y del estado actual del objeto.

La Figura 16 muestra un ejemplo para la clase Paquete, donde se halla definida la siguiente evaluación:

<i>Atributo:</i> Cantidad	<i>Evento:</i> ModCantidad(IngresaCantidad)
<i>Categoría:</i> De estado	<i>Condición:</i> IngresaCantidad > 0
<i>Efecto:</i> IngresaCantidad	

Figura 16. Parte del Modelo Funcional de la Clase Paquete.

A continuación se construye el **Modelo de Navegación** donde se estructura el acceso de cada usuario al sistema. La Figura 17 presenta el mapa de navegación del agente Usuario Navegante con sus contextos de navegación que han sido identificados en las primeras fases de la especificación del problema. También aparecen sobre el mapa los servicios que son ejecutados al iniciar y finalizar una sesión. Cuando el servidor web recibe una petición de un cliente, ejecuta el servicio crear del Usuario Navegante asociándole además un Carrito de Compra. Cuando el Usuario Navegante



abandona el sistema se ejecuta el servicio destruir, eliminando además, si no ha sido confirmado, su Carrito de Compra asociado.

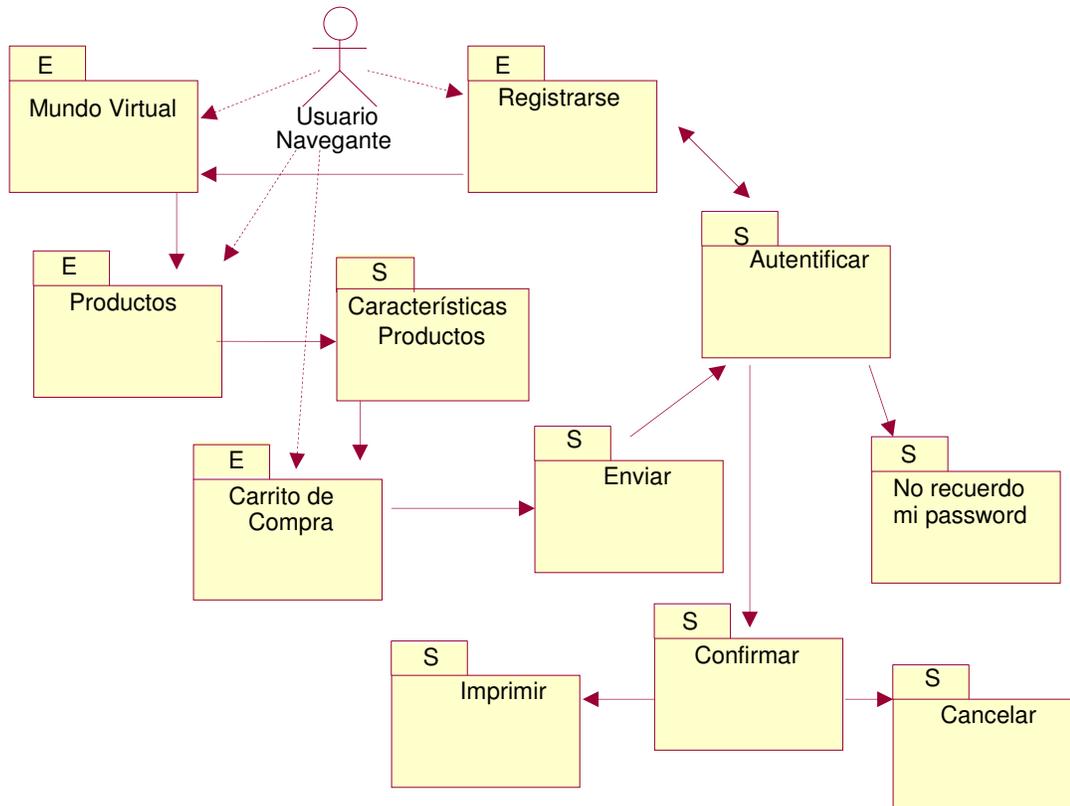


Figura 17. Mapa de Navegación del Usuario Navegante.

En este mapa de navegación se aprecia que el Usuario Navegante siempre tendrá disponibles los contextos (marcados como contextos de exploración) Mundo virtual, Productos, Carrito de Compra y Registrarse. A partir de éstos y siguiendo diferentes caminos navegacionales, podrá alcanzar los demás (Características del Producto, Nota, Registro, Autenticación, Confirmar Pedido, etc.).



La Figura 18 presenta el mapa de navegación del agente Administrador con sus contextos de navegación que han sido identificados en las primeras fases de la especificación del problema.

En este mapa de navegación se aprecia que el Administrador siempre tendrá disponibles los contextos Altas, Bajas, Modificaciones, Visualizar, Inventarios y Pedido. A partir de éstos y siguiendo diferentes caminos navegacionales, podrá alcanzar los demás (Clientes, Empleados, Productos, Proveedores, Control de inventarios, Compras, Ventas, Pendientes y Estado).

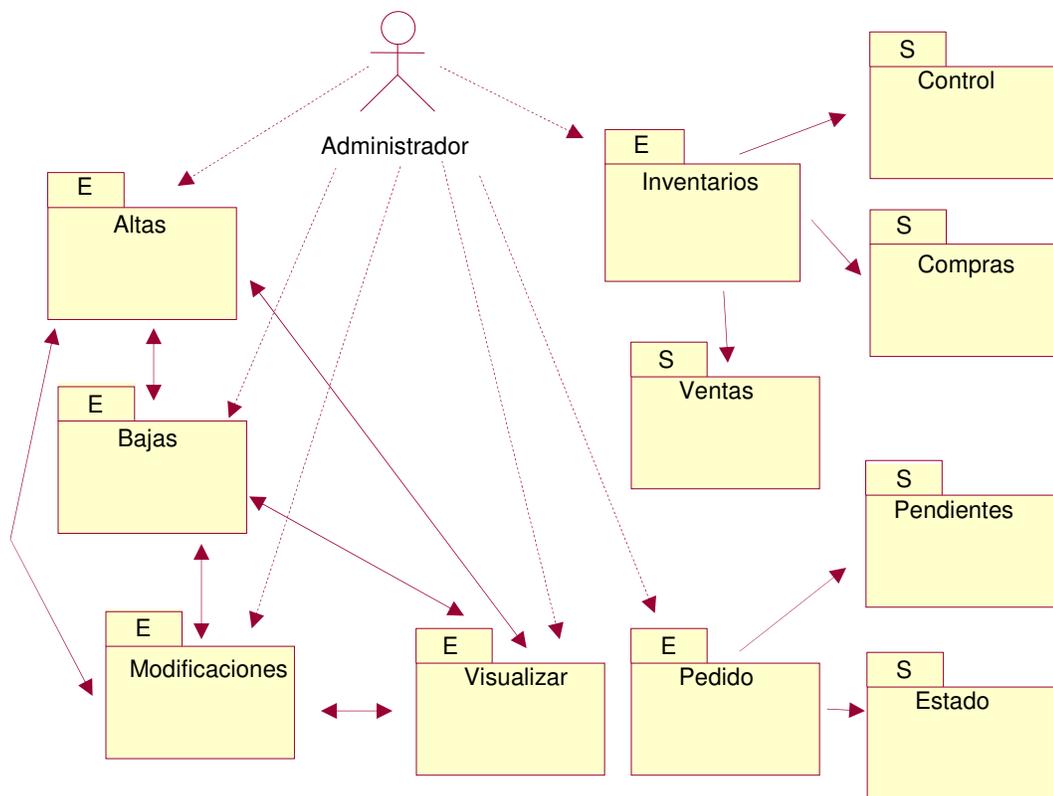


Figura 18. Mapa de Navegación del Administrador.



En la Figura 19 se describe con detalle el contexto Producto, donde se recupera la información sobre un producto seleccionándolo en el mundo virtual se puede navegar al contexto Producto, donde se proporcionará información adicional del producto y se podrá comprar.

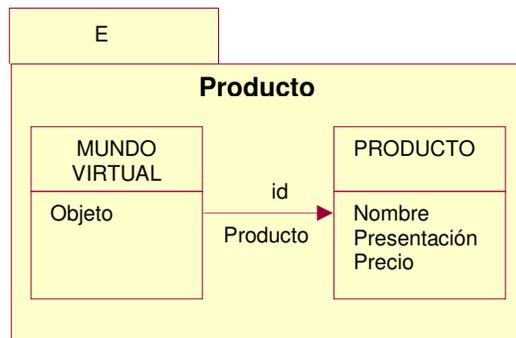


Figura 19. Contexto Productos para el Usuario Navegante.

En la Figura 20 se describe con detalle el contexto Altas, donde se proporciona la información sobre un empleado, producto, cliente o proveedor y se puede dar de alta.

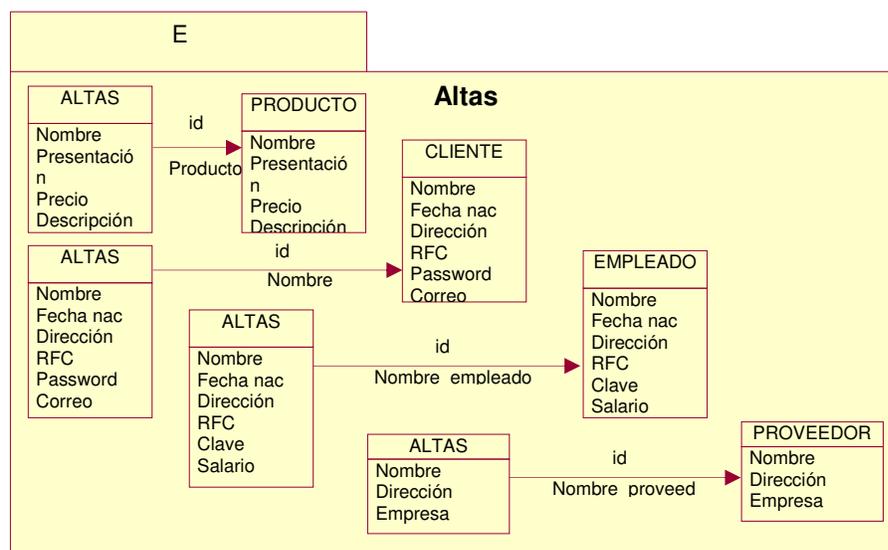


Figura 20. Contexto Altas para el Administrador.



En la Figura 21 se describe con detalle el contexto Bajas, donde se recupera la información sobre un empleado, producto, cliente o proveedor seleccionándolo en el menú se puede navegar al contexto Bajas, donde se proporcionará información adicional del empleado, producto, cliente o proveedor y se podrá dar de baja.

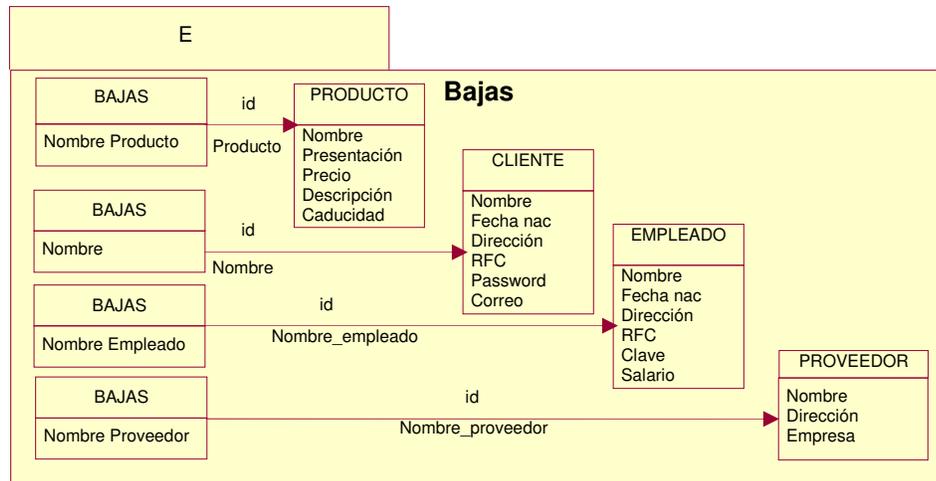


Figura 21. Contexto Bajas para el Administrador.

En la Figura 22 se describe con detalle el contexto Modificaciones, donde se recupera la información sobre un empleado, producto, cliente o proveedor seleccionándolo en el menú se puede navegar al contexto Modificaciones, donde se proporcionará información adicional del empleado, producto, cliente o proveedor y se podrá modificar la información.

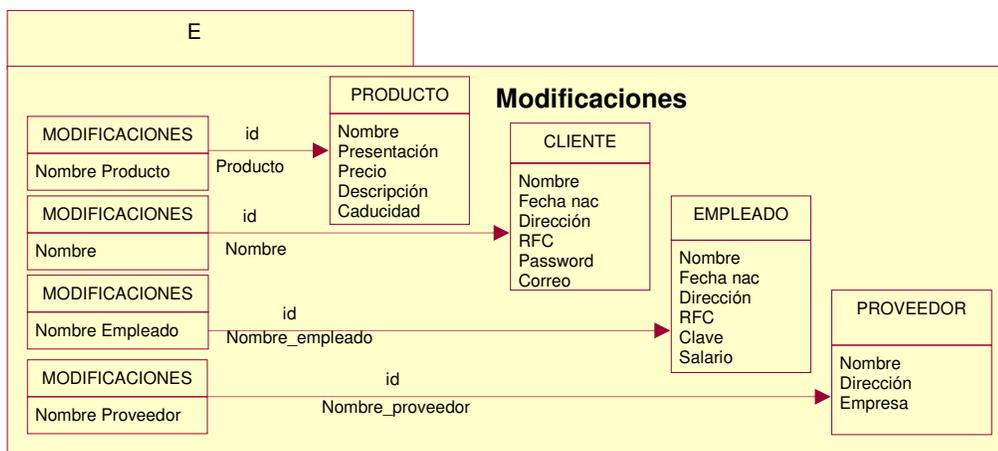


Figura 22. Contexto Modificaciones para el Administrador.



En la Figura 23 se describe con detalle el contexto Visualizar, donde se recupera la información sobre un empleado, producto, cliente o proveedor seleccionándolo en el menú se puede navegar al contexto visualizar, donde se proporcionará información adicional del empleado, producto, cliente o proveedor y se podrá visualizar la información, además de vincularse a los contextos: Altas, Bajas y Modificaciones.

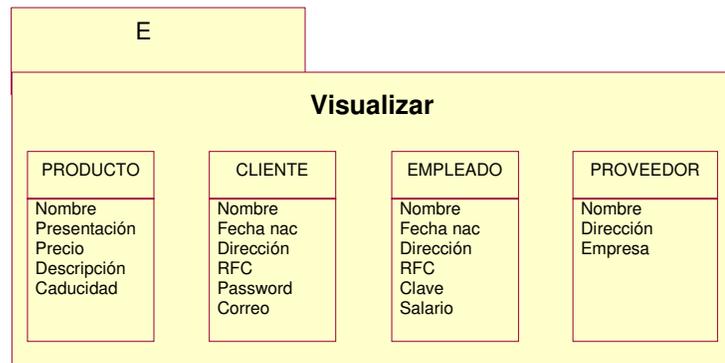


Figura 23. Contexto Visualizar para el Administrador.

En la Figura 24 se describe con detalle el contexto Inventario, donde se recupera la información sobre el Control de Inventario, Compras y Ventas seleccionándolo en el menú se puede navegar al contexto Inventario, donde se proporciona información adicional del Control de inventario, Compras y Ventas y se podrá visualizar.

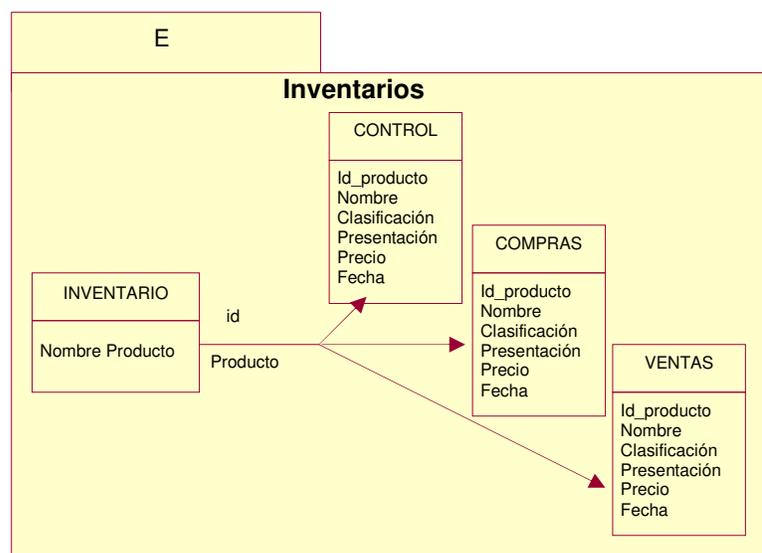


Figura 24. Contexto Inventarios para el Administrador.



En la Figura 25 se describe con detalle el contexto Pedido, donde se recupera la información sobre Pendientes y Estado, seleccionándolo en el menú se puede navegar al contexto Pedido, donde se proporciona información adicional de los pedidos pendientes y el estado de los mismos y se pueden surtir y actualizar.

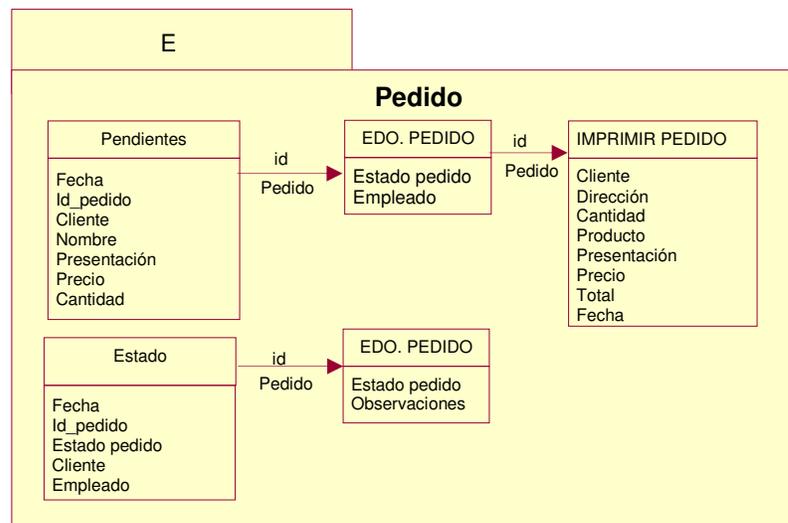


Figura 25. Contexto Visualizar para el Administrador.

Finalmente, se construye el **Modelo de Presentación** donde se captan los requisitos de presentación de información para cada contexto del mapa de navegación. En la Figura 26 se muestra la plantilla de presentación asociada al contexto *Productos* para el *Usuario Navegante*.

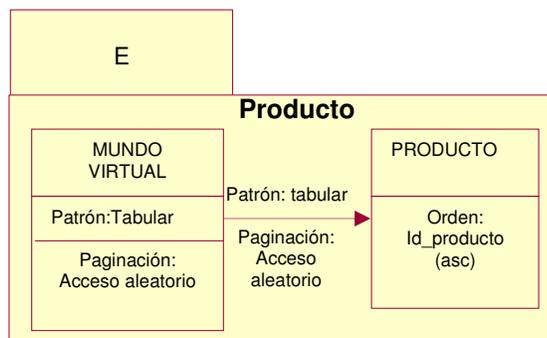


Figura 26. Modelo de Presentación del Contexto Producto para el Usuario Navegante.



En la Figura 27 se muestra la plantilla de presentación asociada al contexto *Altas* para el *Administrador*.

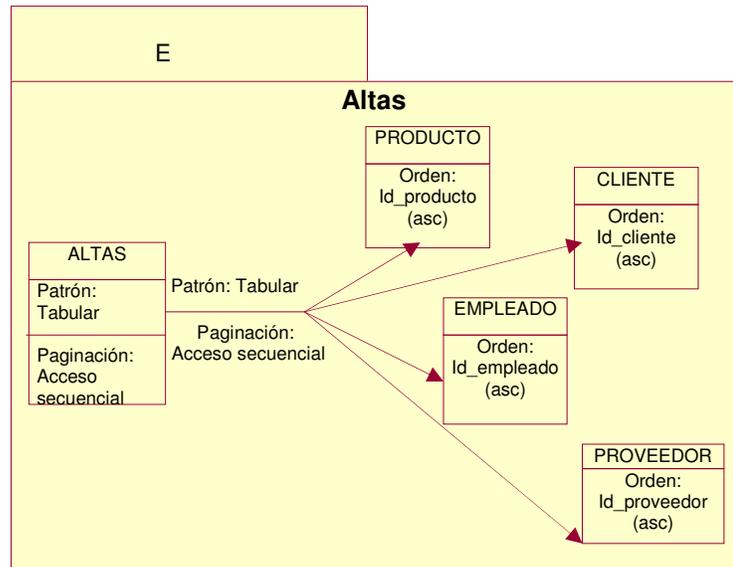


Figura 27. Modelo de Presentación del Contexto Altas para el Administrador.

En la Figura 28 se muestra la plantilla de presentación asociada al contexto *Bajas* para el *Administrador*.

