



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE ESTUDIOS SUPERIORES

CAMPUS ARAGÓN

## *INGENIERÍA EN COMPUTACIÓN*

“CREAR E IMPLEMENTAR UN NUEVO  
SOFTWARE PARA ADMINISTRAR LOS  
BIENES DE MOVISAT EN CONTEL  
IZTAPALAPA”

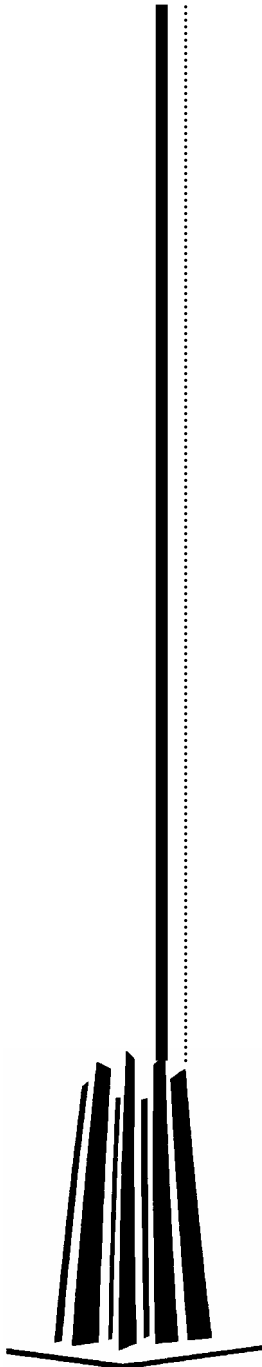
PROYECTO BAJO LA MODALIDAD DE  
“DESARROLLO DE UN CASO PRÁCTICO” QUE,  
PARA OBTENER EL TÍTULO DE

“INGENIERO EN COMPUTACIÓN”

PRESENTA:

Christian Padilla Cabrera

Dirigido por la M. en E. Imelda de la Luz Flores Díaz





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*Quiero agradecer a mis padres por el apoyo, cariño, amor que me brindaron, también por los buenos consejos que me sirvieron de guía para ser el hombre que soy ahora, gracias a todo esto, estoy aquí presentando mi tesis, un paso muy importante que marcará el futuro de mi vida.*

*Gracias Papá y Mamá.*

*Quiero agradecer a mi hermana ya que se la paso dándome lata todo el tiempo sobre echarle ganas a mis estudios y no estar perdiendo el tiempo en tonterías que solo me hacían perder tiempo muy valioso.*

*Gracias Gaby.*

*Quiero agradecer también a Rosario por apoyarme y ayudarme en todo momento y en toda situación. Por decirme mis errores y ayudarme a corregirlos, por todos esos momentos felices que pasamos, ya que me dieron fuerza para seguir siempre adelante ante cualquier adversidad.*

*Gracias por todo Rosario.*

*A mi asesora la M. en E. Imelda de la Luz Flores Díaz, por haberme ayudado y apoyado en todo lo posible para realizar este trabajo.*

*Y por último (y no por eso menos importantes) a mis amigos (los puse en orden cronológico):*

*Iván: Por mostrarme que la verdadera amistad perdura a pesar del tiempo y la distancia.*

*Arturo: Gracias a ti aprendí que no hay que encajonarse en la escuela, sino que también hay que disfrutar de las cosas divertidas de la vida, de los juegos, las fiestas, el relajo, claro sin llegar al extremo del libertinaje y poner atención, seriedad y ahínco al trabajo.*

*Emmanuel: Que me enseñó que hay que seguir tus sueños por sobre todas las cosas, por que alguien sin sueños es una persona vacía que solo hace, lo que los demás le dicen que haga, es decir, un autómata que ha perdido toda condición para ser feliz.*

*Verónica: Por mostrarme que las cosas que creemos imposibles de alcanzar, podemos alcanzarlas el día menos pensado, y que no hay que preocuparse por lo que no tienes, sino cuidar y valorar con lo que ya cuentas, como tu amistad.*

## Índice

<b>Introducción.....</b>	<b>v</b>
<b>Capítulo I Marco Teórico.....</b>	<b>1</b>
I.1 Linux Slackware.....	1
I.2 Apache.....	3
I.3 HTML.....	3
I.4 PHP.....	4
I.5 SQL Server 2000.....	5
I.6 Windows 2003 Server.....	6
<b>Capítulo II Planteamiento del problema y elección de la solución.....</b>	<b>7</b>
II.1 Planteamiento del problema.....	7
II.2 Elección de la solución.....	7
II.2.1 Determinación de requisitos.....	7
II.2.1.1 Recolección de datos.....	8
II.2.2 Solución.....	11
II.2.2.1 Base de datos.....	12
II.2.2.1.1 Análisis de las tareas y determinación de la finalidad de la B.D.....	12
II.2.2.1.2 Esquematización del flujo de tareas.....	13
II.2.2.2 El equipo de cómputo y conexión a la red interna (nodo de red en la bodega).....	14
II.2.2.3 Servidor de páginas Web.....	16
<b>Capítulo III Análisis y Diseño del Sistema.....</b>	<b>17</b>
III.1 Análisis de necesidades.....	17
III.2 Análisis costo beneficio.....	17
III.3 Diseño del sistema.....	19
III.3.1 Diagrama de flujo de datos.....	19
III.3.2 Modelo entidad relación.....	29
III.3.3 Diseño de la base de datos.....	34
III.3.4 Diagrama entidad-relación.....	38
III.3.5 Identificación de posibles valores que el usuario introducirá al sistema.....	39
III.3.6 Ruta crítica.....	45
<b>Capítulo IV Desarrollo e implementación del sistema.....</b>	<b>51</b>
IV.1 Codificación.....	51
IV.1.1 Portal.php.....	51
IV.1.2 EvaluaUsuario.php.....	52
IV.1.3 Frame.php.....	53
IV.1.3.1 Administradores.php.....	54
IV.1.3.2 Entradas0.php.....	55