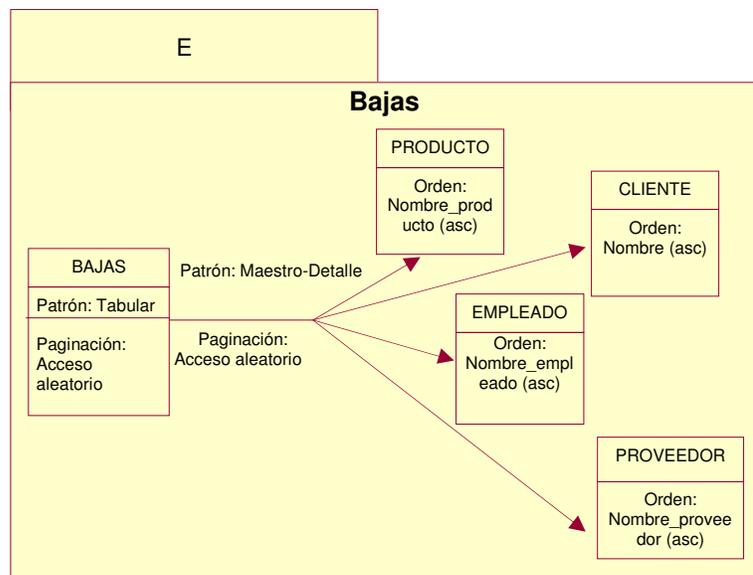


Figura 28. Modelo de Presentación del Contexto Bajas para el Administrador.



En la Figura 29 se muestra la plantilla de presentación asociada al contexto *Modificaciones para el Administrador*.

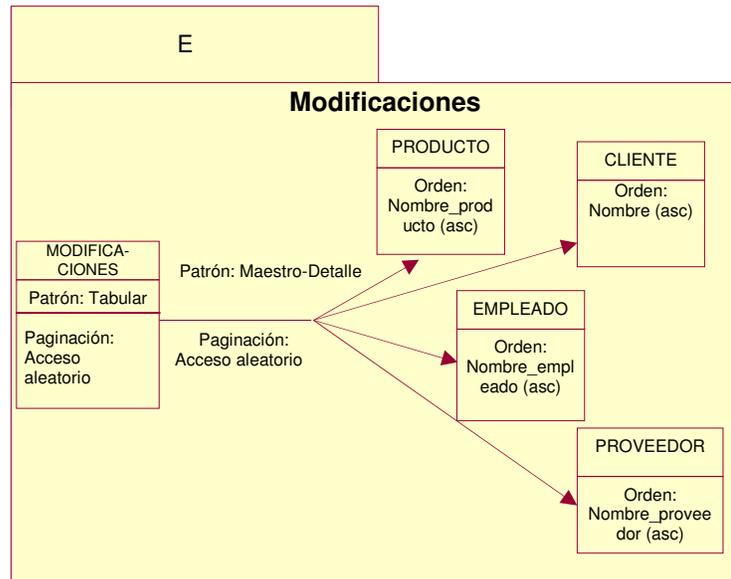


Figura 29. Modelo de Presentación del Contexto Modificaciones para el Administrador.

En la Figura 30 se muestra la plantilla de presentación asociada al contexto *Inventarios para el Administrador*.

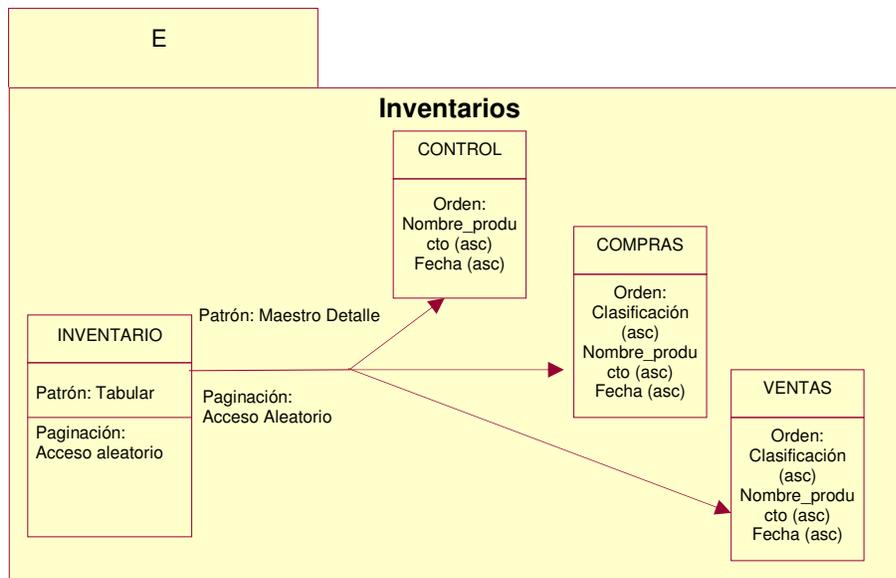


Figura 30. Modelo de Presentación del Contexto Inventarios para el Administrador.



En la Figura 31 se muestra la plantilla de presentación asociada al contexto *Pedido* para el *Administrador*.

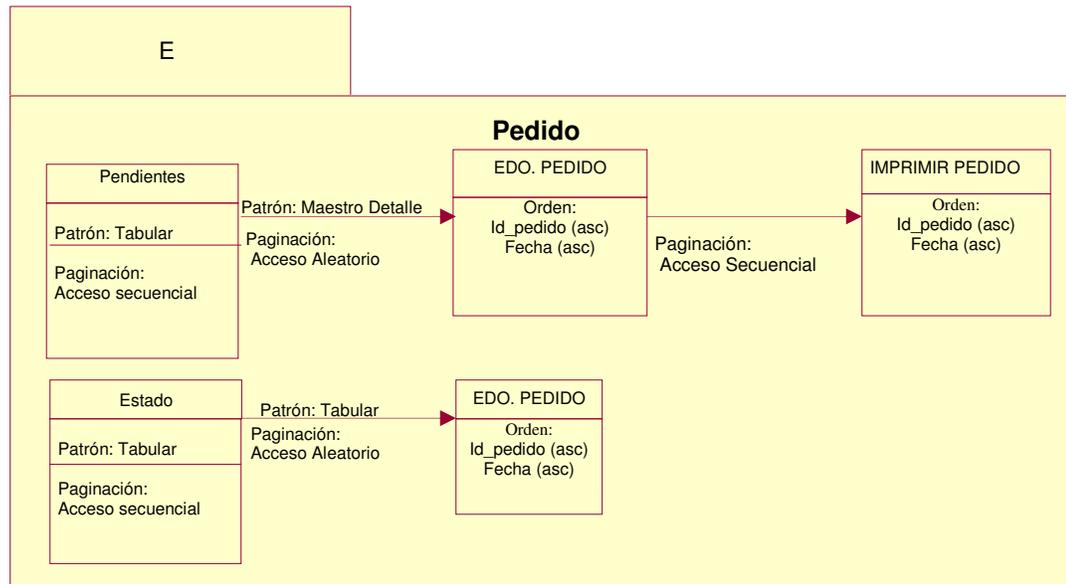


Figura 31. Modelo de Presentación del Contexto Pedido para el Administrador.

Diseño de la Base de Datos

El diseño de la base de datos se divide en: diseño conceptual, diseño lógico y diseño físico.

Diseño Conceptual: Modelo Entidad-Relación

En esta etapa se describe el contenido de la información de la base de datos.

Las tareas a realizar en el diseño conceptual son las siguientes: identificación de entidades, relaciones, atributos, dominios de los atributos, identificadores y se realizó el diagrama entidad-relación.



Diagrama Entidad-Relación

La figura 32 muestra el diagrama entidad-relación.

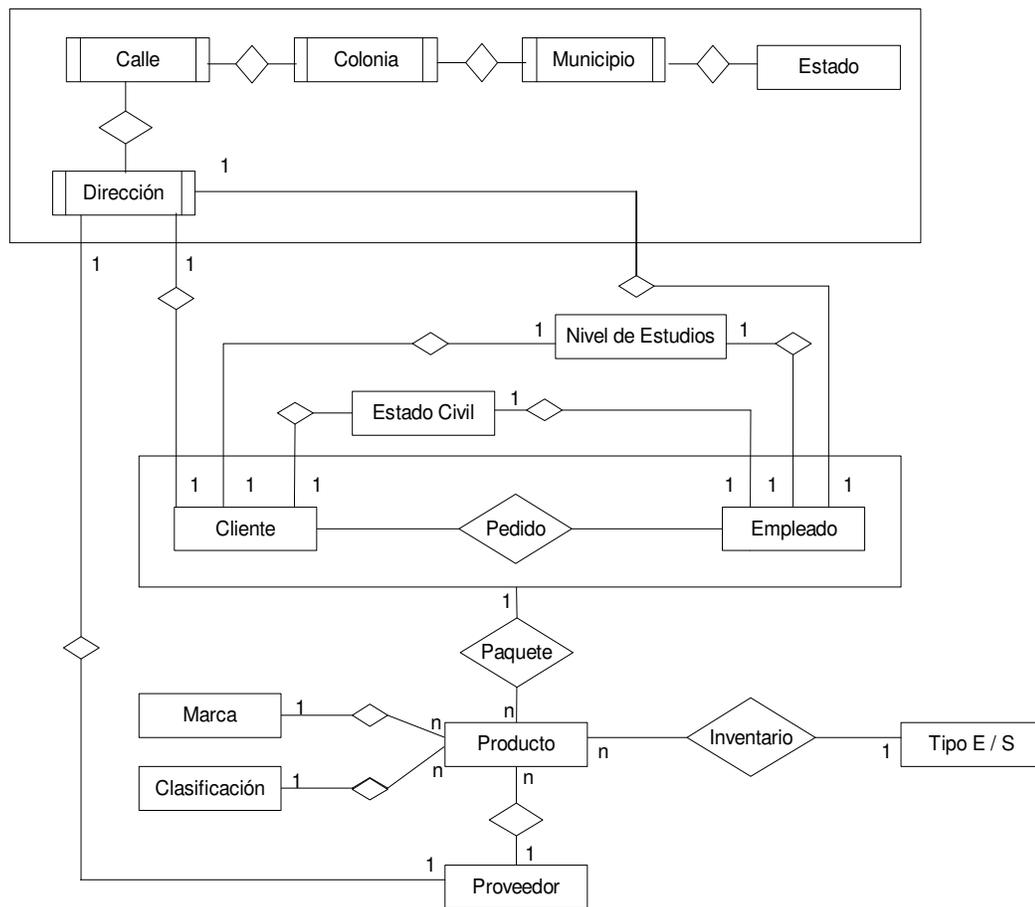


Figura 32. Diagrama Entidad-Relación.

**Diseño Físico: Diccionario de Datos**

Un esquema físico es una descripción de la implementación de una base de datos en memoria secundaria, la cual incluye las estructuras de almacenamiento y los métodos de acceso que se utilizarán para conseguir un acceso eficiente a los datos.

Tabla 6. Tabla de Calle.

calle

Campo	Tipo	Nulo	Predeterminado
id_calle	int(11)	No	
Nombre_calle	varchar(250)	Sí	NULL
id_colonia	int(11)	No	0

Tabla 7. Tabla de Clasificación.

clasificación

Campo	Tipo	Nulo	Predeterminado
id_clasificacion	int(11)	No	
Nom_clasificacion	varchar(250)	Sí	NULL

Tabla 8. Tabla de Cliente.

cliente

Campo	Tipo	Nulo	Predeterminado
id_cliente	int(11)	No	
Nombre	varchar(250)	Sí	NULL
Apell_pat	varchar(250)	Sí	NULL
Apell_mat	varchar(250)	Sí	NULL
Fecha_nac	varchar(250)	Sí	NULL
RFC	varchar(100)	Sí	NULL
Correo	varchar(150)	Sí	NULL
Password	varchar(100)	Sí	NULL
id_nivel_estudio	int(11)	No	0
id_estado_civil	int(11)	No	0

Tabla 9. Tabla de Colonia.

colonia

Campo	Tipo	Nulo	Predeterminado
id_colonia	int(11)	No	
Nombre_colonia	varchar(250)	Sí	NULL
CP	int(11)	Sí	NULL
id_municipio	int(11)	No	0



Tabla 10. Tabla de Dirección.

dirección

Campo	Tipo	Nulo	Predeterminado
id_direccion	int(11)	No	
Telefono	varchar(150)	Sí	NULL
Numero	varchar(100)	Sí	NULL
id_calle	int(11)	No	0
id_cliente	int(11)	No	0
Clave	varchar(150)	No	0
id_proveedor	int(11)	No	0

Tabla 11. Tabla del Empleado.

empleado

Campo	Tipo	Nulo	Predeterminado
Clave	varchar(150)	No	0
Nombre_empleado	varchar(250)	Sí	NULL
Apell_pat_empleado	varchar(250)	Sí	NULL
Apell_mat_empleado	varchar(250)	Sí	NULL
Fec_nac	varchar(50)	Sí	NULL
RFCE	varchar(100)	No	
Salario	Flota	Sí	NULL
id_estado_civil	int(11)	No	0
id_estatus	int(11)	No	0
id_nivel_estudio	int(11)	No	0

Tabla 12. Tabla de la Empresa.

empresa

Campo	Tipo	Nulo	Predeterminado
id_empresa	int(11)	No	
Nombre_empresa	varchar(250)	Sí	NULL
Razón_social	varchar(100)	Sí	NULL
Giro	varchar(100)	Sí	NULL

Tabla 13. Tabla del Estado.

estado

Campo	Tipo	Nulo	Predeterminado
id_estado	int(11)	No	
Nombre_estado	varchar(250)	Sí	NULL



Tabla 14. Tabla del Estado Civil.

estado_civil

Campo	Tipo	Nulo	Predeterminado
id_estado_civil	int(11)	No	
Edo_civil	varchar(150)	Sí	NULL

Tabla 15. Tabla del Estado del Pedido.

estado_pedido

Campo	Tipo	Nulo	Predeterminado
id_estado_pedido	int(11)	No	
Edo_pedido	varchar(150)	Sí	NULL

Tabla 16. Tabla del Estatus.

estatus

Campo	Tipo	Nulo	Predeterminado
id_estatus	int(11)	No	
Estatus_emp	varchar(150)	Sí	NULL

Tabla 17. Tabla del Inventario.

inventario

Campo	Tipo	Nulo	Predeterminado
id_inventario	int(11)	No	
Fecha_inv	varchar(100)	Sí	NULL
Entradap	int(11)	Sí	NULL
Salidap	int(11)	Sí	NULL
Saldop	int(11)	Sí	NULL
Entradac	Flota	Sí	NULL
Salidac	Flota	Sí	NULL
Saldoc	Flota	Sí	NULL
Costop	Flota	Sí	NULL
Precio	Flota	Sí	NULL
id_producto	int(11)	No	0
id_pedido	int(11)	No	0

Tabla 18. Tabla de la Marca.

marca

Campo	Tipo	Nulo	Predeterminado
id_marca	int(11)	No	
Marca_prod	varchar(200)	Sí	NULL



Tabla 19. Tabla del Municipio.

municipio

Campo	Tipo	Nulo	Predeterminado
id_municipio	int(11)	No	
Nombre_municipio	Varchar(250)	Sí	NULL
id_estado	int(11)	No	0

Tabla 20. Tabla del Nivel de Estudios.

nivel_estudios

Campo	Tipo	Nulo	Predeterminado
id_nivel_estudio	int(11)	No	
Nivel	varchar(100)	Sí	NULL

Tabla 21. Tabla del Paquete.

paquete

Campo	Tipo	Nulo	Predeterminado
Cantidad	int(11)	Sí	NULL
id_producto	int(11)	No	0
id_paquete	int(11)	No	
id_pedido	int(11)	No	0
id_cliente	int(11)	No	0

Tabla 22. Tabla del Pedido.

pedido

Campo	Tipo	Nulo	Predeterminado
id_pedido	int(11)	No	
Fecha	varchar(50)	Sí	NULL
Observaciones	varchar(250)	No	
Clave	varchar(100)	No	0
id_cliente	int(11)	No	0
id_estado_pedido	int(11)	No	0

Tabla 23. Tabla del Producto.

producto

Campo	Tipo	Nulo	Predeterminado
id_producto	int(11)	No	
Nombre_producto	varchar(250)	Sí	NULL
Descripcion	varchar(250)	Sí	NULL
Presentacion	varchar(100)	Sí	NULL
Caducidad	varchar(50)	Sí	NULL



Campo	Tipo	Nulo	Predeterminado
Precio_prov	Flota	Sí	NULL
Precio_uni	Flota	Sí	NULL
Existencias	int(11)	Sí	NULL
Fech	varchar(50)	No	
id_marca	int(11)	No	0
id_clasificacion	int(11)	No	0
id_proveedor	int(11)	No	0

Tabla 24. Tabla del Proveedor.

proveedor

Campo	Tipo	Nulo	Predeterminado
id_proveedor	int(11)	No	
Nombre_proveedor	varchar(250)	Sí	NULL
Apell_pat_proveedor	varchar(250)	Sí	NULL
Apell_mat_proveedor	varchar(250)	Sí	NULL
id_empresa	int(11)	No	0

Diseño del Mundo Virtual

Para construir TAV-Hidalgo es necesario diseñar un plano de la tienda de abarrotes virtual, debido a que permite conocer las dimensiones de la construcción a modelar, determinar fácilmente las distancias entre los elementos y ubicarlos en una escena (ver figura 34 y 35).

En el diseño también se determinan los tipos de objetos que contendrá la tienda, en este caso son estáticos y dinámicos, debido a que tendrá figuras geométricas que dan forma a la tienda de abarrotes virtual que se va a modelar, además de un evento a través de los objetos por medio de una ruta, los cuales simulan los productos de la tienda, cada uno de ellos con diferentes formas, tamaños, colores y texturas.

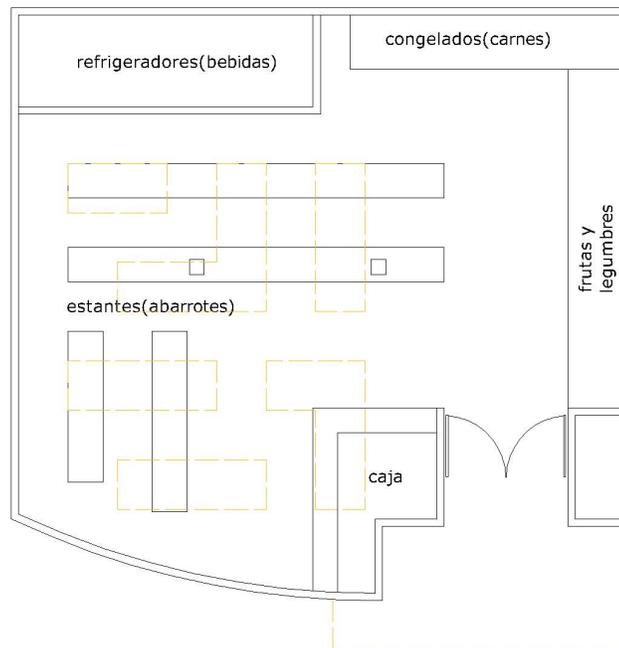


Figura 34. Plano de Planta de TAV-Hidalgo.

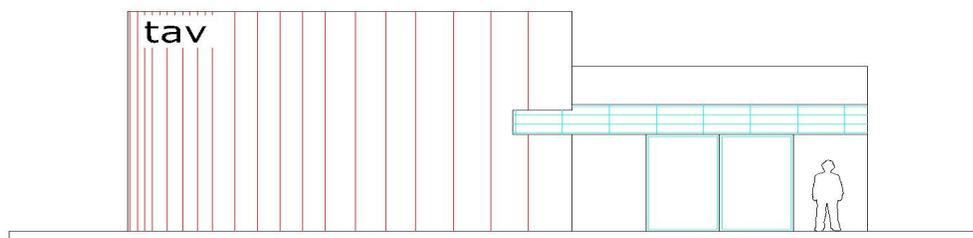


Figura 35. Fachada de TAV-Hidalgo.



4.4.2 Desarrollo de la Solución

Una vez instalados y configurados *Apache*, *PHP*, *MySQL*, *PhpMyAdmin*, 3D Studio Max, Cortona y ArGoSoft Mail Server, para tener mayor información de como instalar y configurar las tecnologías antes mencionadas consultar los siguientes manuales: [Apache + PHP + MySQL + PhpMyAdmin como módulo de Apache \[25\]](#) y [Servidor de Correo \[35\]](#).

En esta fase se realiza el desarrollo e implementación del mundo virtual, la base de datos, el sitio web, para finalmente hacer la integración total del sistema.

Mundo Virtual

El mundo virtual que simula la tienda física y los productos de TAV-Hidalgo se desarrolló con la ayuda de 3D Studio Max. En la figura 36 se pueden observar diferentes perspectivas de la tienda.

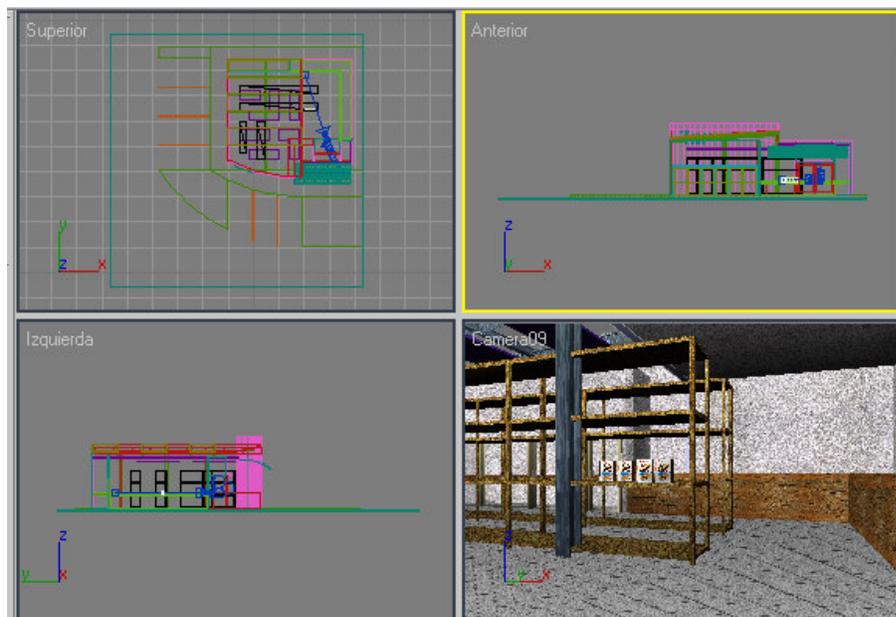


Figura 36. Perspectivas de TAV-Hidalgo.



Una vez que se termino el mundo virtual se exportó a formato VRML (ver figura 37), para lo que es necesario en el *Exportador VRML* especificar el prefijo que se usará en las rutas de las texturas (ver figura 38), con la finalidad de que la tienda se pueda ver desde el explorador a través de un plug in de realidad virtual.

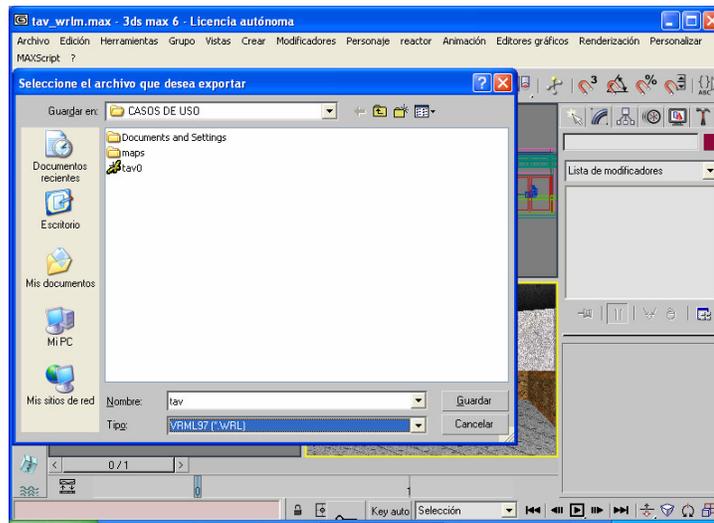


Figura 37. Pantalla para exportar el Mundo Virtual a formato VRML.

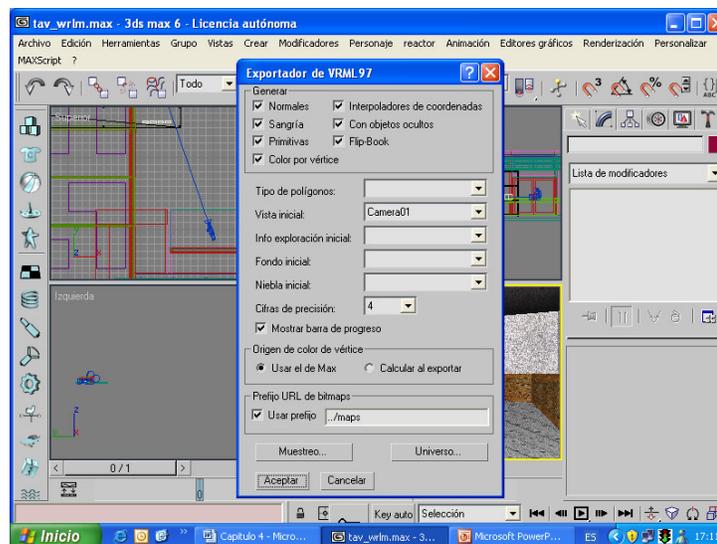


Figura 38. Pantalla del Exportador de VRML.



Para poder hacer que un objeto del mundo virtual reaccionara a un evento del ratón, se utilizó una función de VRML que lleva por nombre *Anchor*, esta función lo que hace es convertir un determinado objeto del mundo virtual en un objeto activo para que al dar click en ese objeto se vincule hacia una página PHP (ver figura 39), lo que permite que se pueda pasar un parámetro que en este caso es el *id del producto* que permite que se identifique y se pueda realizar una consulta a la base de datos, esto se logra por medio de ciertas instrucciones SQL, PHP y HTML (ver figura 40), esta función se implemento a cada uno de los objetos que simulan los productos en el mundo virtual.

```
Anchor {
  url
  ["http://localhost/Pruebas1/index.php?variable=10"]
  parameter ["target=mainFrame"]
  children [
    DEF cocacola Transform {
      translation 0 0 0
      children [
        DEF cocacola-TIMER TimeSensor { loop TRUE
        cycleInterval 1 },
        Shape {
          appearance Appearance {
            material Material {
              diffuseColor 0.8039 0.7255 0.1961
              ambientIntensity 0.1795
              specularColor 0.045 0.045 0.045
              shininess 0.2875
              transparency 0 }
            texture ImageTexture { url
            "../maps/BASSWD.JPG" }
          }
        }
      ]
    }
  ]
}
```

Figura 39. Función Anchor en VRML.



```
<?
mysql_connect("localhost","isamiri","mica1982");
//Ejecutamos la sentencia SQL
$result=mysql_db_query("fin","select * from producto Where
id_producto='$variable'");
echo"<FORM
METHOD='POST'ACTION='validar.php?id_producto=".$variable."&".
$Nombre.'">";
?>
<CENTER>
<table border=1 cellpadding="3">
  <td bgcolor=#dcd8fc><b>Nombre producto</b></td>
  <td bgcolor=#dcd8fc><b>Presentación</b></td>
  <td bgcolor=#dcd8fc ><b>Precio</b></td>
  <td bgcolor=#dcd8fc><b>Cantidad</b></td>
</table>
<?
while ($row=mysql_fetch_array($result))
{
  echo '<tr><td
bgcolor=#f9cafb>'.$row["Nombre_producto"].'</td>';
  echo '<td bgcolor=#f9cafb>'.$row["Presentacion"].'</td>';
  echo '<td bgcolor=#f9cafb>'.$row["Precio_uni"].'</td>';
  $Nombre = $row["Nombre_producto"];
  $Presentacion = $row["Presentacion"];
  $Precio_uni = $row["Precio_uni"];
}
mysql_free_result($result)
?>
```

Figura 40. Código SQL, PHP y HTML para Obtener y Visualizar los Datos del Producto.

Base de Datos

Una vez terminado el diseño de la base de datos se procedió a hacer el modelo de la misma en Desing/IDEF (ver figura 41), posteriormente éste se exportó a código SQL (ver figuras 42 y 43) y finalmente se implementó en PhpMyAdmin. Lo primero que se hace es crear una nueva base de datos que en éste caso se denomina *fin* (ver figura 44), una vez que se crea la base de datos se pega el código SQL que se generó



anteriormente (ver figura 45), si no existen errores en el código se crea la base de datos para así poder trabajar con ella (ver figura 46).

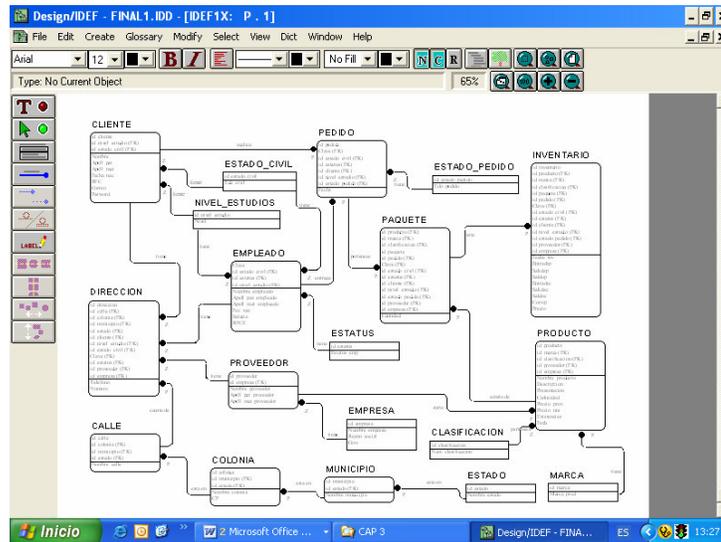


Figura 41. Modelo de la Base de Datos.

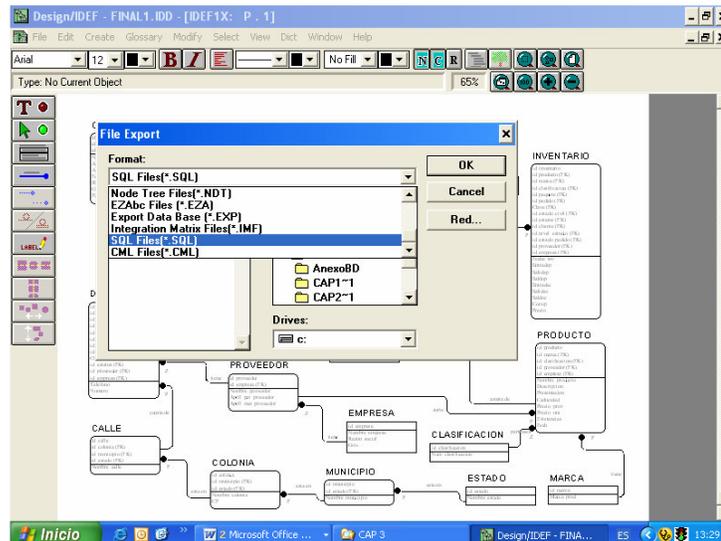


Figura 42. Pantalla para Exportar la Base de Datos a Código SQL.



```
CREATE TABLE CLIENTE
(
    id_cliente    int NOT NULL,
    Nombre        char,
    Apell_pat     char,
    Apell_mat     char,
    Fecha_nac    char,
    RFC           char,
    Correo        char,
    Password     char,
    id_nivel_estudio int NOT NULL,
    id_estado_civil int NOT NULL
);

CREATE UNIQUE INDEX IXCLIENTE ON
CLIENTE
(
    id_cliente    ASC,
    id_nivel_estudio ASC,
    id_estado_civil ASC
);
```

Figura 43. Parte del Código SQL Generado por Desing/IDEF.

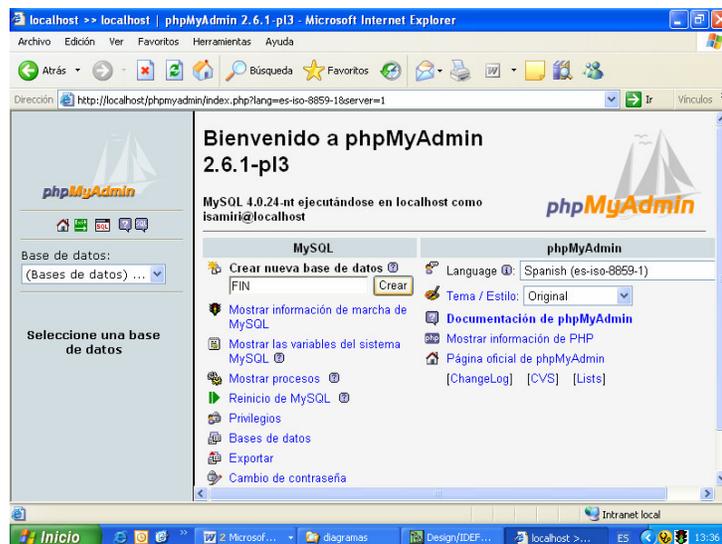


Figura 44. Pantalla para Crear una Base de Datos en phpMyAdmin

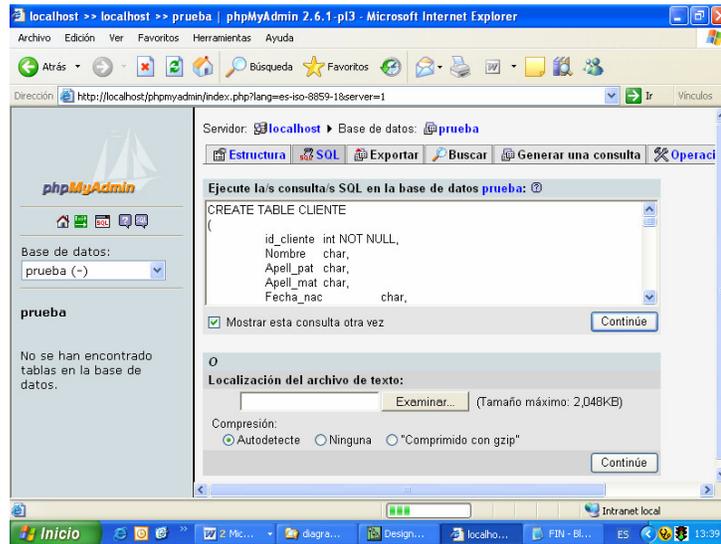


Figura 45. Pantalla para Generar la Base de Datos en phpMyAdmin.

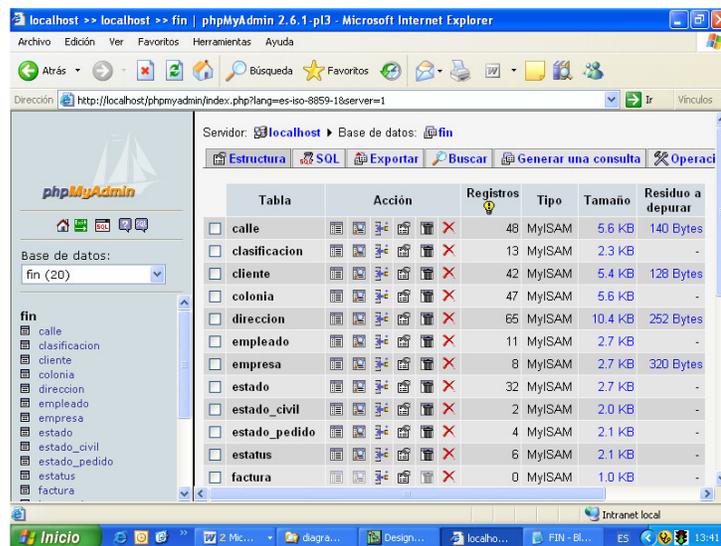


Figura 46. Pantalla que Visualiza la Base de Datos en phpMyAdmin.

Así mismo se programó el código PHP y HTML necesario para generar el carrito de la compra, la nota del pedido, que es generada de forma dinámica en formato PDF mediante una clase escrita en PHP que se llama **FPDF** y el registro del cliente de la interfaz principal de usuario, además de las altas, bajas, modificaciones, inventario y



pedido de la interfaz principal del administrador, este código fue generado en un editor de texto (ver Anexo 1).

FPDF

Es una clase escrita en PHP que permite generar documentos PDF directamente desde PHP, es decir, sin usar la biblioteca PDFlib. La ventaja es que, mientras PDFlib es de pago para usos comerciales, la F de FPDF significa *Free* (gratis y libre), se puede usar para cualquier propósito y modificarla para satisfacer las necesidades.

FPDF tiene otras ventajas: *funciones de alto nivel*. Esta es una lista de sus principales características:

- ◆ Elección de la unidad de medida, formato de página y márgenes.
- ◆ Gestión de cabeceras y pies de página.
- ◆ Salto de página automático.
- ◆ Salto de línea y justificación del texto automáticos.
- ◆ Admisión de imágenes (JPEG y PNG).
- ◆ Colores.
- ◆ Enlaces.
- ◆ Admisión de fuentes TrueType, Type1 y codificación.
- ◆ Compresión de página.



FPDF no necesita de ninguna extensión para PHP (excepto la biblioteca zlib si se va a activar la opción de compresión) y funciona con PHP4 y PHP5.

Por último se realizó la integración de todo el sitio web por medio de Dreamweaver y para poder visualizarlo, en el Explorador de Windows se ejecuta <http://localhost/Final/uno.htm> (ver figura 47).



Figura 47. Interfaz Principal de TAV-Hidalgo.

Como se puede apreciar en la figura 47, esta implementación recoge las características navegacionales y de presentación de información que se habían especificado en la definición del problema.

En la figura 48 se muestra una vista de la tienda de abarrotes virtual (TAV-Hidalgo) en la cual se pueden visualizar los productos que están en venta.



Figura 48. Vista de la Tienda TAV-Hidalgo.

En la figura 49 se muestra la interfaz administrativa de la tienda de abarrotes virtual (TAV-Hidalgo).

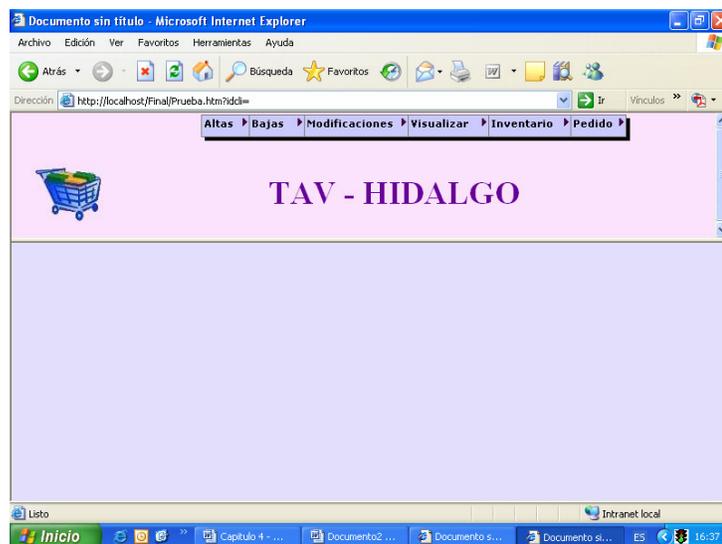


Figura 49. Interfaz Administrativa de TAV-Hidalgo.



4.5 Funcionamiento de TAV-Hidalgo

TAV-Hidalgo es un aplicación web instalada en un servidor web, para que los usuarios puedan interactuar con ella es necesario que se conecten a la página del sitio. En la figura 50 se muestra la forma en que interactúan el mundo virtual, el servidor y la base de datos, para la interfaz del Usuario Navegante.

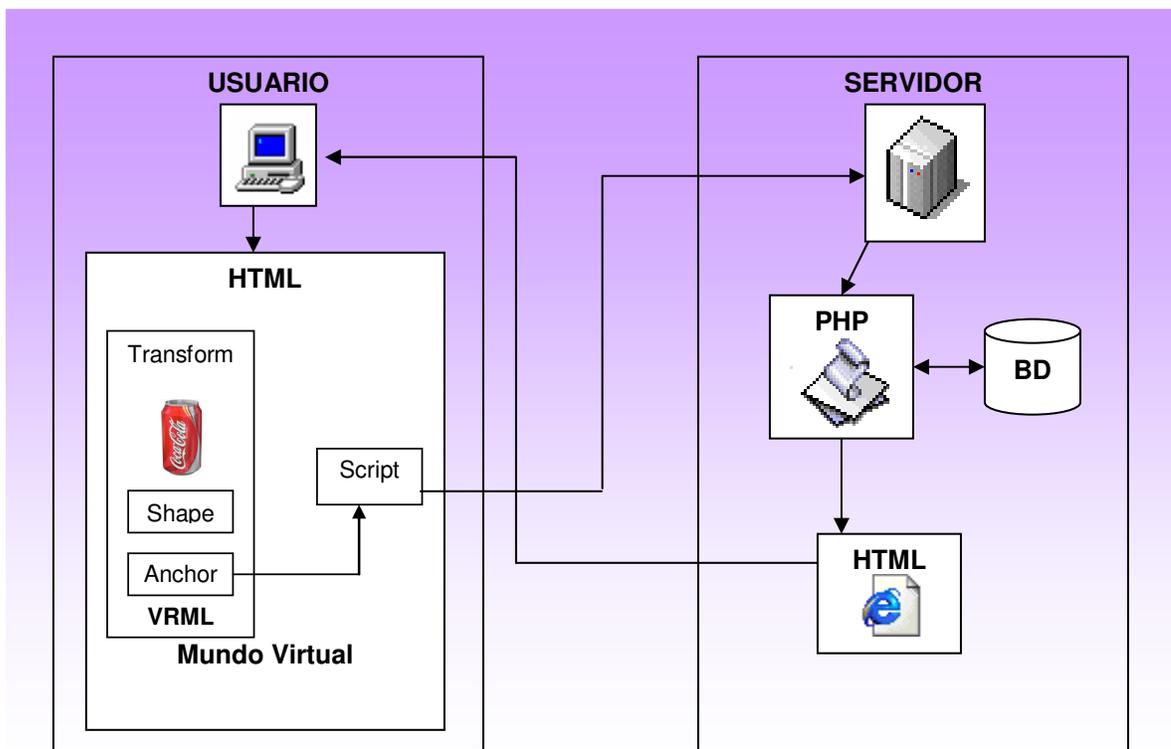


Figura 50. Funcionamiento de TAV-Hidalgo para el Usuario Navegante.

En la figura 51 se muestra la forma en que interactúan el servidor y la base de datos, para la interfaz del Administrador.

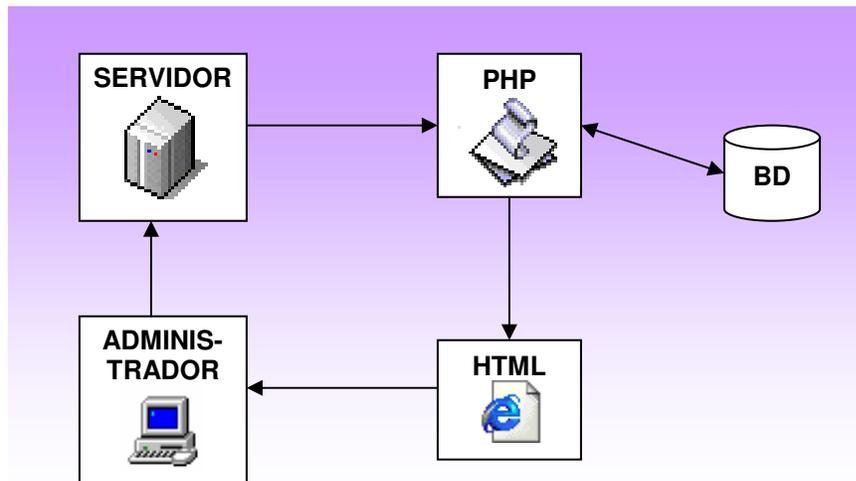


Figura 51. Funcionamiento de TAV-Hidalgo para el Administrador.