IV.1.3.2.1 Entradas.php.....	57
IV.1.3.3 Autorizapedido.php.....	61
IV.1.3.4 Administrausuarios.php.....	62
IV.1.3.5 Reportes.php.....	63
IV.1.4 FrameB.php.....	68
IV.1.4.1 Bodegas.php.....	68
IV.1.4.2 Autorizasalida.php.....	69
IV.1.4.3 Autorizasalida2.php.....	71
IV.1.5 FrameU.php.....	73
IV.1.5.1 Usuarios.php.....	73
IV.1.5.2 Pedido2.php.....	74
IV.2 Pruebas.....	76
IV.3 Implantación.....	77
<b>Capítulo V Manual de Usuario y Administrador.....</b>	<b>78</b>
V.1 Manual de responsable de bodega.....	79
V.2 Manual de usuario.....	83
V.3 Manual de administrador.....	88
<b>Conclusiones.....</b>	<b>112</b>
<b>Glosario.....</b>	<b>113</b>
<b>Bibliografía.....</b>	<b>116</b>
<b>Páginas Web.....</b>	<b>116</b>

## Introducción

Hoy en día los portales en Internet son la herramienta más usada ya que se tiene acceso a la información desde cualquier parte en la que uno se encuentre. No solo sirven para consultar información, también para realizar diversas transacciones como por ejemplo la compra de artículos por Internet, presentar declaraciones ante hacienda y consultar el buró de crédito.

La información y transacciones que se realizan vía Internet deben de almacenarse en una base de datos segura que soporte los procesos vía Internet. Existen distintos manejadores de bases de datos como SQLServer (de Microsoft) y Oracle que son software licenciado (que se tiene que pagar para poder usarlo) y por otra parte esta el software libre (que no se paga por usarlo y que se refiere principalmente al ambiente de Linux) MySQL y POSTGRESQL.

Para que una página de Internet interaccione con una base de datos se necesitan dos cosas: que las páginas estén montadas en un servidor de páginas Web<sup>1</sup> y que las páginas Web sean dinámicas<sup>2</sup>.

Se puede resumir que para que un sitio Web realice transacciones y administre bien la información se necesita:

1. Elegir el lenguaje que más convenga según las necesidades que se tengan para auxiliar al HTML.
2. Elegir un servidor de páginas Web.
3. Elegir el servidor de bases de datos según sea conveniente.
4. Montar todo lo anterior en una computadora servidor.
5. Hacer las páginas.
6. Habilitar los servicios para que los usuarios tengan acceso a las páginas Web.
7. Y poner el servidor en la red.

Esta tesis tiene como objetivo construir un sistema por medio del cual, se administren los bienes de MOVISAT en CONTEL IZTAPALAPA; este sistema mejorará el sistema administrativo de bienes con el que ya se contaba, ya que optimizará los tiempos en los que se realizaba el pedido, disminuirá el número de bienes perdidos, los reportes se harán de manera más rápida, los pedidos se pueden hacer desde la oficina de quien realiza el pedido, el administrador puede autorizar o cancelar pedidos en la comodidad de su oficina, el encargado de la bodega de manera fácil y rápida, puede consultar el pedido que tiene que

---

<sup>1</sup> **Un servidor de páginas Web** es aquel que pone a disposición de los usuarios de la red sus recursos (páginas Web en éste caso). El servicio que da un servidor de páginas Web va desde que éste brinde el contenido de una página, hasta que almacene información estadística de: ¿qué página fue visitada?, ¿Qué día fue visitada?, ¿a qué hora fue visitada? y ¿quién la visito?. Existen varios servidores de páginas Web pero los más usados son IIS (Internet Information Server) de Microsoft y APACHE de Apache Foundation Software.

<sup>2</sup> **Que una página Web sea dinámica** significa que puedan realizar procesos como sumar, restar, comparar, comunicarse con una base de datos, etc.; en conclusión procesos que con HTML no serian posibles, por lo tanto, una página Web dinámica no sólo cuenta con código HTML, si no también, con otro código que auxilia al HTML. Existen varios lenguajes que auxilian al HTML como por ejemplo ASP.NET de Microsoft, JAVA y JAVASCRIPT de SUN y PHP por mencionar los más usados. Cada uno de estos lenguajes tiene su propia sintaxis y forma de funcionar pero con un mismo objetivo hacer más funcional una página WEB.

entregar, para verificar que lo que le están pidiendo en ese momento, sea lo que se autorizó y por último se puede consultar las existencias de cada bien.

El sistema se conformará por una base de datos montada en SQLSERVER y un sitio Web, que contendrá páginas Web dinámicas desarrolladas con HTML y PHP. El sitio Web se montará en Apache.

El equipo de cómputo que se usará para el sitio Web, contará con sistema operativo Linux (Slackware).

Para escoger el software se tomaron en cuenta los siguientes aspectos:

- Ya se cuenta con 2 equipos de cómputo
- Se cuenta con un servidor que tiene instalado SQLServer 2000
- Se cuenta con muy poco presupuesto
- Cuentan con una intranet

Por lo tanto para montar el servidor de páginas Web se escogió:

- **Un equipo de cómputo** con procesador Intel Celeron a 800 Mhz. Y 256 de RAM y disco duro de 40GB (ya se contaba con éste equipo).
- **Linux distribución Slackware** como sistema operativo debido a que no se tiene que pagar licencias, consume pocos recursos del equipo de cómputo y es **especial para servidores**.
- **Apache** como software para el servidor de páginas Web, ya que no se paga licencia, corre en Linux, sus opciones de configuración son las mejores ya que se nos permite manipular este software a nuestro gusto y **tiene compatibilidad para trabajar con PHP**.
- **PHP** para que nuestras páginas sean dinámicas, ya que: tiene más estabilidad en Linux, no se paga licencia, **puede realizar conexiones a bases de datos montadas en SQLSERVER 2000**.

Y para la base de datos se dispondrá del servidor con el que ya se cuenta, recordando que tiene sistema operativo Windows 2003 Server y SQLSERVER 2000.

Tomando en consideración todo esto, es que se desarrolló el presente trabajo en el cual se ve una aplicación real de lo antes mencionado. Dicho trabajo se llevo acabo en TELECOMM, donde el problema que se presentó fue una mala administración de bienes debido al sistema con el que se estaban administrando éstos. Para darle solución al mismo y fundamentar dicha solución es que se han desarrollado los siguientes capítulos:

#### ✂ Capítulo I Marco Teórico

En este capítulo se da un resumen sobre todo el software que se va a utilizar, aquí se dan a detalle las características de cada programa usado para el desarrollo y funcionamiento del sistema.

#### ✂ Capítulo II Planteamiento del problema y elección de la solución

Aquí se plantea el problema que existe en el sistema administrativo de bienes, se detallan las características de dicho problema y las posibles soluciones a éste.

Al final de este capítulo se elige y describe la mejor solución.

☞ Capítulo III Análisis y diseño del sistema

En resumen, se analizan todos los requerimientos para realizar el diseño del nuevo sistema, que dará solución al problema existente. También se estiman los tiempos de las actividades a realizar y el tiempo total de desarrollo.

☞ Capítulo IV Desarrollo e implementación del sistema

Aquí se resume todo el código fuente del nuevo sistema y se hace un resumen de cómo se implantó.

☞ Capítulo V Manual de usuario y administrador

Este último capítulo contiene los manuales del nuevo sistema, para que las personas involucradas en la administración de bienes, puedan manejar el nuevo sistema administrativo de bienes.

Esta tesis se desarrolló para aplicar los conocimientos adquiridos en la universidad, dar solución al problema que existía en el sistema administrativo de bienes en MOVISAT y obtener experiencia en el desarrollo de sistemas.



# Capítulo I

## Marco Teórico

### I.1 Linux Slackware

En 1990 un estudiante Finlandés de nombre Linus Torvalds inició un proyecto personal basado en Minix (un sistema tipo UNIX), su intención era desarrollar un sistema compatible con plataforma PC el cual fuera más confiable que Minix.

Linus Torvalds comenzó creando drivers para dispositivos de hardware en lenguaje C, y fue hasta 1991 que libera la primera versión de Linux sin ser una distribución oficial, la 0.01.

En el mismo año pone a disposición del mundo a través de Internet la primera versión oficial de Linux, la 0.02, la intención de poner a Linux en Internet era que todos los usuarios en el mundo pudieran ejecutarlo, copiarlo, modificarlo y distribuirlo para que los usuarios lo mejoren, pero hacía falta algo que garantizara que Linux fuese un software libre.

“Para asegurar que el software permaneciera libre, para que todos los usuarios pudieran ‘ejecutarlo, copiarlo, modificarlo y distribuirlo’, el proyecto debía ser liberado bajo una licencia diseñada para garantizar esos derechos al tiempo que evitase restricciones posteriores de los mismos. La idea se conoce en Inglés como copyleft (por ser lo contrario a copyright) y está contenida en la licencia pública general (GNU).”<sup>1</sup> La licencia pública general (GNU) está destinada a garantizar la libertad de compartir y cambiar el software libre.

Algunas de las características de Linux son:

- ☞ Es un sistema operativo de tiempo compartido
- ☞ Multiusuario
- ☞ Multitarea
- ☞ Multiproceso
- ☞ Multiplataforma
- ☞ Modular
- ☞ Portable entre sistemas abiertos
- ☞ Programable
- ☞ Es un sistema tipo cliente–servidor

“*Multitarea*: es la capacidad que el sistema operativo tiene, de ejecutar varios programas al mismo tiempo. Linux utiliza lo que se conoce por multitarea preventiva, que permite que todos los programas que se hayan abierto sean ejecutados.[...]

*Multiusuario*: Es la posibilidad de que varias personas usen una misma máquina [...], con sus recursos y aplicaciones, sin interferir entre ellas, de esta forma un determinado documento no va ser manipulado y menos borrado, por otro usuario.

---

<sup>1</sup> <http://es.wikipedia.org/wiki/Gnu>

*Multiplataforma*: se puede ejecutar un Linux no sólo con los microprocesadores típicos del mundo PC (Intel y AMD), sino que también se piensa en [...]”<sup>2</sup> otras plataformas.

Las plataformas principales de Linux son:

I386 Para equipos tipo PC compatibles

PPC Para equipos Maquintosh

Sparc Para estaciones de trabajo

Alpha Para arquitectura de trabajo con procesadores en paralelo de tipo alpha

Existen muchas distribuciones Linux que pueden ser adquiridas por medio de Internet, dependiendo la distribución que se elija (FEDORA, DEBIAN, SUSE, SLACKWARE, etc.) se comprará en el sitio oficial o se descargará gratuitamente de los sitios de FTP.

Slackware de Patrick Volkerding es un avanzado sistema operativo Linux, diseñado con dos objetivos: facilidad para usar y estabilidad, además es un sistema enfocado a servidores. Es enfocado a servidores ya que detecta la mayoría de hardware para servidores, viene configurado para que arranque en modo consola (consume menos recursos, lo que hace que el servidor sea más eficiente), tiene preinstalado software para montar un servidor de páginas Web (APACHE), tiene preinstalado un servidor de bases de datos (MySQL) y tiene preinstalado PHP para hacer páginas Web dinámicas por mencionar algunos, así que esta distribución viene configurada para ser un servidor.