Este trabajo ha participado en algunos eventos del 1er Congreso Estatal Universitario en Tecnologías de Información y Comunicaciones TIC'S, organizado por la Universidad Autónoma del Estado de Hidalgo, la Dirección General de Servicios Académicos, el Instituto de Ciencias Básicas e Ingeniería y el Centro de Cómputo Académico, tanto en la ponencia "Tienda Virtual de Abarrotes-Hidalgo TAV-Hidalgo"; así como en la Exposición Tecnológica (ver Anexo 2).

Del mismo modo participó en el Seminario de Tesis que organiza el Centro de Investigación en Tecnologías de Información y Sistemas de la Universidad Autónoma del Estado de Hidalgo.



CONCLUSIONES Y TRABAJOS FUTUROS

5.1 Conclusiones

Debido al florecimiento del comercio electrónico, gracias a la expansión del Internet y los beneficios que esto genera como el ahorro económico que representa el no tener un local físico, gran número de empleados, entre otros; se pensó en la realización de una tienda de abarrotes virtual que introdujera algo atractivo y novedoso, en este caso un mundo virtual que le ofrece a los usuarios la oportunidad de hacer un recorrido a través de él e interactuar con los objetos que en él se encuentran.

La presente tesis, presentó el desarrollo de una tienda virtual en 3D utilizando la metodología OOWS para la construcción de la aplicación web. El trabajo realizado ha consistido en el diseño e implementación de la una Base de Datos, del Mundo Virtual, así como del Sitio Web.

A lo largo del desarrollo de este sistema se resolvieron diferentes contratiempos como fue el lograr la comunicación entre el mundo virtual implementado con VRML con la tecnología PHP, así como de PHP con la base de datos implementada en MySQL para poder mostrar en la página web la información correcta de cada producto seleccionado por el usuario.

Otro inconveniente con el que nos encontramos a lo largo del desarrollo del presente trabajo es la necesidad de generar la Nota de Compra para el Cliente, en un formato que no permitiera su modificación, el formato que nos proporcionó esta particularidad es el *PDF*, para poder generar un documento en formato PDF mediante PHP, tuvimos la necesidad de utilizar una clase escrita en PHP llamada *FPDF*, la cual permite generar un documento en formato *PDF*.



Del mismo modo se logró implementar un sistema de control de inventarios el cual permite tener un registro de los movimientos que se hacen con cada uno de los productos que se ofertan en TAV-Hidalgo (compra, venta, etc.), así como realizar el cálculo del costo promedio de cada uno de los productos.

También se logró que a través de la interfaz administrativa se pudiera tener un control completo sobre los pedidos en lo que respecta a estado y el empleado que lo surte.

5.2 Trabajos Futuros

Los trabajos futuros que se tienen contemplados para que TAV-Hidalgo tenga un óptimo funcionamiento y se encuentre a la vanguardia en lo que se refiere a comercio electrónico son los siguientes:

- ◆ Implementar el pago en línea.
- ◆ Incorporar un agente inteligente en el Mundo Virtual que simule al cliente.
- ◆ Realizar un sistema de facturación.
- ◆ Implementar un método de encriptación para mejorar la seguridad.
- ◆ Ampliar la gama de productos que se ofrecen con artículos electrodomésticos, blancos, ropa, artículos de jardinería, etc. Para que pase a ser de una tienda de abarrotes a un centro comercial como los que en la actualidad existen.
- ◆ Ampliar la información de los productos que se muestra al cliente con los datos nutrimentales.



- ◆ Incorporar un carrito en el mundo virtual en el que se puedan ir depositando los productos elegidos por el cliente.
- ◆ Implementar un mecanismo que elimine los productos que se venden del mundo virtual.
- ◆ Hacer que los informes del inventario puedan ser descargados en una hoja de cálculo.
- ◆ Enviar emails a los clientes sobre las ofertas especiales.
- ◆ Dar asistencia sobre el proceso de compra por medio de un vendedor por voz o video.



GLOSARIO

A

Avatar, representación gráfica (mediante un dibujo o fotografía) de una persona para su identificación.

B

Banner, imagen, gráfico o texto utilizado con fines publicitarios la cual generalmente presenta un pequeño tamaño, aparece en una página web y habitualmente la enlaza con el sitio web del anunciante.

Base de Datos, es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso..

BSD *Berkeley Software Distribution* (**Distribución de Software Berkeley**), se utiliza para identificar un sistema operativo derivado del sistema Unix nacido a partir de las aportaciones realizadas a ese sistema por la Universidad de California en Berkeley.

C

Carrito de Compra, área de un sitio Web de comercio electrónico donde el usuario va colocando los objetos o servicios a medida que los va comprando, de la misma manera que haría en un supermercado. Al final el usuario decide cuáles de ellos comprar.

Certificado Digital, acreditación emitida por una entidad o un particular debidamente autorizado garantizando que un determinado dato (una firma

electrónica o una clave pública) pertenece realmente a quien se supone.

Ciberespacio, término concebido por el escritor William Gibson en su novela de ciencia ficción "Neuromancer" (1984) con el propósito de describir un mundo de redes de información. Actualmente es utilizado para referirse al conjunto de información digital y a la comunicación que se realiza a través de las redes, un espacio en el cual casi todo lo que contiene información; o puede transmitirla, debe ser incluido.

Cibernética, ciencia interdisciplinar que trata de los sistemas de comunicación y control en los organismos vivos, las máquinas y las organizaciones.

Cliente, aplicación que permite a un usuario obtener un servicio de un servidor localizado en la red. Sistema o proceso el cual le solicita a otro sistema o proceso la prestación de un servicio.

Comercio Electrónico, es la compra y venta de bienes y servicios realizados a través del internet, habitualmente con el soporte de plataformas y protocolos de seguridad estandarizados.

Cracker, delincuente informático que usa su conocimiento con fines maliciosos, antimorales o incluso bélicos, como intrusión de redes, acceso ilegal a sistemas gubernamentales, robo de información, distribuir material ilegal o moralmente inaceptable, fabricación de virus, herramientas de crackeo.

**D**

Data Mining (Minería de Datos), se encuentra dentro del conjunto de técnicas KDD (Knowledge Discovery in Databases). Se puede definir como "extracción no trivial de información implícita, desconocida previamente, y potencialmente útil desde los datos", y consiste en el conjunto de técnicas avanzadas para la extracción de información escondida en grandes bases de datos.

Data Warehouse (Almacén de datos), es una colección de datos orientadas a un dominio, integrado, no volátil y varía en el tiempo que ayuda a la toma de decisiones de la empresa u organización. Es un expediente de una empresa más allá de la información transaccional y operacional, almacenado en una base de datos diseñada para favorecer análisis y la divulgación eficientes de datos.

E

Encriptar (Cifrado), la encriptación es el procedimiento por medio del cual se convierte un texto, mensaje, archivo, documento, etc., de su formato original a otro ilegible o ininteligible para quien no posea la clave necesaria para revertir el proceso.

F

FTP File Transfer Protocol (Protocolo de Transferencia de Archivos), es uno de los diversos protocolos de la red Internet, ideal para transferir grandes bloques de datos por la red.

H

Hacking, se basa en ejecutar programas para la obtención de contraseñas, cuentas bancarias,

programas, virus, etc., también en elaborar programas nuevos que nunca acabara, es una lucha constante de información e innovar.

Hardware, equipo utilizado para el funcionamiento de una computadora. El *hardware* se refiere a los componentes materiales de un sistema informático.

Hiperdocumento, es el contenido de información, incluyendo los fragmentos de información y las conexiones entre esos fragmentos, indiferente sea el sistema utilizado para leer o escribir tal documento.

Hipermedia, en informática, integración de gráficos, sonido y vídeo en cualquier combinación para formar un sistema de almacenamiento y recuperación de información relacionada y de control de referencias cruzadas.

Hipertexto, en informática, método de presentación de información en el que el texto, las imágenes, los sonidos y las acciones están unidos mediante una red compleja y no secuencial de asociaciones que permite al usuario examinar los distintos temas, independientemente del orden de presentación de los mismos.

HTTP HyperText Transfer Protocol (Protocolo de Transferencia de Hipertexto), es el protocolo usado en cada transacción de la Web.

HTTPS, versión segura del protocolo HTTP. El sistema HTTPS utiliza un cifrado basado en las Secure Socket Layers (SSL) para crear un canal cifrado más apropiado para el tráfico de información sensible que el protocolo HTTP.



Host, servidor que provee de la información que se requiere para realizar algún procedimiento desde una aplicación cliente a la que se tenga acceso de diversas formas (SSH, FTP, WWW, email, etc.). Al igual que cualquier computadora conectada a Internet, debe tener una dirección o número IP y un nombre.

Hosting, el servicio de web hosting consiste en el almacenamiento de datos, aplicaciones o información dentro de servidores diseñados para llevar a cabo esta tarea. Los servidores a su vez se deben colocar en edificios o estructuras denominadas data centers, con su debida planta eléctrica, seguridad y conectividad con los mayores proveedores de telecomunicaciones (backbones) del mundo, para poder ofrecer buen ancho de banda.

I

IMAP Internet Message Access Protocol (Protocolo de Red de Acceso a Mensajes Electrónicos Almacenados en un Servidor), mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet.

Interfase (Interfaz), zona de contacto o conexión entre dos componentes de "hardware"; entre dos aplicaciones; o entre un usuario y una aplicación. Apariencia externa de una aplicación informática.

Internet. Es una red de redes a escala mundial de millones de computadoras interconectadas con el conjunto de protocolos TCP/IP.

L

Lenguaje de Definición de Datos (DDL), es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios de la misma llevar a cabo las tareas de descripción de las estructuras que almacenarán los datos y de procedimientos o funciones que permitan consultarlos.

Lenguaje de Manipulación de Datos (DML), es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos adecuado.

M

Modelo Entidad-Relación, es el modelo conceptual más utilizado para el diseño conceptual de bases de datos, está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

Módem, equipo utilizado para la comunicación de computadoras a través de líneas analógicas de transmisión de datos. El módem convierte las señales digitales del emisor en otras analógicas susceptibles de ser enviadas por teléfono.

Mundo Virtual, corresponde a un conjunto de objetos (2D ó 3D) donde una persona, representada por un avatar, puede interactuar.

**N**

Navegador (Browser), aplicación para visualizar todo tipo de información y navegar por el www con funcionalidades plenamente multimedia. Como ejemplo de navegadores tenemos Internet Explorer, Firefox y Safari (Mac).

NNTP Network News Transport Protocol (Protocolo de Transferencia de Noticias), es un Protocolo de red usado para subir y bajar, así como para transferir artículos entre servidores.

R

Realidad virtual, sistema que permite a uno o más usuarios ver, moverse y reaccionar en un mundo simulado por una computadora.

S

Script. Secuencia de comandos que se le dan a un módem con el propósito de configurarlo (velocidad, compresión de datos, etc.) o para realizar tareas específicas (llamar al proveedor, colgar, etc.). A veces es necesario modificar un script o cadena de inicio la cual establece las condiciones iniciales del módem (por ejemplo cambiar ATDT que establece una línea telefónica por tonos a ATDP que indica una línea telefónica por pulsos, etc.).

Sistema Gestor de Bases de Datos (SGBD), es un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

Sistema Operativo, software básico que controla una computadora. El sistema operativo tiene tres grandes funciones: coordina y manipula el *hardware* de la computadora, como la memoria, las impresoras, las unidades de disco, el teclado o el *mouse*; organiza los archivos en diversos dispositivos de almacenamiento, como discos flexibles, discos duros, discos compactos o cintas magnéticas, y gestiona los errores de *hardware* y la pérdida de datos.

SMTP Simple Mail Transfer Protocol (protocolo simple de transferencia de correo electrónico), protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras o distintos dispositivos.

SNMP Simple Network Management Protocol (Protocolo Simple de Administración de Redes), es el protocolo de gestión de red más importante y usado en la actualidad. Forma parte del conjunto de protocolos TCP/IP y está definido en la capa de aplicación del mismo. SNMP busca la sencillez y es por ello que en la capa de transporte está soportado por el protocolo UDP (caracterizado por su rapidez y su falta de fiabilidad) a través del puerto 170.

Software, programas de computadoras. Son las instrucciones responsables de que el *hardware* (la máquina) realice su tarea.

SSL (Secure Socket Layer). Protocolo creado por Netscape con el fin de hacer posible la transmisión encriptada y por ende segura, de información a través de la red donde sólo el servidor y el cliente podrán entender un determinado texto. Utiliza una



llave de 50 hasta 128 bits (más bits, mayor el grado de encriptación de la data). El navegador del cliente dictamina el rango.

Stock. Representa el almacenamiento de insumos directos e indirectos y/o productos terminados a la espera de consumirse en el proceso de producción, servicios, mantenimiento y venta en un tiempo más o menos cercano, el objetivo es abastecer en el momento oportuno, en la cantidad suficiente, con la calidad requerida y la financiación adecuada, las demandas originadas por el proceso de producción por la comercialización del producto.

T

TCP/IP acrónimo de *Transmission Control Protocol/Internet Protocol*, protocolos usados para el control de la transmisión en Internet. Permite que diferentes tipos de computadoras se comuniquen a través de redes heterogéneas.

Tienda virtual. Página Web donde se pueden realizar compras en línea.

U

URL, acrónimo de *Universal Resource Locator*, método de identificación de documentos o lugares en Internet, que se utiliza principalmente en World Wide Web (WWW). Un URL es una cadena de caracteres que identifica el tipo de documento, la computadora, el directorio y los subdirectorios en el que se encuentra el documento y su nombre.

V

Virtual, término de frecuente utilización en el mundo de las tecnologías de la información y de

las comunicaciones el cual designa dispositivos o funciones simulados.

W

Web, forma abreviada de referirse a la World Wide Web, se utiliza para denominar uno de los servicios más importantes de la red Internet. Son páginas que utilizan un lenguaje especial llamado HTML, que permite presentar en pantalla texto y gráficos en el formato deseado.

X

XML Extensible Markup Language (Lenguaje de Mercado Extensible), es un lenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).



REFERENCIAS BIBLIOGRÁFICAS

- [1] Annwn Page, Khristine. (Mayo de 2004), *"Dreamweaver MX 2004 Versión Dual"*. Anaya Multimedia.
- [2] Bonney, Sean. (Noviembre de 2001), *"3ds Max 4 Magic"*. Buenos Aires. Prentice Hall, Primer Edición.
- [3] Booch, Grady, Jacobson, Ivar, Runbaugh, James. (1999), *"El Lenguaje Unificado de Modelado"*. Madrid. Addison Wesley.
- [4] Booch, Grady, Jacobson, Ivar, Runbaugh, James. (2000), *"El Proceso Unificado de Desarrollo de Software"*. Madrid. Addison Wesley.
- [5] Cabezas Granado, Luís Miguel. (Noviembre de 2004). *"PHP 4"*. Anaya Multimedia, Primer Edición.
- [6] Cebolla, Castell. (2002), *"3ds Max 4 Curso Práctico"*. México. Alfaomega-rama.
- [7] Christerson, M., Jacobson, I., Jonsson, P., Overgaard, G. (1992), *"OO Software Engineering, a Use Case Driven Approach"*. Reading, Massachusetts. Addison-Wesley.
- [8] Flanagan, David. *"JavaScript Definitive Guide"*. O'Reilly & Associates
- [9] García Álvarez, Rubén. (2001), *"Guías Visuales Flash 5"*. España. Anaya Multimedia.



- [10] Gilfillan, Ian. (Julio de 2003), *“La Biblia de MySQL”*. Anaya Multimedia.
- [11] Guerrero, Javier. *“Vrml 2.0 El Lenguaje 3d De Internet”*. Abeto.
- [12] Jamsa. *“Vrml Biblioteca Del Programador”*. Mcgraw-Hill
- [13] Kabir, Mohammed J. *“La Biblia del Servidor Apache 2”* Anaya Multimedia-Anaya Interactiva.
- [14] Karlins, David. (Julio de 2001), *“Macromedia Flash 5: Anime sus páginas web”*. Buenos Aires. Prentice Hall, Primer Edición.
- [15] Lemay, Laura. (1998), *“Aprende HTML 4 para Web en una semana”*. México. Prentice Hall, Tercera Edición.
- [16] Matossian, Michele. (2000), *“3D STUDIO MAX Guía de aprendizaje”*. Madrid. Prentice Hall, Primera Edición.
- [17] Meloni, Julie C. *“PHP FAST AND EASY WEB DEVELOPMENT.”* Thomson Learning
- [18] Pardo Niebla, Miguel. (2002), *“Guías Visuales Creación de páginas Web”*. España. Anaya Multimedia.
- [19] Pascual González, Francisco. (Mayo de 2004), *“Navegar en Internet: Macromedia Dreamweaver MX 2004”*. Ra-ma.
- [20] Pavón, J. (Septiembre de 2004), *“Creación de un Portal con PHP y MySQL”*. RA-MA



- [21] Pérez, C. (Diciembre de 2003), *“My SQL para Windows y Linux”*. RA-MA
- [22] Rasmus Lerdorf, Kevin Tatroe. (Abril de 2002), *“PROGRAMMING PHP”*. O'REILLY & ASSOCIATES
- [23] Soria, Ramón. (1999), *“Navegando en Internet Diseño y Creación para páginas Web, HTML 4”*. México. Alfaomega-rama.

REFERENCIAS ELECTRÓNICAS

- [24] *“AB-Shop”*. URL: <http://www.ab-shop.com/>
- [25] Atanasio Negrete, Fernando. (Octubre 19 de 2003), *“Apache + PHP + MySql + PhpMyAdmin como módulo de Apache”*.
URL: <http://www.maestrosdelweb.com/editorial/phpmysqlap/>
- [26] Cárdenas, Lola, Gracia, Joaquín. (Enero, 3 de 2003), *“JavaScript”*.
URL: <http://www.webestilo.com/javascript/js00.phtml>
- [27] Cisco Systems, Inc. *“Comercio Electrónico”*.
URL:
http://www.cisco.com/global/ES/solutions/ent/bus_solutions/ec_home.shtml
- [28] *“Comercio Electrónico”*. URL: <http://www.asenmac.com/ebusiness.htm>



- [29] Conferencia de las Naciones Unidas sobre Comercio y Desarrollo. (Mayo 3 de 2002), *“Estrategias en materia de comercio electrónico para el desarrollo: elementos básicos de un entorno propicio para este comercio”*.
URL: <http://www.unctad.org/sp/docs/c3em15d2.sp.pdf>
- [30] *“Cube Stores”*. URL: <http://www.cubestores.com/es/>
- [31] *“Ecommerce”*.
URL: http://www.google.com.mx/search?q=cache:1bLOsWE0_hoJ:ecommerce.wipo.int/meetings/1999/papers/zarate-es.html+antecedentes+del+comercio+electr%C3%B3nico&hl=es
- [32] *“El Corte Inglés Supermercado”*.
URL:
<https://supermercado.elcorteingles.es/secciones/entrada/nolidentificado.asp>
- [33] Escalona, María José, Koch, Nora. (Diciembre de 2003), *“Ingeniería de Requisitos en Aplicaciones para la Web – Un estudio comparativo”*
URL: <http://www.lsi.us.es/docs/informes/LSI-2002-4.pdf>
- [34] *“e-Shop”*. URL: <http://www.eshop.es/>
- [35] Esteban Martí, José Ramón. (Julio, 7 de 2001), *“Servidor de Correo”*.
URL: <http://www.internautas.org/index.php?op=1&id=167>
- [36] Iriarte Ahon, Erick. *“Comercio Electrónico en América Latina Realidades y Perspectivas”*.
URL: http://lac.derechos.apc.org/investigacion/comercio_electronico.doc



- [37] “La Playa”. URL: <http://www.superlaplaya.com.mx/>
- [38] La Primera Telecomunicación. “Tienda Virtual”.
URL: http://www.laprimera.net/modules.php?name=News&new_topic=14
- [39] “My Brands”. URL: <http://www.mybrandsinc.com/ShopOnline/catalog.asp>
- [40] “Neostore”. URL: <http://www.neostore.com.ar/>
- [41] Prats, Alberto. “Curso de introducción al VRML”.
URL: <http://www.med.ub.es/~aprats/curvrml/html/cvrml03.htm>
- [42] Quinteros, Jorge Nicolás. (2003), “Metodologías para el desarrollo de Sitios Web”. URL: <http://www.virtual.unlar.edu.ar/catedras-virtuales/seminario-de-act-tecnologica-3/art-disc-x-alum/014.pdf>
- [43] Ramos Salavert, Isidro, Fernández Caballero, Antonio, Lozano Pérez, Maria Dolores. “Tendencias Actuales en la Interacción Persona-Ordenador: Accesibilidad, Adaptabilidad y Nuevos Paradigmas”
URL: <http://www.info-ab.uclm.es/personal/caballer/download/papers/CursoVerano2003.pdf>
- [44] “SACE (Sistema de Administración de Comercio Electrónico)”.
URL: <http://www.saceonline.com/>
- [45] “Sumercado.com”. URL: <http://www.sumercado.com/Lima/abarrotes.htm>
- [46] Talens Oliag, Sergio, Hernández Orallo, José. (Febrero 4 de 1997), “HTML Manual de Referencia”. URL: <http://www.ivia.es/htmlref/s1.html>



- [47] *"Tienda Virtual"*.
URL: <http://www.teide.net/scripts/motor.asp?file=/tiendavirtual.htm>
- [48] *"Trico"*. URL: <http://www.trico.com.mx/asp/>
- [49] Universidad de Navarra, Centro de Tecnología Informática. (Septiembre 9 de 2000), *"Introducción al 3D Studio Max"*.
URL: <http://www.unav.es/cti/manuales/3DStudioMax/indice.html>
- [50] Universidad de Navarra. (Septiembre de 1998), *"Manual de HTML"*.
URL: <http://www.unav.es/cti/manuales/HTML/indice.html>
- [51] Universidad de Navarra, Centro de Tecnología Informática. "Tutorial de JavaScript". URL: <http://www.unav.es/cti/manuales/TutorialJavaScript/indices/>
- [52] Valderas Aranda, Pedro J. (Julio 2004), *"Especificación de Requisitos en el desarrollo de Aplicaciones Web"*.
URL:
<http://oomethod.dsic.upv.es/anonimo/..%5Cfiles%5CTechnicalReport%5Creq.pdf>
- [53] Viaplustech.com. (Octubre 16 de 2003), *"Cómo vender en Internet: introducción al e-commerce"*.
URL: <http://www.mailxmail.com/curso/empresa/ecommerce>



CÓDIGO FUENTE

Código para Generar el Carrito de Compra

<?

```
class carrito {
    //Atributos de la clase
    var $num_productos;
    var $array_id_prod;
    var $array_nombre_prod;
    var $array_presentacion;
    var $array_precio_prod;
    var $array_cantidad;
    var $array_id;
    var $array_pedido;
    //Constructor. Realiza las tareas de inicializar los objetos cuando se instancian inicializa el numero de productos
    //a 0
    function carrito () {
        $this->num_productos=0;
    }
    //Introduce un producto en el carrito. Recibe los datos del producto. Se encarga de introducir los datos en los
//arrays del objeto carrito, luego aumenta en 1 el numero de productos
    function introduce_producto($id_producto,$Nombre,$Presentacion,$Precio_uni,$cantidad){
        $this->array_id_prod[$this->num_productos]=$id_producto;
        $this->array_nombre_prod[$this->num_productos]=$Nombre;
        $this->array_presentacion[$this->num_productos]=$Presentacion;
        $this->array_precio_prod[$this->num_productos]=$Precio_uni;
        $this->array_cantidad[$this->num_productos]=$cantidad;
        $this->num_productos++;
    }
    //Muestra el contenido del carrito de la compra, además pone los enlaces para eliminar un producto del carrito
    function imprime_carrito(){
        $suma = 0;
        $scanti = 0;
        echo '<table border=1 cellpadding="2.5">
            <tr><td bgcolor=#dcd8fc><b>Nombre producto</b></td>
            <td bgcolor=#dcd8fc><b>Presentación</b></td>
            <td bgcolor=#dcd8fc><b>Precio</b></td>
            <td bgcolor=#dcd8fc><b>Cantidad</b></td>
            <td bgcolor=#dcd8fc>&nbsp;</td></tr>';
        for ($i=0;$i<$this->num_productos;$i++){
            if($this->array_id_prod[$i]!=0){
                if($this->array_cantidad[$i]!=NULL){
                    echo '<tr>';
                }
            }
        }
    }
}
```



```
echo "<td bgcolor=#f9cafb>" . $this->array_nombre_prod[$i] . "</td>";
echo "<td bgcolor=#f9cafb>" . $this->array_presentacion[$i] . "</td>";
echo "<td bgcolor=#f9cafb>" . $this->array_precio_prod[$i] . "</td>";
echo "<td bgcolor=#f9cafb>" . $this->array_cantidad[$i] . "</td>";
echo "<td bgcolor=#f9cafb><a href='eliminar_producto.php?linea=$i'>Eliminar producto</td>";
echo '</tr>';

$canti=$this->array_precio_prod[$i]*$this->array_cantidad[$i];
$suma += $canti;
}
}
}

//Muestra el total
echo "<tr><td bgcolor=#f9cafb><b>TOTAL:</b></td><td bgcolor=#f9cafb> <b>$suma</b></td><td
bgcolor=#f9cafb>&nbsp;</td></tr>";

//Total más IVA
echo "<tr><td bgcolor=#f9cafb><b>IVA (15%):</b></td><td bgcolor=#f9cafb> <b>" . $suma * 1.15 .
"</b></td><td bgcolor=#f9cafb>&nbsp;</td></tr>";
echo "</table>";

}

//Llena la tabla Pedido de la base de datos
function llenar_bd($id){
    $fecha = date("d/m/Y");
    echo $fecha;
    $idcli =$id;
    echo $id;
    $connectid = mysql_connect("localhost","isamiri","mica1982");
    mysql_select_db("fin",$connectid);
    $ssql=mysql_db_query("fin","insert into pedido (id_cliente, id_estado_pedido, Fecha) values ('$id','1','$fecha)");
    $ultimo_id2 = mysql_insert_id($connectid);
    $this->$array_pedido=$ultimo_id2;
    mysql_query($ssql);
}

//Redirecciona el id del pedido
function id_pedido(){
    $id=$this->$array_pedido;
    header ("Location: temporal.php?id=$id");
}

//Obtiene el id del pedido, actualiza las existencias y el inventario
function llenar_paquete($id){
    mysql_connect("localhost","isamiri","mica1982");
    $result=mysql_db_query("fin","select pedido.id_pedido FROM pedido WHERE pedido.id_cliente=$id");
    while ($row=mysql_fetch_array($result))
    {
        $p=$row["id_pedido"];
    }
}
```



```
}
for ($i=0;$i<$this->num_productos;$i++){
    if($this->array_id_prod[$i]!=0){
        if($this->array_cantidad[$i]!=NULL){
            echo '<tr>';
            $a= $this->array_id_prod[$i] ;
            $b= $this->array_cantidad[$i];
            $sSQL="Select Existencias From producto WHERE id_producto='$a'";
            $result=mysql_db_query("fin",$sSQL);
            while ($row=mysql_fetch_array($result))
            {
                echo '<td>'.$row["Existencias"].'</td>';
                $existencias= $row["Existencias"];
            }
            $res=$existencias-$b;
            $sSQL="Update producto Set Existencias='$res' Where id_producto='$a'";
            $ssql=mysql_db_query("fin","insert into paquete (id_cliente, id_producto,Cantidad,id_pedido) values
            ('$id','$a','$b','$p')");
            $fec = date("d/m/Y");
            $ssql1=mysql_db_query("fin","insert into inventario
            (id_cliente,id_producto,Salidap,id_pedido,Fecha_inv) values ('$id','$a','$b','$p','$fec')");
            mysql_db_query("fin",$sSQL);
            mysql_query($ssql);
            mysql_query($ssql1);
        }
    }
}

//Elimina un producto del carrito. Recibe la línea del carrito que debe eliminar, no lo elimina realmente, simplemente pone a
//cero el id, para saber que esta en estado retirado
function elimina_producto($linea){
    $this->array_id_prod[$linea]=0;
    $this->array_cantidad[$linea]=0;
}

//Elimina todos los productos del carrito
function cancelar(){
    for($i=0;$i<=$this->num_productos;$i++){
        $this->array_id_prod[$i]=0;
        $this->array_cantidad[$i]=0;
    }
}

//inicio la sesión
ob_start();
```



```
session_start();  
//si no esta creado el objeto carrito en la sesion, lo crea  
if (!isset($_SESSION["ocarrito"])){  
    $_SESSION["ocarrito"] = new carrito();  
} ?>
```

Código para Generar la Nota del Pedido de Forma Dinámica

```
<?  
define('FPDF_FONTPATH','font');  
require('fpdf.php');  
class PDF extends FPDF  
{  
//Cabecera de página  
function Header()  
{ //Logo  
    $this->Image('logo1.png',10,10,30,0);  
//Arial bold 15  
    $this->SetFont('Arial','B',16);  
//Movernos a la derecha  
    $this->Cell(80);  
//Título  
    $this->SetTextColor(0,0,210);  
    $this->Cell(30,10,'TAV-HIDALGO',0,0,'C');  
    $this->Ln(8);  
    $this->SetFont('Arial','U',14);  
    $this->Cell(80);  
    $this->SetTextColor(70,87,240);  
    $this->Cell(30,10,'http://www.tavhidalgo.com',0,0,'C');  
    $this->Ln(8);  
    $this->SetFont('Arial','I',14);  
    $this->Cell(50);  
    $this->Cell(30,10,'Quejas y sugerencias: ',0,0,'C');  
    $this->SetFont('Arial','U',14);  
    $this->Cell(80,10,'tavhidalgo@yahoo.com.mx',0,0,'C');  
    $this->SetFont('Arial','U',14);  
    $this->Ln(8);  
    $this->SetFont('Times','I',14);  
    $this->Cell(50);  
    $this->Ln(50);  
}  
//Pie de página  
function Footer()  
{
```



//Posición: a 1,5 cm del final

```
$this->SetY(-25);
$this->SetFont('Arial','B',8);
$this->Cell(80);
$this->Cell(30,10,'Gracias por comprar con nosotros',0,0,'C');
$this->Ln(5);
$this->Cell(80);
$fecha = date("d/m/Y h:i");
$this->Cell(30,10,".$fecha.",0,0,'C');
$this->Ln(5);
```

//Arial italic 8

```
$this->SetFont('Arial','I',8);
```

//Número de página

```
$this->Cell(0,10,'Página '.$this->PageNo().'\{nb}',0,0,'C');
```

```
}
```

```
}
```

//Creación del objeto de la clase heredada

```
$pdf=new PDF();
$pdf->AliasNbPages();
$pdf->AddPage();
mysql_connect("localhost","isamiri","mica1982");
```

//Selecciona los productos que se muestran en el archive pdf

```
$result=mysql_db_query("fin","select producto.Nombre_producto,producto.Precio_uni,paquete.Cantidad from producto, paquete
WHERE paquete.id_producto=producto.id_producto AND paquete.id_pedido='$id' ORDER BY Nombre_producto");
$column_pedido = "";
$column_code1 = "";
$column_code = "";
$column_name = "";
$column_price = "";
$column_pricetot = "";
$total = 0;
```

//Para cada producto se agrega una columna

```
while($row = mysql_fetch_array($result))
{
    $name = $row["Nombre_producto"];
    $code = $row["Cantidad"];
    $code1 = $row["Precio_uni"];
    $code2= $code*$code1;
    $real_price = $code2;
    $column_code = $column_code.$code."\n";
    $column_name = $column_name.$name."\n";
    $column_price = $column_price.$code1."\n";
    $column_pricetot = $column_pricetot.$code2."\n";
```



//Se suman todos los precios de los productos

```
$total = $total+$real_price;
}
mysql_connect("localhost","isamiri","mica1982");
$result=mysql_db_query("fin","select cliente.Nombre,cliente.Apell_pat,cliente.Apell_mat, paquete.id_cliente from cliente,paquete
WHERE paquete.id_pedido='$id' AND paquete.id_cliente=cliente.id_cliente");
while($row = mysql_fetch_array($result))
{
    $pedido = $id;
    $cli = $row["Nombre"];
    $p=$row["Apell_pat"];
    $m=$row["Apell_mat"];
}
$column_code1 = $column_code1.$cli.' '.$p.' '.$m.'";
$column_pedido = $column_pedido.$pedido."`n";
mysql_free_result($result);
mysql_close();
```

//Se pone la posición de los campos

```
$Y_Fields_Name_position = 55;
```

//Posición de la tabla debajo del nombre de los campos

```
$Y_Table_Position = 61;
```

//Primero se crea cada campo con su nombre

```
$pdf->SetFillColor(224,235,255);
$pdf->SetFont('Arial','B',14);
$pdf->SetY($Y_Fields_Name_position);
$pdf->SetFont('Arial','I',12);
$pdf->SetY($Y_Fields_Name_position);
$pdf->SetX(25);
$pdf->Cell(50,6,'Número de Pedido:','C');
$pdf->SetX(75);
$pdf->Cell(80,6,$column_pedido,'C');
$pdf->SetFont('Arial','I',12);
$pdf->SetY($Y_Table_Position);
$pdf->SetX(25);
$pdf->MultiCell(50,6,'Nombre del Cliente:','C');
$pdf->SetY($Y_Table_Position);
$pdf->SetX(75);
$pdf->MultiCell(80,6,$column_code1,'C');
$pdf->SetY($Y_Table_Position);
$Y_Fields_Name_position = 70;
$Y_Table_Position = 76;
$pdf->SetFillColor(224,235,255);
$pdf->SetFont('Arial','B',12);
$pdf->SetY($Y_Fields_Name_position);
```



```
$pdf->SetX(25);
$pdf->Cell(22,6,'CANT',1,0,'C',1);
$pdf->SetX(45);
$pdf->Cell(70,6,'PRODUCTO',1,0,'C',1);
$pdf->SetX(115);
$pdf->Cell(30,6,'PRECIO UNI',1,0,'C',1);
$pdf->SetX(145);
$pdf->Cell(30,6,'TOTAL',1,0,'C',1);
$pdf->Ln();
//Muestra tres columnas
$pdf->SetFont('Arial','B',16);
$pdf->SetY($Y_Table_Position);
$pdf->SetX(25);
$pdf->MultiCell(90,6,$column_code,1);
$pdf->SetY($Y_Table_Position);
$pdf->SetX(45);
$pdf->MultiCell(100,6,$column_name,1);
$pdf->SetY($Y_Table_Position);
$pdf->SetX(115);
$pdf->MultiCell(30,6,$column_price,1,'R');
$pdf->SetY($Y_Table_Position);
$pdf->SetX(145);
$pdf->MultiCell(30,6,$column_pricetot,1,'R');
$pdf->SetX(145);
$pdf->MultiCell(30,6,'$.total,1,'R');
//Crea las líneas de las cajas para cada fila de cada producto
$i = 0;
$pdf->SetY($Y_Table_Position);
while ($i < $number_of_products)
{
    $pdf->SetX(45);
    $pdf->MultiCell(120,6,"",1);
    $i = $i + 1;
}
$pdf->Output();
?>
```

Código para Generar el Formulario para Autenticar al Usuario Navegante

```
<html><style type="text/css">
<!--
body { background-color: #fce3fd;}
-->
</style>
```



```
<body>
<form method = "POST" action = "aut.php">
<table align="center" width="250" cellspacing="2" cellpadding="2" border="0">
<tr><td colspan="2" align="center"
  <?if ($_GET["errorusuario"]=="si"){?>
    bgcolor=#990AA0><span style="color:ffffff"><b>Datos incorrectos</b></span>
  <?}else{?>
    bgcolor=#990AA0><span style="color:ffffff"><b>Introduce tus Datos</b></span>
  <?}?></td>
</td></tr><tr> <td align="left">id de Cliente:</td>
<td><input type="text" name="buscar" size="20"></td></tr><tr>
<td align="left">Password:</td>
<td><input type="password" name="buscar1" size="20"></td></tr>
<tr> <td colspan="2" align="right"><input type="submit" value="Aceptar"></td></tr>
<table align="center"><td><a href='Inser.php'>Registrarse</td>
<table align="center"><td><a href='norecuerdo.php'>No recuerdo mi password</td>
</table>
</form>
</body>
</html>
```

Código para Autenticar al Usuario Navegante

```
<?
echo $id_producto;
//Conección con la base de datos
$conn = mysql_connect("localhost","isamiri","mica1982");
//Selección de la BBDD
mysql_select_db("fin",$conn);
//Sentencia SQL para buscar un usuario con esos datos
$query = "SELECT * FROM cliente WHERE id_cliente='$buscar' and Pasword='$buscar1'";
//Se ejecuta la sentencia
$rs = mysql_query($query,$conn);
//Se verifica si el usuario y contraseña son válidos, si la ejecución de la sentencia SQL nos da algún resultadoes que si
//que existe esa combinación usuario/contraseña
if (mysql_num_rows($rs)!=0){
  //usuario y contraseña válidos se define una sesión y se guardan los datos
  session_start();
  session_register("autenticado");
  $autenticado = "SI";
  header ("Location: confirmar.php?idcli=$buscar");
}else{
  header("Location: formulario.php?errorusuario=si");
}
```



```
mysql_free_result($rs);
mysql_close($conn);
?>
```

Código para Generar el Formulario de Registro del Cliente

```
<HTML>
<HEAD>
<TITLE>INSERTAR.HTML</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></HEAD>
<BODY bgcolor="#fce3fd">
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Datos del Usuario</font></p>
<FORM METHOD="POST" ACTION="actCli.php">
<table align="center" border="0">
    <td align="left" width="150"><font size=2>Nombre:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre" size="20" maxlength="100"></td></tr>
    <tr><td align="left" width="150"><font size=2>Apellido Paterno:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_pat"></td></tr>
    <tr><td align="left" width="150"><font size=2>Apellido Materno:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_mat"></td></tr>
    <tr><td align="left" width="150"><font size=2>Fecha de nacimiento:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="Fecha_nac"></td></tr>
    <tr><td align="left" width="150"><font size=2>RFC:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="RFC"></td></tr>
    <tr><td align="left" width="150"><font size=2>Correo:</font></td>
    <td align="left" width="150"><INPUT TYPE="TEXT" NAME="Correo" VALUE="@></td></tr>
    <tr><td align="left" width="150"><font size=2>Password:</font></td>
    <td align="left" width="150"><INPUT TYPE="password" NAME="Pasword"></td></tr>
</table>
<?
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
//Sentencia SQL
echo'<td align="left" width="150"><font size=2>Nivel de Estudios:</font></td>' ;
$$SQL="Select id_nivel_estudio,Nivel From nivel_estudios";
$result=mysql_db_query("fin",$SQL);
echo '<td align="left" width="150"><select name="niv">';
//Lista desplegable
while ($row=mysql_fetch_array($result))
{
    echo '<option value='.$row["id_nivel_estudio"].'>'.$row["Nivel"] . "</option>";
} '</td>';
?>
```



```
<tr></tr><td><font size=2>Estado civil:</font></td>
<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" CHECKED>Soltero
(a)</td><tr></tr></font><td></td>
<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2">Casado (a)</td></font><tr>
<td align="left" width="150"><font size=2>Teléfono:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Telefono"></td></tr>
<tr><td align="left" width="150"><strong><font size=2>Dirección:</font></strong></td></tr>
<tr><td align="left" width="150"><font size=2>Calle:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_calle"></td></tr>
<tr><td align="left" width="150"><font size=2>Número:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Numero"></td></tr>
<tr><td align="left" width="150"><font size=2>Colonia:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_colonia"></td></tr>
<tr><td align="left" width="150"><font size=2>Código Postal:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="CP"></td></tr>
<?>
```

//Conexion con la base de datos

```
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Municipio:</font></td>' ;
```

//Sentencia SQL

```
$$SQL="Select id_municipio,Nombre_municipio From municipio";
$result=mysql_db_query("fin",$$SQL);
echo '<td align="left" width="150"><select name="municipio">';
while ($row=mysql_fetch_array($result))
{
    echo '<option value='.$row["id_municipio"].">".$row["Nombre_municipio"] . "</option>";
}'</td>';
mysql_free_result($result)
?>
<?>
```

//Conexion con la base de datos

```
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Estado:</font></td>' ;
```

//Sentencia SQL

```
$$SQL="Select id_estado,Nombre_estado From estado";
$result=mysql_db_query("fin",$$SQL);
echo '<td align="left" width="150"><select name="estado">';
while ($row=mysql_fetch_array($result))
{
    echo '<option value='.$row["id_estado"].">".$row["Nombre_estado"] . "</option>";
}'</td>';
mysql_free_result($result)
?>
</select>
```



```

<tr><td><td>
<tr><td><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Enviar"></td></td></tr></td></tr>
</table>
</FORM>
</BODY>
</HTML>

```

Código para Validar que los Campos del Formulario del Registro de Usuario no estén Vacíos

```

<HTML>
<HEAD>
<TITLE>actualizar.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></HEAD>
<BODY bgcolor="#fce3fd">
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Verificar Datos</font></p>
<?
$valo=1;
$valor=2;
$Compara=$municipio;
$Compara1=$estado;
$Compara2=$niv;
$a=strlen($Nombrec);
$b=strlen($Apell_pat);
$c=strlen($Apell_mat);
$d=strlen($Fecha_nac);
$e=strlen($RFC);
$f=strlen($Correo);
$g=strlen($Password);
$h=strlen($Telefono);
$i=strlen($Nombre_calle);
$j=strlen($Numero);
$k=strlen($Nombre_colonia);
$l=strlen($CP);
?>
<?if($a!=0 && $b!=0 && $c!=0 && $d!=0 && $e!=0 && $f!=0 && $g!=0 && $h!=0 && $i!=0 && $j!=0 && $k!=0 && $l!=0){?>
<FORM METHOD="POST" ACTION="id_cli.php">
<table align="Center" border="0">
<td align="left" width="150"><font size=2>Nombre:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombrec" VALUE="<?echo "".$Nombrec;?>"size="20"
maxlength="100" "></td>

```



```
<td align="left" width="150"></td>
<tr><td align="left" width="150"><font size=2>Apellido Paterno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_pat" VALUE="<?echo "".$Apell_pat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Apellido Materno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_mat" VALUE="<?echo "".$Apell_mat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Fecha de Nacimiento:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Fecha_nac" VALUE="<?echo "".$Fecha_nac;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>RFC:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="RFC" VALUE="<?echo "".$RFC;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Correo:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Correo" VALUE="<?echo "".$Correo;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Password:</font></td>
<td align="left" width="150"><INPUT TYPE="password" NAME="Pasword" VALUE="<?echo "".$Pasword;?>"></td></tr>
<?>
```