Los requerimientos de Slackware son:

Hardware	Requiere
Procesador	386
RAM	16 MB – 64M
Espacio en Disco	500MB – 3G
Floppy Drive	1.44 MB
Unidad CD-ROM	
Software	(página oficial)

La distribución de paquetes en Slackware se hace principalmente con archivos Tgz., que se instalan desde el modo consola, lo que hace que quien maneje esta distribución debe tener un mayor conocimiento sobre Linux.

---

<sup>2</sup> García Jiménez Francisco, “Guía de campo de Linux” México, RA-MA, pp. 9-10

## I.2 Apache

Apache es un software libre, que al instalarse, funciona como un servidor HTTP<sup>3</sup>, en otras palabras es un “servidor de páginas Web” que corre en plataformas Linux y Windows, por mencionar las más comunes.

Apache cuenta con las siguientes características:

- Es robusto, y soporta un gran número de transacciones
- Configurable para diferentes entornos de trabajo
- Alto nivel de seguridad
- Disponible para una gran variedad de plataformas
- Servicio de proxy
- Soporte para granjas de servidores
- Soporte para Scripting languages integrados como módulos (por ejemplo PHP)
- Incluye el código fuente del servidor
- Soporte para accesos restringidos
- Soporte para SSL
- Es gratuito

## I.3 HTML

“HTML es un lenguaje que se emplea para crear páginas Web. Un código escrito en este lenguaje es, básicamente, un texto que el navegador (Internet Explore, Netscape, Navigator, Opera o cualquier otro) mostrará en formato de página Web. Este texto puede generar color, tamaño y fuente de la letra, fondos, imágenes, hiperenlaces y entrada de datos, así como listas de selección, botones, etc., determinados y configurados mediante los identificadores, también llamados tags.”<sup>4</sup>

Los identificadores o etiquetas de HTML son un conjunto de instrucciones o comandos que después son interpretados por los navegadores WEB, desplegando como resultado de esta interpretación las páginas WEB. Las Etiquetas de HTML se interpretan en el orden en que se ponen y van encerradas entre pico paréntesis (< >).

La estructura básica de una página debe llevar el siguiente esqueleto o estructura base:

```
<html>
<head>
  <title>Espacio para título de la ventana</title>
</head>
```

---

<sup>3</sup> HTTP protocolo de transferencia de hipertexto es el lenguaje que utilizan el cliente y el servidor para comunicarse entre si.

<sup>4</sup> López Quijado José, “Domine HTML y DHTML” México, Alfa omega Ram-Ma, p.p. XI

```
<body>
    Espacio para escribir todo el contenido de la página.....
</body>
</html>
```

Las páginas se guardan con el nombre que elijamos seguido de **.html**, ya que sólo por la extensión el sistema operativo reconoce este archivo como un documento Web.

## I.4 PHP

PHP es un lenguaje de programación usado para la creación de sitios Web, se trata de un lenguaje interpretado<sup>5</sup> usado para elaborar aplicaciones para servidores o creación de contenido dinámico para páginas Web.

PHP fue diseñado en Perl por Rasmus Lerdorf en el año de 1994 para mostrar su currículum vitae y guardar ciertos datos, en un principio fue conocido como “Personal Home Page Tools”.

En 1999 se libera la versión 4.0 y en 2004 se crea el motor Zend 2.0 y se libera PHP 5.0 que introduce un modelo de orientación a objetos muy similar al de Java.

PHP funciona de la siguiente forma:

- ✂ Primero el cliente hace una petición al servidor de páginas Web, para que se le envíe la página que está solicitando.
- ✂ El servidor ejecuta el intérprete PHP, el cual procesa la página, generada en PHP.
- ✂ El intérprete PHP regresa la página ya procesada al servidor.
- ✂ Y por último, el servidor le envía la página ya procesada al cliente.

Las características más importantes de PHP son:

- “Es un lenguaje multiplataforma.
- Permite la conexión a diferentes tipos de servidores de Bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, Microsoft SQLServer, Firebird y SQLite.
- Leer y manipular datos desde diversas fuentes.
- Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial.
- Es libre por lo que se presenta como una alternativa de fácil acceso.
- Permite las técnicas de Programación Orientada a Objetos.
- Permite crear formularios para la Web”<sup>6</sup>.
- Puede interactuar con los servidores Web más populares como ISAP, IIS y Apache.

---

<sup>5</sup> Los lenguajes interpretados van siendo codificados por la computadora mientras se están ejecutando

<sup>6</sup> <http://es.wikipedia.org/wiki/Php>

## I.5 SQL SERVER 2000

Microsoft SQL Server es un sistema de gestión de **bases de datos relacionales**<sup>7</sup> basada en el **lenguaje SQL**<sup>8</sup>, que pone a disposición de muchos usuarios, grandes cantidades de información.

Entre sus características figuran:

- ☞ “Soporte de transacciones.
- ☞ Gran estabilidad
- ☞ Gran seguridad
- ☞ Escalabilidad
- ☞ Es un producto de base de datos totalmente habilitado para la Web
- ☞ Proporciona soporte para XML y la capacidad para hacer consultas en Internet y a través del firewall”<sup>9</sup>

Incluye también un potente entorno gráfico de administración, que permite el uso de comandos **DDL**<sup>10</sup> y **DML**<sup>11</sup> gráficamente.

Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle o Sybase ASE.

---

<sup>7</sup> **Bases de datos relacionales:** Representan los datos y las relaciones entre los datos mediante una colección de tablas, cada una de las cuales tienen un número de columnas con nombres únicos.

<sup>8</sup> **Lenguaje de Consulta Estructurado (Structured Query Language)** es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas.

<sup>9</sup> <http://www.microsoft.com/spain/sql/2000/productinfo/whatsnew.aspx>

<sup>10</sup> **Data Definition Language**(Lenguaje de definición de datos): Comandos SQL que permiten crear y definir nuevas bases de datos, campos e índices.