//Conexion con la base de datos

```
mysql_connect("localhost","isamiri","mica1982");
```

//Sentencia SQL

```
echo '<tr><td align="left" width="150"><font size=2>Nivel de Estudios:</font></td>' ;
```

```
$$SQL="Select id_nivel_estudio, Nivel From nivel_estudios";
```

```
$result=mysql_db_query("tiendota",$sSQL);
```

```
echo '<td align="left" width="150"><select name="niv">';
```

//Lista desplegable

```
while ($row=mysql_fetch_array($result))
```

```
{
```

```
    if ($Compara2==$row["id_nivel_estudio"])
```

```
    {
```

```
        echo '<option value='.$row["id_nivel_estudio"]." SELECTED>".$row["Nivel"] . "</option>";
```

```
    }else{
```

```
        echo '<option value='.$row["id_nivel_estudio"].">".$row["Nivel"] . "</option>";
```

```
    }
```

```
}
```

```
'</td></tr>';
```

```
echo '<tr></tr>';
```

```
echo '<td><font size=2>Estado civil:</font></td>';
```

```
if($id_estado_civil==$valo)
```

```
{
```

```
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" CHECKED>Soltero
```

```
(a)</td><tr></tr></font>';
```

```
echo '<td></td>';
```

```
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2">Casado (a)</td></font>';
```

```
}
```

```
if($id_estado_civil==$valor){
```

```
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" >Soltero
```

```
(a)</td><tr></tr></font>';
```



```
echo '<td></td>';
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2" CHECKED>Casado
(a)</td></font>';
}
?>
<tr><td align="left" width="150"><b><font size=2>Dirección:</font></b></td></tr>
<tr><td align="left" width="150"><font size=2>Teléfono:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Telefono" VALUE="<?echo "".$Telefono;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Calle:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_calle" VALUE="<?echo
"".$Nombre_calle;?><?"".$id_calle;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Número:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Numero" VALUE="<?echo "".$Numero;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Colonia:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_colonia" VALUE="<?echo "".$Nombre_colonia;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Código Postal:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="CP" VALUE="<?echo "".$CP;?>"></td></tr>
<?
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Municipio:</font></td>' ;
$$SQL="Select id_municipio, Nombre_municipio From municipio";
$result=mysql_db_query("tiendota",$$SQL);
echo '<td align="left" width="150"><select name="municipio">';
while ($row=mysql_fetch_array($result))
{
    if ($Compara==$row["id_municipio"])
    {
        echo '<option value=.'.$row["id_municipio"].' SELECTED>.'.$row["Nombre_municipio"] . '</option>';
    }else{
        echo '<option value=.'.$row["id_municipio"].'>.'.$row["Nombre_municipio"] . '</option>';
    }
}
'</td></tr>';
?>
<?
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Estado:</font></td>' ;
$$SQL="Select id_estado, Nombre_estado From estado";
$result=mysql_db_query("tiendota",$$SQL);
echo '<td align="left" width="150"><select name="estado">';
while ($row=mysql_fetch_array($result))
{
    if ($Compara1==$row["id_estado"])
    {
```



```
        echo '<option value='.$row["id_estado"]." SELECTED>".$row["Nombre_estado"] . "</option>";
    }else{
        echo '<option value='.$row["id_estado"].">".$row["Nombre_estado"] . "</option>";
    }
}
'</td></tr>';
?>
<br><td> <td><tr><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Confirmar"></td></td></tr></td></td>
</table>
</FORM>
<?>else{?>
<table align="center" width="250" cellspacing="2" cellpadding="2" border="0">
<tr> <td colspan="2" align="center"
        bgcolor=#990AA0><span style="color:ffffff"><b>Existen campos vacios</b></span></td></tr>
<?echo"<FORM METHOD='POST' ACTION='actCli.php'>";?>
<table align="Center" border="0">
<td align="left" width="150"><font size=2>Nombre:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombrec" VALUE="<?echo "".$Nombrec;?>"size="20"
maxlength="100" "></td>
<td align="left" width="150"></td>
<tr><td align="left" width="150"><font size=2>Apellido Paterno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_pat" VALUE="<?echo "".$Apell_pat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Apellido Materno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apell_mat" VALUE="<?echo "".$Apell_mat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Fecha de Nacimiento:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Fecha_nac" VALUE="<?echo "".$Fecha_nac;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>RFC:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="RFC" VALUE="<?echo "".$RFC;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Correo:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Correo" VALUE="<?echo "".$Correo;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Password:</font></td>
<td align="left" width="150"><INPUT TYPE="password" NAME="Pasword" VALUE="<?echo "".$Pasword;?>"></td></tr>
<?
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Nivel de Estudios:</font></td>' ;
$$SQL="Select id_nivel_estudio, Nivel From nivel_estudios";
$result=mysql_db_query("tiendota",$$SQL);
echo '<td align="left" width="150"><select name="niv">';
while ($row=mysql_fetch_array($result))
{
    if ($Compara2==$row["id_nivel_estudio"])
    {
        echo '<option value='.$row["id_nivel_estudio"]." SELECTED>".$row["Nivel"] . "</option>";
    }else{
```



```
        echo '<option value='.$row["id_nivel_estudio"].">".$row["Nivel"] . "</option>";
    }
}
'</td></tr>';
echo '<tr></tr>';
echo '<td><font size=2>Estado civil:</font></td>';
if($id_estado_civil==$valo)
{
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" CHECKED>Soltero
(a)</td><tr></tr></font>';
echo '<td></td>';
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2">Casado (a)</td></font>';
}
if($id_estado_civil==$valor){
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" >Soltero
(a)</td><tr></tr></font>';
echo '<td></td>';
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2" CHECKED>Casado
(a)</td></font>';
}
if($id_estado_civil==NULL){
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" >Soltero
(a)</td><tr></tr></font>';
echo '<td></td>';
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2" >Casado (a)</td></font>';
}
?>
<tr><td align="left" width="150"><b><font size=2>Dirección:</font></b></td></tr>
<tr><td align="left" width="150"><font size=2>Teléfono:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Telefono" VALUE="<?echo "".$Telefono;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Calle:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_calle" VALUE="<?echo "".$Nombre_calle;?><?"".$id_calle;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Número:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Numero" VALUE="<?echo "".$Numero;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Colonia:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_colonia" VALUE="<?echo "".$Nombre_colonia;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Código Postal:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="CP" VALUE="<?echo "".$CP;?>"></td></tr>
<?
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Municipio:</font></td>' ;
$$SQL="Select id_municipio, Nombre_municipio From municipio";
$result=mysql_db_query("tiendota",$sSQL);
```



```
echo '<td align="left" width="150"><select name="municipio">';
while ($row=mysql_fetch_array($result))
{
    if ($Compara==$row["id_municipio"])
    {
        echo '<option value=.'$row["id_municipio"].' SELECTED>'.$row["Nombre_municipio"] . "</option>";
    }else{
        echo '<option value=.'$row["id_municipio"].'>'.$row["Nombre_municipio"] . "</option>";
    }
}
'</td></tr>';
?>
<?
mysql_connect("localhost","isamiri","mica1982");
echo'<tr><td align="left" width="150"><font size=2>Estado:</font></td> ' ;
$$SQL="Select id_estado, Nombre_estado From estado";
$result=mysql_db_query("tiendota",$sSQL);
echo '<td align="left" width="150"><select name="estado">';
while ($row=mysql_fetch_array($result))
{
    if ($Compara1==$row["id_estado"])
    {
        echo '<option value=.'$row["id_estado"].' SELECTED>'.$row["Nombre_estado"] . "</option>";
    }else{
        echo '<option value=.'$row["id_estado"].'>'.$row["Nombre_estado"] . "</option>";
    }
}
'</td></tr>';
?>
<br><td> <td><tr><td><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Confirmar"></td></tr></td></td>
</table>
</FORM>
<?}?>
</BODY>
</HTML>
```

Código para Insertar los Datos del Usuario en la Base de Datos

```
<TITLE>Id.php</TITLE>
<style type="text/css">
<!--
body {
    background-color: #fce3fd;
}
```



```
-->
</style></HEAD>
<BODY>
<?
$conn = mysql_connect("localhost","isamiri","mica1982");
mysql_select_db("fin",$conn);
$sql = "SELECT * FROM cliente WHERE Nombre='$Nombre' and Apell_pat='$Apell_pat' AND Apell_mat='$Apell_mat' AND
Correo='$Correo'";
$rs = mysql_query($sql,$conn);
if (mysql_num_rows($rs)!=0){
echo '<form method = "POST" action = "formulario.php">';
echo '<table align="center" width="250" cellspacing="2" cellpadding="2" border="0">';
    echo '<tr>';
    echo '<td colspan="2" align="center" ';
    echo '<td colspan="2" align="center" ';
    echo 'bgcolor=#990AA0><span style="color:ffffff"><b>Ya estas registrado</b></span>';
    echo '</td></tr>';
    echo '<tr> <td colspan="2" align="right"><input type="submit" value="Autenticar"></td></tr>';
'</table>';
'</form>';
}else{
$id_nivel=$niv;
$id_municipio=$municipio;
$id_estado=$estado;
$conn1 = mysql_connect("localhost","isamiri","mica1982");
mysql_select_db("fin",$conn1);
$sql1 = "SELECT * FROM calle WHERE Nombre_calle='$Nombre_calle'";
$result1=mysql_db_query("fin","select id_calle FROM calle WHERE Nombre_calle='$Nombre_calle'");
while ($row=mysql_fetch_array($result1))
{
    $id_calle=$row["id_calle"];
}
$rs1 = mysql_query($sql1,$conn1);
$conn2 = mysql_connect("localhost","isamiri","mica1982");
mysql_select_db("fin",$conn2);
$sql2 = "SELECT * FROM colonia WHERE Nombre_colonia='$Nombre_colonia'";
$result2=mysql_db_query("fin","select id_colonia FROM colonia WHERE Nombre_colonia='$Nombre_colonia'");
while ($row=mysql_fetch_array($result2))
{
    $id_colonia=$row["id_colonia"];
}
$rs2 = mysql_query($sql2,$conn2);
$connectid = mysql_connect("localhost","isamiri","mica1982");
$calle = mysql_connect("localhost","isamiri","mica1982");
```



```
$colonia = mysql_connect("localhost","isamiri","mica1982");
mysql_select_db("fin",$connectid);
if (mysql_num_rows($rs1)!=0){
    if (mysql_num_rows($rs2)!=0){
        mysql_db_query("fin","insert into cliente (Nombre, Apell_pat, Apell_mat, Fecha_nac, RFC, Correo, Pasword,
id_nivel_estudio, id_estado_civil) values ('$Nombrec', '$Apell_pat', '$Apell_mat', '$Fecha_nac', '$RFC', '$Correo',
'$Pasword', '$id_nivel', '$id_estado_civil')");
        $ultimo_id = mysql_insert_id($connectid);
        mysql_db_query("fin","insert into direccion (Telefono,Numero,id_cliente,id_calle,id_colonia,id_municipio, id_estado) values
('$Telefono','$Numero','$ultimo_id','$id_calle','$id_colonia','$id_municipio','$id_estado')");
        header("Location: mail1.php?id=$ultimo_id");
    }else{
        mysql_select_db("fin",$colonia);
        mysql_db_query("fin","insert into cliente (Nombre, Apell_pat, Apell_mat, Fecha_nac, RFC, Correo, Pasword,
id_nivel_estudio, id_estado_civil) values ('$Nombrec', '$Apell_pat', '$Apell_mat', '$Fecha_nac', '$RFC', '$Correo',
'$Pasword', '$id_nivel', '$id_estado_civil')");
        $ultimo_id = mysql_insert_id($connectid);
        mysql_db_query("fin","insert into colonia (Nombre_colonia, CP) values ('$Nombre_colonia','$CP')");
        $ultimo_id2 = mysql_insert_id($colonia);
        mysql_db_query("fin","insert into calle (Nombre_calle) values ('$Nombre_calle')");
        mysql_db_query("fin","insert into direccion (Telefono,Numero,id_cliente,id_calle,id_colonia,id_municipio, id_estado) values
('$Telefono','$Numero','$ultimo_id', '$id_calle', '$ultimo_id2', '$id_municipio', '$id_estado')");
        header("Location: mail1.php?id=$ultimo_id");
    }
}
}else{
    if (mysql_num_rows($rs2)!=0){
        mysql_select_db("fin",$connectid);
        mysql_select_db("fin",$calle);
        mysql_select_db("fin",$colonia);
        mysql_db_query("fin","insert into cliente (Nombre, Apell_pat, Apell_mat, Fecha_nac, RFC, Correo, Pasword, id_nivel_estudio,
id_estado_civil) values ('$Nombrec', '$Apell_pat', '$Apell_mat', '$Fecha_nac', '$RFC', '$Correo', '$Pasword', '$id_nivel',
'$id_estado_civil')");
        //Se inserta en la base de datos y se recibe el último id
        $ultimo_id = mysql_insert_id($connectid);
        mysql_db_query("fin","insert into calle (Nombre_calle) values ('$Nombre_calle')");
        $ultimo_id1 = mysql_insert_id($calle);
        mysql_db_query("fin","insert into direccion (Telefono,Numero,id_cliente,id_calle,id_colonia,id_municipio, id_estado) values
('$Telefono','$Numero','$ultimo_id','$ultimo_id1','$id_colonia','$id_municipio','$id_estado')");
        header("Location: mail1.php?id=$ultimo_id");
    }else{
        mysql_db_query("fin","insert into cliente (Nombre, Apell_pat, Apell_mat, Fecha_nac, RFC, Correo, Pasword,
id_nivel_estudio, id_estado_civil) values ('$Nombrec', '$Apell_pat', '$Apell_mat', '$Fecha_nac', '$RFC', '$Correo',
'$Pasword', '$id_nivel', '$id_estado_civil')");
```



```
//Se inserta en la base de datos y se recibe el último id
$ultimo_id = mysql_insert_id($connectid);
mysql_db_query("fin","insert into colonia (Nombre_colonia, CP) values ('$Nombre_colonia','$CP')");
$ultimo_id2 = mysql_insert_id($colonia);
mysql_db_query("fin","insert into calle (Nombre_calle) values ('$Nombre_calle')");
$ultimo_id1 = mysql_insert_id($calle);
mysql_db_query("fin","insert into direccion (Telefono,Numero,id_cliente,id_calle,id_colonia,id_municipio, id_estado) values
('$Telefono','$Numero','$ultimo_id','$ultimo_id1','$ultimo_id2','$id_municipio','$id_estado')");
header("Location: mail1.php?id=$ultimo_id");
}
}
}
mysql_free_result($rs);
mysql_close($conn);
?>
</BODY>
</HTML>
```

Nota: Los tres códigos anteriores son semejantes para dar de alta a los Productos, Proveedores y Empleados.

Código para Mostrar la Lista Desplegable para Seleccionar el Nombre del Cliente que se Desea Eliminar

```
<HTML>
<HEAD>
<TITLE>borrar1.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></HEAD>
<BODY bgcolor="#fce3fd">
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Borrar Registro</font></p>
<table align="center" border="0">
<FORM METHOD="POST" ACTION="borradonuevo.php">
<?
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
echo'<td align="left" width="80"><font size=2>Nombre:</font>' ;
//Sentencia SQL
$$SQL="Select id_cliente,Nombre, Apell_pat, Apell_mat From cliente ORDER BY Nombre";
$result=mysql_db_query("fin",$$SQL);
echo '<td align="left" width="150"><select name="Nombre">';
```

**//Muestra la lista desplegable**

```
while ($row=mysql_fetch_array($result))
{
    echo '<option value='.$row["id_cliente"].">".$row["Nombre"]." ".$row["Apell_pat"]." ".$row["Apell_mat"] . " </option>";
}
mysql_free_result($result)
?>
</select>
<tr>
<td><td>
<tr><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Borrar"></td></tr>
</td></tr>
</FORM>
</div>
</BODY>
</HTML>
```

Código para Borrar los Datos del Cliente que se Desea Eliminar en la Base de Datos

```
<HTML>
<HEAD>
<TITLE>borradonuevo.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></HEAD>
<BODY bgcolor="#fce3fd">
<?
$id_cliente=$_GET[Nombre];
$id_cliente=$Nombre;
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
//Sentencia SQL para eliminar los datos del cliente de la base de datos
$$SQL="Delete From direccion Where id_cliente='$id_cliente'";
$$SQL1="Delete From cliente Where id_cliente='$id_cliente'";
mysql_db_query("fin",$sSQL);
mysql_db_query("fin",$sSQL1);
?>
<p align="center"><font color="#990AA0" size="3" face="verdana">Registro Eliminado</font></p>
<table align="center">
<FORM METHOD="POST" ACTION="lectura.php">
<tr><td align="center" width="150"><input type="Submit" value="Visualizar"></td>
</FORM>
</table>
```



</BODY></HTML>

Nota: El código anteriormente mencionado es similar al que se utiliza para eliminar a los empleados, proveedores y productos.

Código para Visualizar los Datos de los Clientes de la Base de Datos

```
<HTML>
<HEAD>
<TITLE>lectura.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><style type="text/css">
<!--
body {
    background-color: #fce3fd;
}
-->
</style></HEAD>
<BODY>
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Clientes</font></p>
<?
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
//Sentencia SQL para seleccionar los datos de los clientes almacenados en la base de datos
$result=mysql_db_query("fin","select cliente.Nombre, cliente.Apell_pat, cliente.Apell_mat, cliente.Fecha_nac, cliente.RFC,
cliente.Correo, cliente.Password, nivel_estudios.Nivel, estado_civil.Edo_civil, direccion.Telefono, direccion.Numero,
calle.Nombre_calle, colonia.Nombre_colonia, colonia.CP, municipio.Nombre_municipio, estado.Nombre_estado FROM cliente,
nivel_estudios, estado_civil, direccion, calle, colonia, municipio, estado WHERE
cliente.id_nivel_estudio=nivel_estudios.id_nivel_estudio AND cliente.id_estado_civil=estado_civil.id_estado_civil AND
cliente.id_cliente=direccion.id_cliente AND calle.id_calle=direccion.id_calle AND colonia.id_colonia=direccion.id_colonia AND
direccion.id_municipio=municipio.id_municipio AND estado.id_estado=direccion.id_estado ORDER BY Nombre");
?>
//Se crea la tabla con los datos que se desean mostrar
<table align="center" border="1">
<tr><th bgcolor=#dcd8fc><font size=2>Nombre</th></font>
<th bgcolor=#dcd8fc><font size=2>Apellido Paterno</th></font>
<th bgcolor=#dcd8fc><font size=2>Apellido Materno</th></font>
<th bgcolor=#dcd8fc><font size=2>Fecha de Nacimiento</th></font>
<th bgcolor=#dcd8fc><font size=2>RFC</th></font>
<th bgcolor=#dcd8fc><font size=2>Correo</th></font>
<th bgcolor=#dcd8fc><font size=2>Password</th></font>
```



```
<th bgcolor=#dcd8fc><font size=2>Nivel de Estudios</th></font>
<th bgcolor=#dcd8fc><font size=2>Estado Civil</th></font>
<th bgcolor=#dcd8fc><font size=2>Teléfono</th></font>
<th bgcolor=#dcd8fc><font size=2>Calle</th></font>
<th bgcolor=#dcd8fc><font size=2>Numero</th></font>
<th bgcolor=#dcd8fc><font size=2>Colonia</th></font>
<th bgcolor=#dcd8fc><font size=2>CP</th></font>
<th bgcolor=#dcd8fc ><font size=2>Municipio</th></font>
<th bgcolor=#dcd8fc><font size=2>Estado</th></font></tr>
<?>
```

//Se muestran los registros

```
while ($row=mysql_fetch_array($result))
{
echo '<tr><td bgcolor=#f9cafb><font size=2>'. $row["Nombre"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Apell_pat"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Apell_mat"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Fecha_nac"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["RFC"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Correo"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Pasword"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Nivel"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Edo_civil"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Telefono"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Nombre_calle"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Numero"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Nombre_colonia"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["CP"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Nombre_municipio"].'</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Nombre_estado"].'</td></tr></font>';
}
mysql_free_result($result)
?>
</table>
<BR>
<table align="center">
<FORM METHOD="POST" ACTION="Inser.php">
<tr><td align="center" width="150"><input type="Submit" value=" Agregar "></td>
</FORM>
<FORM METHOD="POST" ACTION="actualizarCli.php">
<td align="center" width="150"><input type="Submit" value=" Actualizar "></td>
</FORM>
<FORM METHOD="POST" ACTION="borrado.php">
<td align="center" width="150"><input type="Submit" value=" Eliminar "></td></tr>
</FORM>
```



```
</table>
</BODY>
</HTML>
```

Nota: El código anteriormente mencionado es similar al que se utiliza para visualizar a los empleados, proveedores y productos.

Código para Mostrar la Lista Desplegable para Seleccionar el Nombre del Cliente que se Desean Modificar sus Datos Personales

```
<HTML>
<HEAD>
<TITLE>ActualizarCli.php</TITLE>
</HEAD><meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><style type="text/css">
<!--
body {
background-color: #fce3fd;
}
-->
</style>
<BODY>
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Actualizar Registro</font></p>
<FORM METHOD="POST" ACTION="actCli1.php">
<table align="center" border="0">
<?
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
echo '<td align="left" width="65"><font size=2>Nombre:</font></td>' ;
//Sentencia SQL
$$SQL="Select id_cliente,Nombre, Apell_pat, Apell_mat From cliente ORDER BY Nombre";
$result=mysql_db_query("fin",$sSQL);
echo '<td align="left" width="150"><select name="Nombre">';
//Muestra la lista desplegable
while ($row=mysql_fetch_array($result))
{
".$row["Apell_mat_empleado"]. "</option>";
echo '<option value='.$row["id_cliente"].">".$row["Nombre"]. " " . $row["Apell_pat"] . " " . $row["Apell_mat"] . "</option>";
}
'</td>';
mysql_free_result($result)
?>
```



```
</select>
<tr><td><td>
<tr><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Actualizar"></td></tr>
</td></tr>
</table>
</FORM>
</div>
</BODY></HTML>
```

Código para Mostrar los Datos del Cliente de la Base de Datos que se Desean Actualizar

```
<HTML>
<HEAD>
<TITLE>actualizar.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><style type="text/css">
<!--
body {
    background-color: #fce3fd;
}
-->
</style></HEAD>
<BODY>
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Actualizar Cliente</font></p>
<FORM METHOD="POST" ACTION="actualizaC.php">
<?
$valo=1;
$valor=2;
$id_cliente=$Nombre;
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
/Sentencia SQL
$result=mysql_db_query("fin","select cliente.Nombre, cliente.Apell_pat, cliente.Apell_mat, cliente.Fecha_nac, cliente.id_cliente,
cliente.RFC, estatus.id_estatus, nivel_estudios.id_nivel_estudio, estado_civil.id_estado_civil, direccion.Telefono, direccion.Numero,
calle.Nombre_calle, calle.id_calle, colonia.Nombre_colonia, colonia.id_colonia, colonia.CP, municipio.id_municipio,
estado.id_estado, cliente.Correo, cliente.Pasword FROM cliente, estatus, nivel_estudios, estado_civil, direccion, calle, colonia,
municipio, estado WHERE cliente.id_cliente='$id_cliente' AND cliente.id_nivel_estudio=nivel_estudios.id_nivel_estudio AND
cliente.id_estado_civil=estado_civil.id_estado_civil AND cliente.id_cliente=direccion.id_cliente AND calle.id_calle=direccion.id_calle
AND direccion.id_colonia=colonia.id_colonia AND direccion.id_municipio=municipio.id_municipio AND
estado.id_estado=direccion.id_estado ORDER BY Nombre");
?>
```



<?

//Se muestran los registros

while (\$row=mysql_fetch_array(\$result))

{

```
    '$row["id_cliente"].';
    $Cla=$row["id_cliente"];
    '$row["Nombre"].';
    $Nom=$row["Nombre"];
    '$row["Apell_pat"].';
    $Pat=$row["Apell_pat"];
    '$row["Apell_mat"].';
    $Mat=$row["Apell_mat"];
    '$row["Fecha_nac"].';
    $Fec= $row["Fecha_nac"];
    '$row["RFC"].';
    $RFC=$row["RFC"];
    '$row["Correo"].';
    $Cor=$row["Correo"];
    '$row["Pasword"].';
    $Pas=$row["Pasword"];
    '$row["id_nivel_estudio"].';
    $Niv=$row["id_nivel_estudio"];
    '$row["id_estado_civil"].';
    $Civ=$row["id_estado_civil"];
    '$row["Telefono"].';
    $Tel=$row["Telefono"];
    '$row["Nombre_calle"].';
    $Cal=$row["Nombre_calle"];
    $id_calle=$row["id_calle"];
    '$row["Numero"].';
    $Num=$row["Numero"];
    '$row["Nombre_colonia"].';
    $Col=$row["Nombre_colonia"];
    '$row["id_colonia"].';
    $id_colonia=$row["id_colonia"];
    '$row["CP"].';
    $CP=$row["CP"];
    '$row["id_municipio"].';
    $Mun=$row["id_municipio"];
    '$row["id_estado"].';
    $Est=$row["id_estado"];

```

}

mysql_free_result(\$result)

?>



```
<table align="center" border="0">
<td align="left" width="150"><font size=2>id_Cliente:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Clave" VALUE="<?echo "".$Cla;?>"></td>
<tr><td align="left" width="150"><font size=2>Nombre del Empleado:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre" VALUE="<?echo "".$Nom;?>" size="20" maxlength="100"
"></td></tr><td align="left" width="150"></td>
<tr><td align="left" width="150"><font size=2>Apellido Paterno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apellido_paterno" VALUE="<?echo "".$Pat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Apellido Materno:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Apellido_materno" VALUE="<?echo "".$Mat;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Fecha de Nacimiento:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Fecha_nac" VALUE="<?echo "".$Fec;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>RFC:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="RFC" VALUE="<?echo "".$RFC;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Correo:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Correo" VALUE="<?echo "".$Cor;?>"></td></tr>
<tr><td align="left" width="150"><font size=2>Password:</font></td>
<td align="left" width="150"><INPUT TYPE="password" NAME="Pasword" VALUE="<?echo "".$Pas;?>"></td></tr>
<?>
```

//Conexion con la base de datos

```
mysql_connect("localhost","isamiri","mica1982");
```

//Sentencia SQL

```
echo'<tr><td align="left" width="150"><font size=2>Nivel de Estudios:</font></td>' ;
```

```
$$SQL="Select id_nivel_estudio, Nivel From nivel_estudios";
```

```
$result=mysql_db_query("fin",$sSQL);
```

```
echo '<td align="left" width="150"><select name="niv">';
```

//Se genera el menu desplegable

```
while ($row=mysql_fetch_array($result))
```

```
{
```

```
if ($Niv==$row["id_nivel_estudio"])
```

```
{
```

```
echo '<option value='.$row["id_nivel_estudio"].' SELECTED>'.$row["Nivel"] . '</option>';
```

```
}else{
```

```
echo '<option value='.$row["id_nivel_estudio"].'>'.$row["Nivel"] . '</option>';
```

```
}
```

```
}
```

```
'</td></tr>';
```

```
echo '<td><font size=2>Estado civil:</font></td>';
```

```
if($Civ==$valo)
```

```
{
```

```
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" CHECKED>Soltero
```

```
(a)</font></td><tr></tr>';
```

```
echo '<td></td>';
```

```
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2">Casado (a)</font></td>';
```



```
}
if($Civ==$valor){
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="1" >Soltero
(a)</font></td><tr></tr>';
echo '<td></td>';
echo '<td align="left" width="150"><font size=2><input type="radio" name="id_estado_civil" value="2" CHECKED>Casado
(a)</font></td>';
}
'</td></tr>';
?>
<tr><td align="left" width="150"><font size=2>Teléfono:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Telefono" VALUE="<?echo "".$Tel;?> "></td></tr>
<tr><td align="left" width="180"><strong><font size=2>Dirección:</font></strong></td></tr>
<tr><td align="left" width="150"><font size=2>Calle:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_calle" VALUE="<?echo "".$Cal;?><?"".$id_calle;?>
"></td></tr>
<tr><td align="left" width="150"><font size=2>Número:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Numero" VALUE="<?echo "".$Num;?> "></td></tr>
<tr><td align="left" width="150"><font size=2>Colonia:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="Nombre_colonia" VALUE="<?echo "".$Col;?> "></td></tr>
<tr><td align="left" width="150"><font size=2>Código Postal:</font></td>
<td align="left" width="150"><INPUT TYPE="TEXT" NAME="CP" VALUE="<?echo "".$CP;?> "></td></tr>
<?
//Conexion con la base de datos
mysql_connect("localhost","isamiri","mica1982");
//Sentencia SQL
echo'<tr><td align="left" width="150"><font size=2>Municipio:</font></td>' ;
$sSQL="Select id_municipio, Nombre_municipio From municipio";
$result=mysql_db_query("fin",$sSQL);
echo '<td align="left" width="150"><select name="municipio">';
//Se genera el menu desplegable
while ($row=mysql_fetch_array($result))
{
    if ($Mun==$row["id_municipio"])
    {
        echo '<option value='.$row["id_municipio"].' SELECTED>'.$row["Nombre_municipio"] . "</option>";
    }else{
        echo '<option value='.$row["id_municipio"].'>'.$row["Nombre_municipio"] . "</option>";
    }
}
}
'</td></tr>';
?>
<?

```

**//Conexion con la base de datos**

```
mysql_connect("localhost","isamiri","mica1982");
```

//Sentencia SQL

```
echo '<tr><td align="left" width="150"><font size=2>Estado:</font></td>' ;
$$SQL="Select id_estado, Nombre_estado From estado";
$result=mysql_db_query("fin", $$SQL);
echo '<td align="left" width="150"><select name="estado">';
//Generamos el menu desplegable
while ($row=mysql_fetch_array($result))
{
    if ($Est==$row["id_estado"])
    {
        echo '<option value='.$row["id_estado"].' SELECTED>'.$row["Nombre_estado"] . "</option>";
    }else{
        echo '<option value='.$row["id_estado"].'>'.$row["Nombre_estado"] . "</option>";
    }
}
'</td></tr>';
?>
<input name="id_calle" type="hidden" value="<?echo "".$id_calle;?>">
<input name="id_colonia" type="hidden" value="<?echo "".$id_colonia;?>">
<br>
<td><td>
<tr><td align="center" width="150"><INPUT TYPE="SUBMIT" value="Actualizar"></td></td></tr>
</td></td>
</table>
</FORM>
</BODY>
</HTML>
```

Código para Actualizar los Datos del Cliente en la Base de Datos

```
<HTML>
<HEAD>
<TITLE>actualizar2.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><style type="text/css">
<!--
body {
background-color: #fce3fd;
}
-->
</style></HEAD>
<BODY>
```



<?

\$Clave;

\$Nombre;

\$Telefono;

\$id_estado_civil;

\$Apellido_paterno;

\$Apellido_materno ;

\$RFC;

\$Fecha_nac;

\$Password;

\$Correo;

\$Nombre_colonia;

\$id_calle;

\$id_colonia;

\$id_municipio;

\$municipio;

\$mun=\$municipio;

\$conn1 = mysql_connect("localhost","isamiri","mica1982");

//Se selecciona la base de datos y la tabla en donde se sobrescribirán los datos a modificar

mysql_select_db("fin",\$conn1);

\$ssql1 = "SELECT * FROM calle WHERE Nombre_calle='\$Nombre_calle'";

\$result1=mysql_db_query("fin","select id_calle FROM calle WHERE Nombre_calle='\$Nombre_calle'");

while (\$row=mysql_fetch_array(\$result1))

{ \$id_calle=\$row["id_calle"];}

\$rs1 = mysql_query(\$ssql1,\$conn1);

\$conn2 = mysql_connect("localhost","isamiri","mica1982");

mysql_select_db("fin",\$conn2);

\$ssql2 = "SELECT * FROM colonia WHERE Nombre_colonia='\$Nombre_colonia'";

\$result2=mysql_db_query("fin","select id_colonia FROM colonia WHERE Nombre_colonia='\$Nombre_colonia'");

while (\$row=mysql_fetch_array(\$result2))

{ \$id_colonia=\$row["id_colonia"];}

\$rs2 = mysql_query(\$ssql2,\$conn2);

\$connectid = mysql_connect("localhost","isamiri","mica1982");

\$calle = mysql_connect("localhost","isamiri","mica1982");

\$colonia = mysql_connect("localhost","isamiri","mica1982");

mysql_select_db("fin",\$connectid);

\$id_calle;

\$id_colonia;

if (mysql_num_rows(\$rs1)!=0){

if (mysql_num_rows(\$rs2)!=0){

\$sSQL2="Update cliente Set Nombre='\$Nombre', Apell_pat='\$Apellido_paterno', Apell_mat='\$Apellido_materno',

Fecha_nac='\$Fecha_nac', RFC='\$RFC', Correo='\$Correo', Password='\$Password', id_nivel_estudio='\$niv',

id_estado_civil='\$id_estado_civil' Where id_cliente='\$Clave'";



```
$ultimo_id2 = mysql_insert_id($colonia);
echo $ultimo_id2;
mysql_db_query("fin","insert into calle (Nombre_calle) values ('$Nombre_calle')");
$ultimo_id1 = mysql_insert_id($calle);
echo $ultimo_id1;
$$SQL3="Update direccion Set Telefono='$Telefono', Numero='$Numero', id_calle='$ultimo_id1', id_colonia='$ultimo_id2',
id_municipio='$municipio',id_estado='$estado' Where id_cliente='$Clave'";
echo '<p align="center"><font color="#990AA0" size="5" face="verdana">Registro Actualizado</font></p>';
echo '<table align="center">';
echo '<FORM METHOD="POST" ACTION="lectura.php">';
echo '<tr><td align="center" width="150"><input type="Submit" value=" Visualizar "></td>';
echo '</FORM>';
}
}
mysql_free_result($rs1);
mysql_free_result($rs2);
mysql_db_query("fin",$sSQL1);
mysql_db_query("fin",$sSQL2);
mysql_db_query("fin",$sSQL3);
?>
</BODY>
</HTML>
```

Nota: El código anteriormente mencionado es similar al que se utiliza para modificar los datos personales de los empleados, proveedores y productos.

Código para Generar el Costo Promedio de un Determinado Producto

```
<HTML>
<HEAD>
<TITLE>lectura.php</TITLE>
</HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><style type="text/css">
<!--
body {
    background-color: #fce3fd;
}
-->
</style></HEAD>
<BODY>
<font color="Navy" face="verdana" size=6>
<p align="center"><font color="#990AA0" size="5" face="verdana">Inventario</font></p>
```



<?

//Conexion con la base de datos

mysql_connect("localhost","isamiri","mica1982");

?>

<table align="center" border="1">

<tr>

<td bgcolor=#dcd8fc colspan=50 ><div align="center">Control de Existencias</div></td>

</tr>

<?

//Sentencia SQL

```
$result=mysql_db_query("fin","select producto.Nombre_producto, clasificacion.Nom_clasificacion, inventario.Fecha_inv,
inventario.Entradap, inventario.Entradac,inventario.Salidap, inventario.Precio FROM inventario, producto, clasificacion WHERE
inventario.id_producto='$Nombre_producto' AND producto.id_producto='$Nombre_producto' AND
producto.id_clasificacion=clasificacion.id_clasificacion ORDER BY Fecha_inv, Nombre_producto");
while ($row=mysql_fetch_array($result))
```

{

\$r=\$row[Nom_clasificacion];

\$n=\$row[Nombre_producto];

}

```
$result1=mysql_db_query("fin","select producto.Nombre_producto, clasificacion.Nom_clasificacion, inventario.Fecha_inv,
inventario.Entradap, inventario.Entradac,inventario.Salidap, inventario.Precio FROM inventario, producto, clasificacion WHERE
inventario.id_producto='$Nombre_producto' AND producto.id_producto='$Nombre_producto' AND
producto.id_clasificacion=clasificacion.id_clasificacion ORDER BY Fecha_inv, Nombre_producto");
```

echo '<tr><td bgcolor=#dcd8fc colspan=3>Producto: '.\$n.'</td>';

echo '<th bgcolor=#dcd8fc colspan=3>Referencia: '.\$Nombre_producto.'</th>';

echo '<th bgcolor=#dcd8fc colspan=3>Ubicación: '.\$r.'</th>';

echo '</tr>';

echo '<tr><th bgcolor=#dcd8fc>Fecha</th>';

echo '<th bgcolor=#dcd8fc>Entradas</th>';

echo '<th bgcolor=#dcd8fc >Salidas</th>';

echo '<th bgcolor=#dcd8fc>Saldo</th>';

echo '<th bgcolor=#dcd8fc>Entradas</th>';

echo '<th bgcolor=#dcd8fc>Salidas</th>';

echo '<th bgcolor=#dcd8fc>Saldo</th>';

echo '<th bgcolor=#dcd8fc>Costo Promedio</th>';

echo '</tr>';

while (\$row=mysql_fetch_array(\$result1))

{

\$x=\$row[Entradap];

\$n=\$row[Nombre_producto];

if(\$x!=0)

{

echo '<tr><td bgcolor=#f9cafb>'.\$row["Fecha_inv"].'</td>';

echo '<td bgcolor=#f9cafb>'.\$row["Entradap"].'</td>';



```
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Salidap"]. '</td></font>';
$Saldop=$Saldop+$row[Entradap]-$row[Salidap];
echo '<td bgcolor=#f9cafb><font size=2>'. $Saldop. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Entradac"]. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Salidac"]. '</td></font>';
$Saldoc=$Saldop*$row[Entradac];
echo '<td bgcolor=#f9cafb><font size=2>'. $Saldoc. '</td></font>';
$Costop=$Saldoc/$Saldop;
echo '<td bgcolor=#f9cafb><font size=2>'. $Costop. '</td></font></tr>';
}else{
echo '<tr><td bgcolor=#f9cafb><font size=2>'. $row["Fecha_inv"]. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Entradap"]. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Salidap"]. '</td></font>';
$Saldop=$Saldop+$row[Entradap]-$row[Salidap];
echo '<td bgcolor=#f9cafb><font size=2>'. $Saldop. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $row["Entradac"]. '</td></font>';
echo '<td bgcolor=#f9cafb><font size=2>'. $Costop. '</td></font>';
$Saldoc1=$Saldop+$row[Entradap]*$row[Entradac]-$row[Salidap]*$Costop;
echo '<td bgcolor=#f9cafb><font size=2>'. $Saldoc1. '</td></font>';
$Costop=$Saldoc/$Saldop;
echo '<td bgcolor=#f9cafb><font size=2>'. $Costop. '</td></font></tr>';
}
}
mysql_free_result($result1)
?>
</table>
</BODY></HTML>
```

Nota: El código anteriormente mencionado es similar al que se utiliza para realizar el inventario de entradas y salidas de los productos.



Tienda Virtual de Abarrotes - Hidalgo TAV-Hidalgo

Área de conocimiento: Sistemas de Información

María Isabel Carbajal Amador¹, Miriam Vargas Cruz², Arturo Curiel Anaya³, Luis Heriberto García Islas⁴

¹ Universidad Autónoma del Estado de Hidalgo, Primera Cerrada 27 de Septiembre 112, C.F.E., Pachuca Hgo., 42090. México isabel_carbajal2000@yahoo.com.mx, (771)7135668

² Universidad Autónoma del Estado de Hidalgo, Av. 3 Mza. I Lt. 12 207, Plutarco E. Calles., Pachuca Hgo., 42035. México mvc_212003@yahoo.com.mx, (771)7187485

³ Universidad Autónoma del Estado de Hidalgo, Mártires de Chicago Norte 408, Francisco I. Madero, Tulancingo de Bravo, Hidalgo, 43650. México curiel@uaeh.edu.mx, 01771 71 72000 ext. 6734

⁴ Universidad Autónoma del Estado de Hidalgo, Cerrada de Aquiles Serdan 102, Centro, Pachuca de Soto, Hidalgo, 42000. México luishg@uaeh.reduaeh.mx, 01771 71 72000 ext. 6734

Resumen. Este artículo presenta el diseño e implementación de una tienda de abarrotes virtual en 3D, que presenta dos interfaces una dirigida al Usuario Navegante (cliente) y la otra a la Administración de la misma, utilizando la metodología OOWS que es una aproximación para el modelado orientado a objetos de soluciones web que proporciona mecanismos para la especificación de sistemas de información dinámicos hipermediales y de comercio electrónico.

Palabras Clave: Tienda Virtual, Mundo Virtual, Carrito de Compra.

1. Introducción

En los últimos años el Internet se ha convertido en un medio comercial masivo, con grandes cantidades de usuarios en todo el mundo.

El comercio electrónico, en la actualidad, se ha difundido a través de *tiendas virtuales*, que tienen la capacidad de ser tiendas físicas pero sin estructura local, personal de ventas, cuentas de luz, agua, etc. Sin embargo no se cuenta con la representación tridimensional de ellas; por lo tanto el presente trabajo implementa una nueva visión al respecto, integrando el comercio electrónico con una tienda virtual en tercera dimensión en donde el cliente podrá simular un recorrido a través de ella.

Una tienda virtual es un sistema que permite a los clientes de un comercio realizar pedidos en línea desde cualquier lugar del planeta a través de Internet. Las tiendas virtuales cuentan con ciertos elementos comunes, como el catálogo de productos, el carrito de compras y el registro de clientes. Las tiendas virtuales, además, permiten que el cliente realice el pago por su compra mediante una tarjeta de crédito y el comercio reciba el dinero directamente en su cuenta bancaria a través de sistemas de pago electrónico [2], [5].

Este artículo está organizado en 6 secciones. La sección 2 describe el estado del arte, describiendo los tipos de software que existen en la actualidad para desarrollar tiendas virtuales. En la sección 3 se



explica la metodología OOWS (Método de Producción de Software Orientado a Objetos), en la sección 4 se aplica la metodología OOWS al caso de estudio TAV-Hidalgo, en la sección 5 se presentan las conclusiones y los trabajos futuros y por último en la sección 6 las referencias.

2. Estado del Arte

En los últimos años han surgido gran cantidad de aproximaciones de software que intentan ayudar en la construcción de tiendas virtuales en ambientes web, proporcionando herramientas que faciliten el desarrollo de estas.

Algunos ejemplos destacables de estos software's son: Cube Stores [3] AB-Shop [1], e-Shop [4], Neostore [6] y SACE [11]. Estos software's permiten la creación de un tienda virtual en muy poco tiempo debido a que no se necesita tener conocimientos de programación ni de diseño web; las tiendas virtuales creadas con estos software's incluyen todos los elementos básicos de un sistema de comercio electrónico: el catálogo de productos, carrito de compras y registro de usuarios, además de diferentes mecanismos de seguridad.

Las principales herramientas que se están utilizando para el desarrollo de las tiendas virtuales son *asp*, *jsp*, *php*, *mysql*, *sql server*, *html*, *front page* y *flash*; todas enfocadas a la búsqueda de la eficiencia y competitividad.