<sup>11</sup> **Data Manipulation Language**(Lenguaje de manipulación de datos): Comandos SQL que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

## I.6 Windows 2003 Server

**Windows Server 2003** es la versión de Windows para servidores lanzada por Microsoft en el año 2003, este Windows está basado en el núcleo de Windows XP.

Windows 2003 Server cuenta con las siguientes características:

- “Sistema de archivos NTFS:
  1. cuotas
  2. encriptación y compresión de archivos, carpetas y unidades completas
  3. permite montar dispositivos de almacenamiento sobre sistemas de archivos de otros dispositivos al estilo unix
- Gestión de almacenamiento, backups... incluye gestión jerárquica del almacenamiento, consiste en utilizar un algoritmo de caché para pasar los datos menos usados de discos duros a medios ópticos o similares más lentos, y volverlos a leer a disco duro cuando se necesitan.
- Windows Driver Model: Implementación básica de los dispositivos más utilizados, de esa manera los fabricantes de dispositivos sólo han de programar ciertas especificaciones de su hardware.
- ActiveDirectory Directorio de organización basado en LDAP, permite gestionar de forma centralizada la seguridad de una red corporativa a nivel local.
- Autenticación Kerberos5.
- DNS con registro de IP's dinámicamente.
- Políticas de seguridad.

Los servidores más importantes que maneja Windows 2003 son:

- Servidor de archivos
- Servidor de impresión
- Servidor de aplicaciones
- Servidor de correo
- Servidor DNS<sup>12</sup>

Podemos pasar al siguiente capítulo para plantear el problema y dar solución a éste, con la ayuda de las herramientas mencionadas en este capítulo.

---

<sup>12</sup> <http://www.microsoft.com/spain/windowsserver2003/default.msp>

## Capítulo II

### Planteamiento del problema y elección de la solución

A lo largo de este capítulo el problema de MOVISAT y con esto, podremos obtener la mejor solución a éste.

#### II.1 Planteamiento del problema

Aquí definiremos el problema.

##### *Identificación del problema*

TELECOMM (Telecomunicaciones de México), tiene una gerencia de nombre MOVISAT, que se encarga de monitorear el envío de datos y la comunicación telefónica vía satélite para hacer los cobros respectivos a los usuarios que utilizan estos servicios.

MOVISAT no cuenta con un sistema de administración de bienes adecuado, esto causa problemas como son:

- ⌘ Pérdida de artículos en la bodega
- ⌘ Pérdida de tiempo al realizar reportes
- ⌘ Pérdida de tiempo en realizar autorizaciones de pedidos
- ⌘ Los bienes se almacenan en un cuaderno, lo que trae como problema poca o nula seguridad de la información.

La suma de todos los problemas antes mencionados, traen como consecuencia, una pérdida de tiempo valioso que pudo haberse aprovechado en algo más importante y también hay una pérdida monetaria, que se puede apreciar en la pérdida de bienes.

Por lo tanto se proponen dos soluciones: un sistema informático que permita llevar un mejor control sobre los bienes o mejorar el sistema que actualmente se utiliza.

#### II.2 Elección de la solución

La solución se obtendrá a partir de datos que se obtienen después aplicar ciertos procesos.

##### II.2.1 Determinación de requisitos

Para determinar cuál es la mejor solución, se realizaron los siguientes procesos:

- ⌘ Aplicación de entrevistas al personal involucrado con el sistema.
- ⌘ Evaluación de los recursos (informáticos, humanos, etc.) con los que cuenta MOVISAT.

### **II.2.1.1 Recolección de datos.**

La información que se recolectó para la elección y creación del sistema es la siguiente:

#### ***Entrevistas.***

Las entrevistas que se hicieron fueron las siguientes:

#### **Entrevista No. 1**

1.- ¿Cuál es su nombre?

R= Soy el Lic. José Guevara Manrique.

2.- ¿Cuál es el cargo que tiene?

R= Soy el coordinador de apoyo administrativo.

3.- ¿Y que funciones desempeña?

R= Administro la nómina y llevo la administración de los bienes de MOVISAT entre otras cosas.

4.- ¿En dónde se ubica la necesidad de realizar un software que le ayude a realizar sus actividades?

R= En la administración de los bienes.

5.- ¿Cuáles bienes y porqué?

R= Los lápices gomas, cd's, etc. y es debido a que no hay un buen sistema administrativo de los bienes, nos gustaría que se hiciera un programa que nos ayude a nuestra labor, ya que no contamos con un sistema que nos diga las existencias de algún artículo, nos es difícil hacer reportes mensuales o anuales, realmente no tenemos un sistema de administración de bienes como tal.

6.- Actualmente ¿qué procesos o actividades se desempeñan en la administración de los bienes?

R= Pues mira hay un proceso de entrada de los bienes y un proceso de salida.

7.- ¿Podría describir los procesos mencionados?

El proceso de entrada no es muy difícil de CTO nos mandan los bienes junto con una hoja, los bienes se guardan en la bodega y la hoja la archivamos.

En el proceso de salida de bienes primero la coordinación que requiere los bienes nos llena un formato especial (Ver Fig.1), una vez llenado lo veo yo para ver si lo acepto o no, si lo acepto lo firmo y el interesado va a la bodega con éste formato y saca los bienes que necesita.

8.- ¿Qué personas intervienen en este proceso administrativo de los bienes?


R= Yo, el Lic. Miguel Ángel Galindo que es el auxiliar administrativo y la gente que se encuentran en la bodega que son 2: el Sr. Gregorio Fuentes Cruz Y EL Sr. Carlos Vega López.



9.- ¿Sabe usted usar Windows e Internet?  
 R=Si.

10.- ¿Las demás personas que intervienen en la administración de bienes saben Windows e Internet?  
 R= Pues Miguel si sabe pero no sabría decirte de Gregorio y Carlos.

11.- ¿Qué tipo de datos le interesa saber de los bienes?, por ejemplo la existencia de un artículo.  
 R= Pues me interesa saber el número de: pedido, licitación, partida general, requisición y de partida de requisición, también la clave presupuestal y el CABM (Clave administrativa de bienes muebles). También necesito saber qué se le entregó a cada coordinación mensualmente y cuantos artículos por mes se consumieron, así como saber cuantos artículos entraron por mes.


**DIRECCIÓN TÉCNICA**  
**SUBDIRECCIÓN DE COMUNICACIÓN MÓVIL Y RURAL SATELITAL**  
**GERENCIA DEL CENTRO OPERATIVO MOVISAT**  
**COORDINACIÓN DE APOYO ADMINISTRATIVO**

No. 00  
 FECHA

**NOTA DE SALIDA DE BIENES DEL ALMACÉN**

RETIRA: ING. ARMANDO TABOADA BALDERAS  
 COORDINACIÓN: DE OPERACIONES DE LAS REDES MOVISAT  
 GERENCIA: CENTRO OPERATIVO MOVISAT

DATOS DEL PEDIDO		
AÑO	NUMERO	NACIONALIDAD
2005		NACIONAL

PARTIDA	No. DE CONTROL	DESCRIPCION DE LOS BIENES	UNIDAD DE MEDIDA	CANTIDAD
	CONSUMO		2	(3) 17
			7	(10) 20

ENTREGA DE CONFORMIDAD LOS BIENES
RECIBE

C. GREGORIO FUENTES CRUZ  
 RESPONSABLE DEL ALMACÉN

ING. ARMANDO TABOADA BALDERAS  
 ENC. DE LA COORD. DE OPERACIONES DE LAS REDES MOVISAT

Vo. Bo.  
 JOSÉ GUEVARA MANRIQUE  
 COORDINADOR DE APOYO ADMINISTRATIVO

JGM\*iv

Fig.1 Hoja de salidas de bienes

## Entrevista No. 2

1.- ¿Cuál es su nombre completo?

R= Gregorio Fuentes Cruz.

2.- ¿Y que puesto desempeña aquí en TELECOMM?

R= A pues estoy aquí en la bodega.

2a.- Si pero cuando lo contrataron ¿qué puesto le dieron?

R= Pues no se, pero pues debo de ser algo así como Responsable de Almacén, pues sí eso.

3.- ¿Y que actividades realiza aquí en la bodega?

R= Pues checo que las cosas que entren estén bien y checo las salidas de los bienes.

4.- ¿Qué hace cuando tiene que recibir artículos?, descríbame detalladamente lo que hace.

R= Pues mira todo lo mandan de C.T.O. cuando van a meter algo checo que todo venga bien y que coincida lo que dice esta hoja con lo que meten, ya que chequé que todo está bien, esta hoja se la mando al Lic. Guevara.

5.- ¿Y que hace cuando alguien tiene que sacar algo de la bodega?

R= Pues vienen sacan las cosas y me firman este cuaderno (ver Fig.2). El cuaderno se lo entrego al Lic. Guevara para que haga sus cosas.

6.- ¿Realiza algún otro tipo de actividad en la bodega?

R= No

7.- ¿Y usted es el único responsable de almacén?

R= No, Carlos también esta aquí y pues también sería Responsable de Almacén, y su nombre es Carlos Vega López.

8.- ¿Y que actividad desempeña el Sr. Carlos?

R= El hace lo mismo que yo.

9.- Volviendo a usted y cambiando de tema ¿sabe manejar una computadora?

R= No

10.- ¿Le gustaría que le enseñaran a usar la computadora?

R= Sí.

11.- ¿Qué haría si le dijeran tiene que usar la computadora porque hicieron un programa de computadora que lo va auxiliar en su trabajo?

R= Pues que padre nada más que les pediría que me ensañaran computación.

12.- ¿Y su compañero que cree que opine?

R= Pues está igual que yo, de hecho el Lic. Guevara ya nos había comentado algo al respecto.

Al final de la entrevista, el Sr. Gregorio comentó que les habían dado una computadora, pero que ya se las quitaron.

Fecha	Cantidad	Concepto	Nombre	Firma

Fig.2

***Evaluación de los recursos (informáticos, humanos, etc.) con los que cuenta MOVISAT.***

MOVISAT cuenta con los siguientes recursos:

Humanos.

- ☞ Un administrador
- ☞ Un auxiliar de administración
- ☞ 2 responsables de almacén

Materiales.

- ☞ Una bodega  
Es donde se almacenan los bienes, esta no cuenta con un nodo de red para conectarse a la red interna de TELECOMM.
- ☞ Un servidor (Windows 2003, SQL Server 2000).  
Cuenta con muy buena tecnología y es bastante rápido y eficiente.
- ☞ 2 PC's (Windows 2000)  
Las 2 funcionan a la perfección, cuentan con procesador INTEL CELERON (800Mhz), tiene 256Mb en RAM y 40Gb en disco duro.
- ☞ Cable UTP
- ☞ Rosetas para red

## **II.2.2 Solución**

De las entrevistas se adquirieron los siguientes datos:

- ☞ Las personas que van a interactuar con el nuevo sistema son: el Lic. José Guevara Manrique (coordinador de apoyo administrativo), el Lic. Miguel Ángel Galindo (auxiliar del coordinador de apoyo administrativo), el Sr. Gregorio Fuentes Cruz (responsable de almacén) y el Sr. Carlos Vega López (responsable de almacén).
- ☞ El señor Gregorio y el señor Carlos no saben usar la computadora.
- ☞ Los usuarios necesitan que el nuevo sistema realice los siguientes procedimientos:
  - a) Administrar los procesos de entradas y salidas de bienes.