3. Metodología OOWS (Método de Producción de Software Orientado a Objetos)

Este método es una extensión del OO-Method con capacidades navegacionales y de presentación [10]. El esquema conceptual de OOWS se compone de cinco modelos: el diagrama de clases, que define la estructura estática, los modelos dinámicos y funcional, que definen el comportamiento de sistema, el modelo de navegación, que definen los aspectos navegacionales y el modelo de presentación, que especifica las características de presentación de información [12], [7], [8], [9].

Siguiendo la aproximación OO-Method, la especificación de los requisitos de usuario debe realizarse en la fase de modelado conceptual. Para modelar la navegación asociada al sistema deseado se propone un proceso de desarrollo de soluciones web con dos pasos principales: Especificación del Problema y Desarrollo de la Solución.

TAV-Hidalgo es una tienda virtual de abarrotes en 3D que pone a disposición de los usuarios de Internet la posibilidad de hacer compras en línea de los productos que utiliza en la vida diaria, con la comodidad de no salir de su hogar o lugar de trabajo, sin pérdida de tiempo y con la oportunidad de emplearlo en otras actividades.

TAV-Hidalgo se compone de dos interfaces una dirigida a los usuarios navegantes (clientes) y la otra a la administración de la tienda. La segunda interfase permite tener un control absoluto de la tienda, debido a que permite administrar a clientes, empleados, proveedores, productos e inventario.

4. Caso de estudio: TAV-Hidalgo

Los requisitos de la aplicación TAV-Hidalgo se presentan a continuación:

“La aplicación esta orientada a la venta en línea de abarrotes, organizados en una clasificación. Existen dos tipos de usuarios de la aplicación: los Administradores y los Usuarios Navegantes (clientes). Los primeros son los encargados de la administración de los productos que se ponen en venta, así como de los usuarios, empleados, proveedores e inventario. Los segundos son los usuarios comunes (compradores) de esta aplicación. La funcionalidad de la aplicación es:



- Usuarios Navegantes (clientes). Las compras que se realicen se deberán ir incluyendo, simbólicamente, en un carrito de compra; el usuario podrá consultar en cualquier momento el contenido de su carrito y realizar modificaciones sobre su contenido. Este carrito de compra se creará en el momento en el que se reciba la petición de entrada en el sistema y pertenecerá al usuario que está navegando en ese momento; todas las operaciones que el usuario realice sobre el sistema se harán de forma anónima, de modo que el usuario no deberá identificarse (registrarse) hasta que no vaya a confirmar su compra; para comprar un producto se deberá llegar a él a través de un recorrido por el mundo virtual y seleccionándolo; cuando se seleccione un producto en el mundo virtual, se mostrarán todos los datos de ese producto y se podrá comprar. Esto hará que el producto sea incluido en el carrito de compra de ese usuario y que se muestre su contenido actual; mientras veamos el contenido del carrito, se podrá cambiar la cantidad de productos que se desea adquirir de cada producto de los comprados hasta el momento o eliminar alguna de las compras del carrito; cuando se decida confirmar la compra se realizarán dos acciones: La primera consistirá en crear una nota de compra (para lo que el comprador debe haberse identificado) y la segunda será reducir las existencias de los productos comprados; cuando se haya confirmado una compra, ya no se podrá modificar el contenido del carrito.
- Administradores. Gestionar y mantener la aplicación, así como el mundo virtual, los clientes, productos, empleados, proveedores e inventario.

El caso de estudio empieza con la obtención de los requisitos del sistema en la fase de **Especificación del Problema**. El sistema se divide en la funcionalidad relevante y estos requisitos son descritos mediante un diagrama de casos de uso. En las Figuras 1 y 2 se muestran los diagramas de casos de uso para el agente Usuario Navegante y para el agente Administrador.

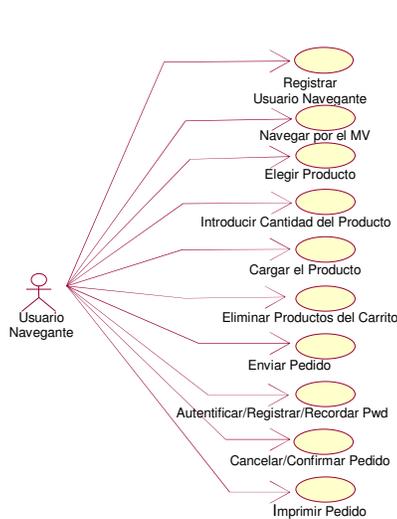


Figura 1. Caso de uso del Agente Usuario Navegante

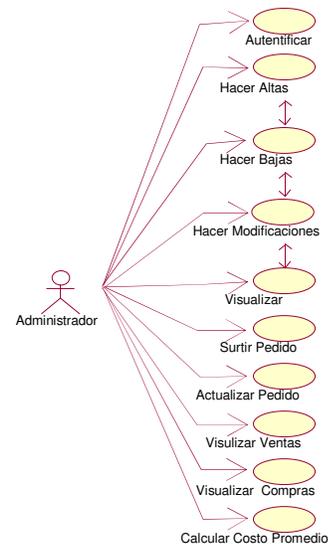


Figura 2. Caso de uso del agente Administrador

En la fase de modelado conceptual se construyen los siguientes modelos: Modelo de Objetos, Modelo Dinámico, Modelo Funcional, Modelo Navegacional y Modelo de Presentación. Estos dos últimos recogen las extensiones incorporadas a OO-Method.



La Figura 3 muestra el **Modelo de Objetos** del caso de estudio. El carrito de compra (paquete) participa directamente en el proceso de compra en línea del Usuario Navegante. Para su construcción se han estudiado los requisitos funcionales del sistema desde un punto de vista OO, y se ha asociado esta funcionalidad a cada usuario, según los casos de uso planteados, además de que se ha incorporado la parte administrativa del sistema.

Después del Modelo de Objetos, se construye el **Modelo Dinámico** donde se describen las vidas válidas de los objetos representando el comportamiento de cada clase del sistema según la interpretación descrita por OO-Method. La Figura 4 presenta una porción del Modelo Dinámico de la clase Pedido.

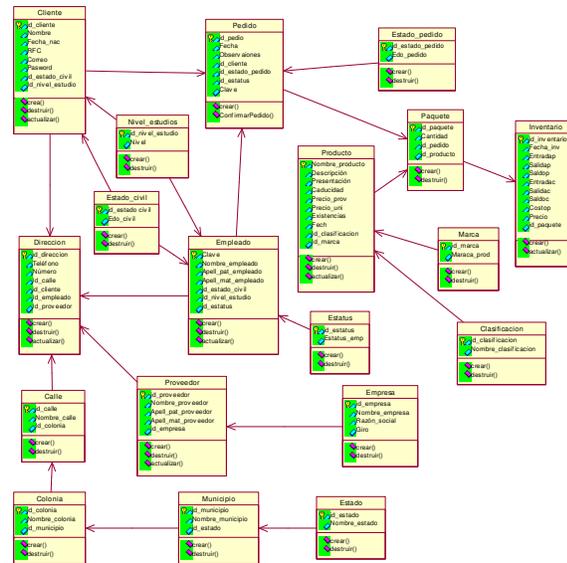


Figura 3. Modelo de Objetos.

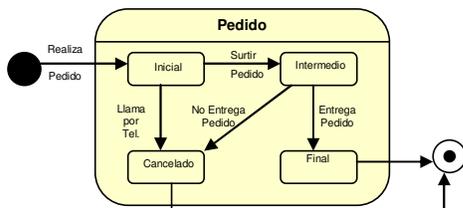


Figura 4. Modelo dinámico del Carrito de Compra.

El **Modelo Funcional** captura la semántica asociada a los cambios de estado de los objetos. El valor de cada atributo es modificado dependiendo de la acción que activó el cambio de estado, de los argumentos de dicho evento y del estado actual del objeto.

La Figura 6 muestra un ejemplo para la clase Paquete, donde se halla definida la siguiente evaluación:

Atributo: Cantidad
Categoría: De estado Evento: ModCantidad(IngresarCantidad)
Efecto: IngresarCantidad Condición: IngresarCantidad > 0

Figura 6. Parte del modelo Funcional de la clase Paquete.

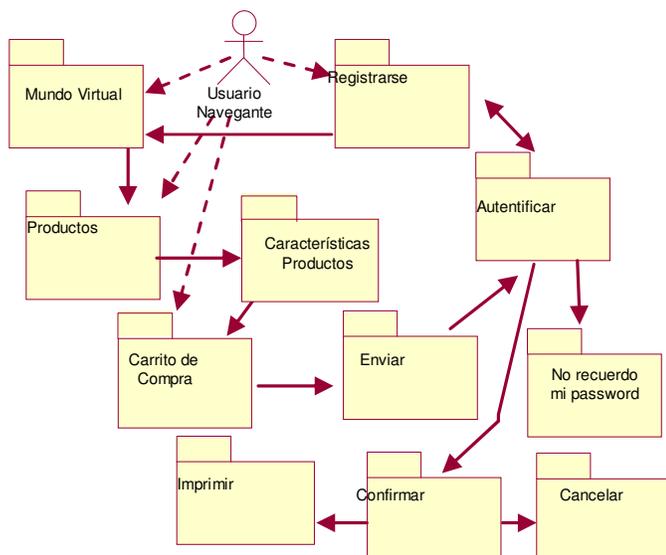


Figura 7. Mapa de Navegación del Usuario Navegante

A continuación se construye el **Modelo de Navegación** donde se estructura el acceso de cada usuario al sistema. La Figura 7 presenta el mapa de navegación del agente Usuario Navegante con sus contextos de navegación que han sido identificados en las primeras fases de especificación del sistema. También aparecen sobre el mapa los servicios que son ejecutados al iniciar y finalizar una sesión. Cuando el servidor Web recibe una petición de un cliente, ejecuta el servicio crear del Usuario Navegante asociándole además un Carrito de compra. Cuando el Usuario Navegante abandona el sistema se ejecuta el servicio destruir, eliminando además, si no ha sido confirmada, su Carrito de compra asociado.



En este mapa de navegación se aprecia que el Usuario Navegante siempre tendrá disponibles los contextos (marcados como contextos de exploración) Mundo virtual, Carrito de compra y Registrarse. A partir de estos, y siguiendo diferentes caminos navegacionales, podrá alcanzar los demás (Producto, Nota, Registro, Autenticación, Confirmar pedido).

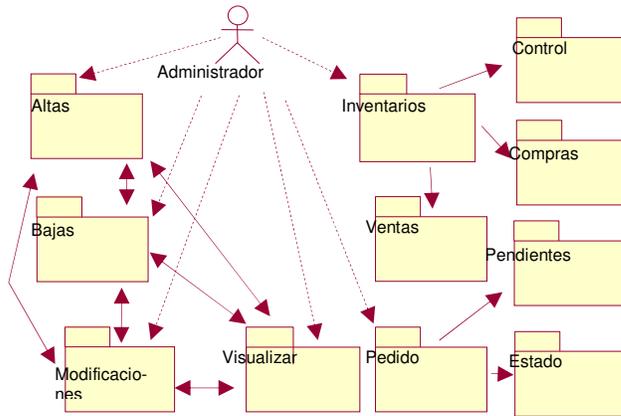


Figura 8. Mapa de Navegación del Administrador

La Figura 8 presenta el mapa de navegación del agente Administrador con sus contextos de navegación que han sido identificados en las primeras fases de especificación del sistema. En este mapa de navegación se aprecia que el Administrador siempre tendrá disponibles los contextos Altas, Bajas, Modificaciones, Visualizar, Inventarios y Pedido. A partir de estos, y siguiendo diferentes caminos navegacionales, podrá alcanzar los demás (Clientes, Empleados, Producto, Proveedores, Control de inventarios, Compras, Ventas, Pendientes y Estado).

En la Figura 9 se describe con detalle el contexto Producto, donde se recupera la información sobre un producto seleccionándolo en el mundo virtual se puede navegar al contexto Producto, donde se proporcionará información adicional del producto y se podrá comprar.

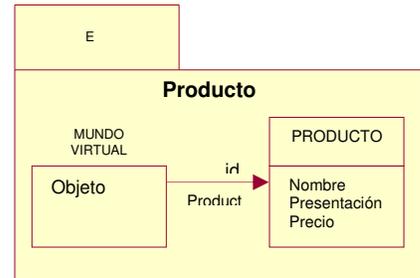


Figura 9. Contexto Productos para el Usuario Navegante.

En la Figura 10 se describe con detalle el contexto Altas, donde se proporciona la información sobre un empleado, producto, cliente o proveedor y se puede dar de alta. De igual forma se construyen los demás contextos para el agente administrador (por razones de espacio no se presentan).

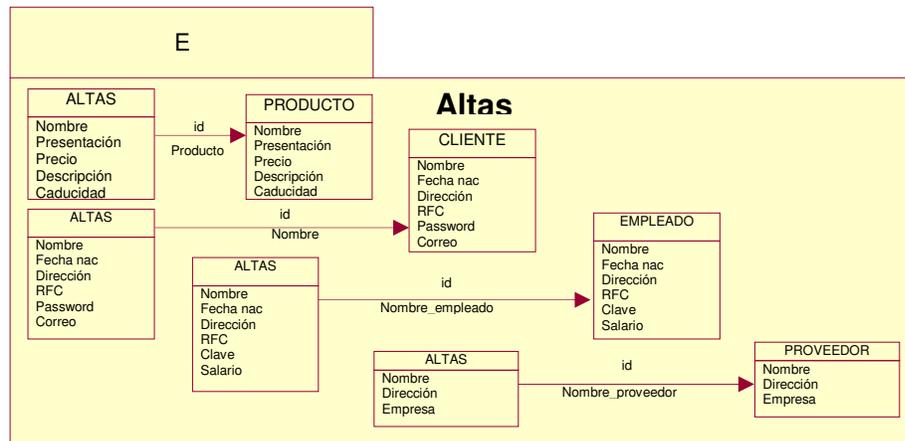


Figura 10. Contexto Altas para el Administrador.



Finalmente, se construye el **Modelo de Presentación** donde se captan los requisitos de presentación de información para cada contexto del mapa de navegación. En la Figura 11 se muestra la plantilla de presentación asociada al contexto *Productos* para el *Usuario Navegante*.

En la Figura 12 se muestra la plantilla de presentación asociada al contexto *Altas* para el *Administrador*.

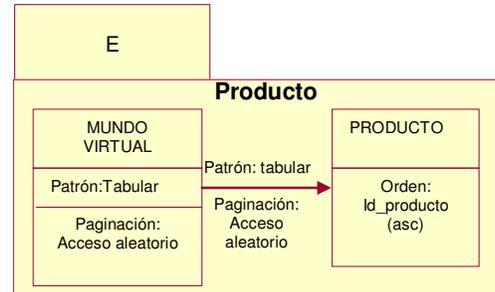


Figura 11. Modelo de presentación Contexto Producto para el Usuario Navegante.

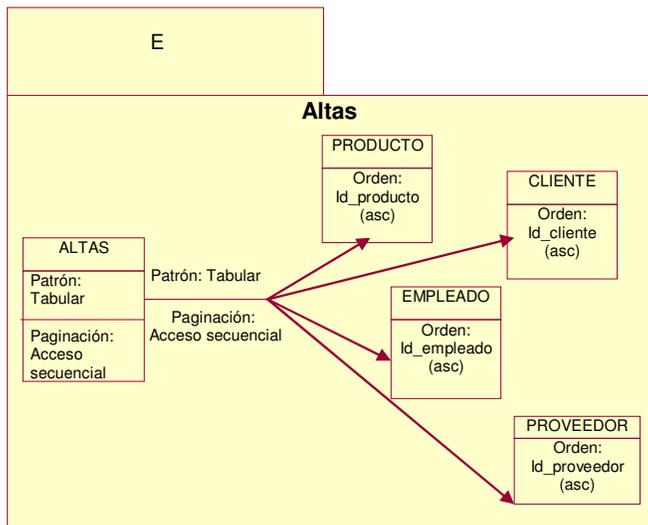


Figura 12. Modelo de presentación Contexto Altas para el Administrador

De igual forma se construyen los demás modelos de presentación para el agente administrador (por razones de espacio no se presentan). Después de construir estos modelos, el proceso llega a su segunda fase, **Desarrollo de la Solución**, donde se obtiene el prototipo software completo siguiendo la especificación realizada del sistema. En esta fase se realiza la implementación de la base de datos, así como del mundo virtual, para finalmente hacer la integración total del sistema.

En la Figura 13 aparece la interfaz principal de la aplicación la cual representa correctamente los requisitos, tanto navegacionales como de interfaz, especificados para el agente Usuario Navegante (cliente) y Administrador.



Figura 13. Interfaz principal de la aplicación TAV-Hidalgo

Como se puede apreciar en la figura 13, esta implementación recoge las características navegacionales y de presentación de información que se habían especificado en la definición del contexto.

5. Conclusiones y Trabajos Futuros

En este artículo hemos presentado el desarrollo de una tienda virtual en 3D utilizando la metodología OOWS para la construcción de la aplicación Web. El trabajo realizado ha consistido en el diseño e implementación de la Base de Datos, del Mundo Virtual, así como del Sitio Web.



Los trabajos futuros que se tienen contemplados son:

- Implementar el pago en línea.
- Incorporar un agente inteligente en el Mundo Virtual que simule al cliente.
- Realizar un sistema de facturación.
- Implementar un método de encriptación para mejorar la seguridad.

6. Referencias

- [1] "AB-Shop". URL: <http://www.ab-shop.com/>
- [2] Cisco Systems, Inc. "Comercio Electrónico". URL: http://www.cisco.com/global/ES/solutions/ent/bus_solutions/ec_home.shtml
- [3] "Cube Stores". URL: <http://www.cubestores.com/es/>
- [4] "e-Shop". URL: <http://www.eshop.es/>
- [5] Iriarte Ahon, Erick. "Comercio Electrónico en América Latina Realidades y Perspectivas".
- [6] "Neostore". URL: <http://www.neostore.com.ar/>
- [7] Pastor Oscar, (2003), "Tecnología Software Avanzada para la Ingeniería del Software de la Sociedad de la Información (TAISSI) TIC2001-3530-C02", Jornada de Seguimiento de Proyectos en Tecnologías Informáticas. Programa Nacional de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Valencia, España.
- [8] Pastor O., Fons J. J., Abrahão S. M., Albert M., Campos E. (2001, Agosto) "OOWS: Una aproximación para el Modelado Conceptual de Aplicaciones Web". VIII Jornadas Iberoamericano de Informática: Tecnologías Software para Ambientes Web. Cartagena de Indias, Colombia.
- [9] Pastor O., Abrahão S. M., Cors J. F. (2000, Octubre). "OOWS: An Object-Oriented Approach for Web-Solutions Modeling". International Conference on Information Society, Media in Information Society (MEIS'00). Ljubljana, Slovenia.
- [10] Ramos Salavert, Isidro; Fernández Caballero, Antonio; Lozano Pérez, Maria Dolores. "Tendencias Actuales en la Interacción Persona-Ordenador: Accesibilidad, Adaptabilidad y Nuevos Paradigmas". URL: <http://www.info-b.uclm.es/personal/caballer/download/papers/CursoVerano2003>
- [11] "SACE". URL: <http://www.saceonline.com/>
- [12] Valderas Aranda, Pedro J. (Julio 2004). "Especificación de Requisitos en el desarrollo de Aplicaciones Web". URL: <http://oomethod.dsic.upv.es/anonimo/..%5Cfiles%5CTechnicalReport%5Creq.pdf>



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO

Instituto de Ciencias Básicas en Ingeniería

Centro de Investigación en tecnologías de Información y Sistemas

Coordinación de La licenciatura en Sistemas Computacionales

C. Maria Isabel Carvajal Amador

Miriam Vargas Cruz

Arturo Curiel Anaya

Luis Heriberto garcia Islas.

PRESENTE.

El comité de Ponencias me ha instruido para comunicarlo que su ponencia “**Tienda Virtual de Abarrotes – Hidalgo TAV-Hidalgo**”. Si ha sido aceptada, para que sea presentada dentro del 1er. Congreso Estatal en Tecnologías de Información y Comunicaciones (TIC’s) 2006 e incluida en las memorias del evento. De acuerdo con los criterios establecidos por el comité, el puntaje obtenido por su ponencia, fue el siguiente:

Aplicación tecnológica	(5)
Calidad de escritura del documento	(5)
Originalidad del trabajo	(5)
Pertinencia y actualidad de las referencias	(5)
Pertinencia del trabajo al evento	(5)
Conclusiones y aportaciones	(5)

Escala: Mala (1) Deficiente (2) Regular (3) Aceptable (4) Buena (5) **Resultado: Aceptado**

Es necesario que:

1. Se registre a mas tardar el 25 de abril para ser incluido en las memorias y en el programa del evento.
2. A la brevedad posible, confirme su asistencia enviando un correo y el articulo en extenso a la dirección de correo torres@uaeh.edu.mx o bien a luishg@uaeh.reduaeh.mx.
3. Confirme la fecha y hora de su ponencia mediante correo electrónico el día 26 de abril a partir de las 15:00 hrs

Estamos seguros de que su compromiso profesional enriquecerá nuestro Congreso. Esperamos seguir contando con su interés y participación. Sin más por el momento, quedo a sus ordenes para cualquier aclaración en los correos electrónicos torres@uaeh.edu.mx y luishg@uaeh.reduaeh.mx.

ATENTAMENTE

Comité de Ponencias

Pachuca Hgo., a 20 de abril de 2006

M. en C. Gonzalo Alberto Torres Samperio

M. en C. Luís Heriberto García Islas