- b) Generar reportes mensuales, que les informen de las existencias en la bodega, así como de las salidas de bienes de la bodega.

MOVISAT cuenta con los recursos materiales y tecnológicos por lo cual, se implementará un sistema informático, ya que la información estaría más segura en una base de datos y ya se cuenta con todo lo necesario para construir e implementar dicho sistema.

El sistema informático constará de tres partes:

- ☞ La base de datos.
- ☞ El equipo de cómputo y conexión a la red interna (nodo de red en la bodega).
- ☞ Servidor de páginas Web (este servidor contendrá las páginas que realizarán los procesos administrativos con respecto a los bienes).

### ***II.2.2.1 Base de datos***

Para desarrollar la base de datos se tomó en cuenta las entrevistas y de las cuales se obtuvieron los siguientes datos:

- ☞ Debe existir una tabla de usuarios para restringir el acceso al sistema
- ☞ También se tendrá una tabla para guardar en ella las entradas y otra tabla para las salidas de bienes.
- ☞ Habrá una tabla donde se guarden los artículos con sus existencias.
- ☞ Se tendrá un catalogo del CABM (Clave Administrativa de Bienes Muebles), ya que es como un CURP pero para los bienes, éste es proporcionado por la empresa en una hoja de Excel.
- ☞ Existirá un catálogo de Gerencias y Coordinaciones, ya que una coordinación pertenece a una gerencia, el porqué de ésta última tabla se debe a que necesitan reportes mensuales de cuanto se le dio a una gerencia.

Se investigó en las hojas de requisición (y otros documentos) los datos que se manejan en el sistema administrativo (no ponemos los datos que se manejan por cuestiones de seguridad de MOVISAT).

#### ***II.2.2.1.1 Análisis de las tareas y determinación de la finalidad de la D.B.***

En la gerencia de MOVISAT (TELECOMM) se cuenta con una bodega donde se almacenan los bienes que llegan para consumo de dicha gerencia. No se tiene un sistema informático con el cual administrar dichos bienes.

La forma en que actualmente se administran los bienes es la siguiente:

Cuando la gerencia de MOVISAT requiere una cierta cantidad de bienes la solicita a la gerencia de CTO (que también es parte de TELECOMM). El CTO entrega los bienes junto con lista de todo lo que se entrega. Los bienes son entregados en la bodega de la gerencia MOVISAT.

Cuando un bien entra a la bodega se debe registrar: CABM del bien (que es una clave única a nivel nacional), nombre del bien, descripción, pedido, licitación, partida general, requisición, partida requisición, unidad de medida, cantidad de unidades, precio por unidad, fecha de entrada y clave presupuestal.

El administrador general es el encargado del control de los bienes.

Los usuarios de las distintas coordinaciones pueden solicitar alguno o varios de los bienes, pero el administrador general es el que autoriza o niega la entrega de los bienes.

Si se autoriza un pedido se genera una hoja de salida con: un número de folio que se reinicializa cada año, fecha de salida de los bienes, fecha del pedido, número de pedido, nacionalidad del pedido, nombre de la persona que retira los bienes, coordinación a la cual pertenece la persona que retira los bienes, gerencia de la coordinación, partida de cada bien, descripción del bien, unidad de medida del bien y la cantidad retirada.

En la bodega hay un responsable, que se encarga de recibir los bienes por parte de CTO y entrega los bienes a los usuarios cuando el administrador general autoriza una salida.

Se necesita un sistema informático para registrar las entradas, salidas, existencias y pedidos de bienes de la bodega dicha gerencia, así como realizar informes que requiere el administrador.

Por seguridad los usuarios que realizan pedidos sólo podrán ver las existencias de los bienes y hacer los pedidos, pero no pueden modificar nada. El administrador tendrá acceso total al sistema y el responsable de la bodega verificará las entradas y salidas.

#### ***II.2.2.1.2 Esquematización del flujo de tareas***

- & Se tiene una bodega de bienes
- & Cada bien tiene ciertos atributos como el CAMB, descripción, etc.
- & Se deben registrar las entradas de los bienes con los datos más importantes, como nombre del bien, cantidad que entra, fecha, etc.
- & El retiro de los bienes se hace por medio de pedidos
- & Los pedidos son realizados por personal de otras coordinaciones
- & Los pedidos deben ser autorizados por el administrador general, es decir, se autoriza la salida de bienes
- & Los pedidos también deben contar con datos como, persona que solicita, nombre del bien, descripción del bien, etc.
- & Cuando un pedido es autorizado se genera una hoja de salida, que cuenta con datos como son no. pedido, fecha, cantidad que se retira, etc.

- & El responsable del almacén es el encargado de entregar los bienes y registrar la salida
- & Se deben actualizar las existencias de la bodega
- & El administrador, necesita hacer informes de lo que se le entrega a cada coordinación mensualmente, cuanto sale de la bodega mensualmente y cuantos bienes entran por mes
- & También se realizan informes anuales de lo que salió y entró a la bodega

### II.2.2.2 El equipo de cómputo y conexión a la red interna (nodo de red en la bodega)

Las opciones de software descritas aquí se obtuvieron por medio de la evaluación de los recursos de MOVISAT.

Software:

Opción A

Nombre del paquete	Descripción	Precio por licencia	Licencias requeridas	Total
Windows XP	Sistema operativo	\$3000	2	\$6000
Windows 2000	Sistema Operativo	\$2500	1	\$2500
Windows 2003 Server	Sistema Operativo	\$25000	1	\$25000
Linux(Slackware)	Sistema Operativo	----- -----	1	\$0
Apache	Servidor WWW	----- -----	1	\$0
Freetds	Librerías para PHP	----- -----	1	\$0
PHP	Preprocesador de HTML	----- -----	1	\$0
SQL Server 2000	Servidor de bases de datos	\$25000	1	\$25000
Quanta Editor de páginas WEB		----- -----	1	\$0
			<b>Gran total</b>	<b>58500</b>

Opción B

Nombre del paquete	Descripción	Precio por licencia	Licencias requeridas	Total
Windows 2000	Sistema operativo	\$2500	2	\$5000
Windows 2000	Sistema	\$2500	1	\$2500

	Operativo			
<b>Windows 2000 Server</b>	Sistema Operativo	\$20000	1	\$20000
<b>Linux(Slackware)</b>	Sistema Operativo	----- -----	1	\$0
<b>Apache</b>	Servidor WWW	----- -----	1	\$0
<b>Freetds</b>	Librerías para PHP	----- -----	1	\$0
<b>PHP</b>	Preprocesador de HTML	----- -----	1	\$0
<b>SQL Server 2000</b>	Servidor de bases de datos	\$25000	1	\$25000
<b>Quanta</b>	Editor de páginas WEB	----- -----	1	\$0
<b>Gran total</b>				<b>52500</b>

#### Hardware

<b>Equipo</b>	<b>Descripción</b>	<b>Unidad</b>	<b>Precio por unidad</b>	<b>Unidades requeridas</b>	<b>Total</b>
<b>Procesador Intel Celeron</b>	800Mhz	Pieza	\$200	2	\$400
<b>Mother board</b>	Para procesadores celeron	Pieza	\$200	2	\$400
<b>RAM</b>	256 PC-133	Pieza	\$400	2	\$800
<b>Disco Duro</b>	Maxtor 40GB	Pieza	\$500	2	\$1000
<b>Monitor</b>	Lanix 17pulgadas	Pieza	\$1000	2	\$2000
<b>Teclado</b>	Lanix	Pieza	\$100	2	\$200
<b>Mouse</b>	Lanix PS2	Pieza	\$100	2	\$200
<b>Tarjeta de red</b>	Motorola 10/100	Pieza	\$100	2	\$200
<b>SERVIDOR HP PROLIANT</b>	SERVIDOR	EQUIPO	\$80000	1	\$80000
<b>Cable UTP</b>	Cable 8 hilos	Metro	\$5	200	\$10000
<b>Gran total</b>				<b>95200</b>	

Se optó por la opción B en cuanto a software, porque MOVISAT ya cuenta con ese software.

La bodega no tiene un nodo de red, por lo que se tendrá que poner uno (Fig. 3).

**Nota: El nodo se instalará y configurará por parte de TELECOMM.**

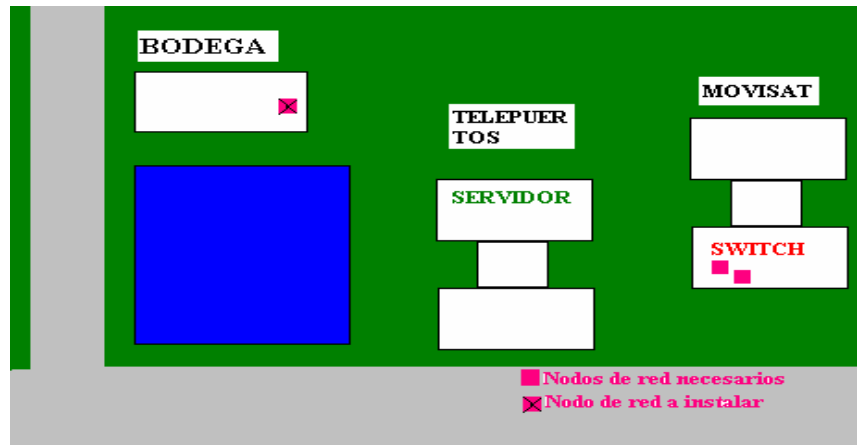


Fig. 3

### ***II.2.2.3 Servidor de páginas WEB***

Este servidor se ubicará en TELEPUERTOS (ver Fig.3), se usará una PC con sistema operativo Linux (Slackware) y el servidor de páginas WEB será Apache.

El lenguaje de programación que se usará es PHP, porque es gratuito, estable y fácil de manipular.

Sabiendo ya que la solución a nuestro problema es un sistema informático, pasaremos al siguiente capítulo para diseñar dicho sistema.

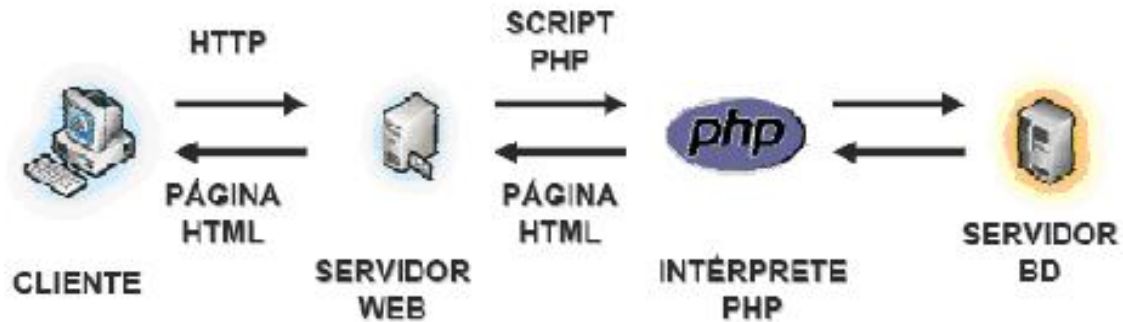


# Capítulo III

## Análisis y diseño del sistema

### III.1 Análisis de necesidades

Diagrama general del sistema.



El sistema funcionara de la siguiente forma:

1. Los usuarios accederán al sitio Web que se encuentra en nuestro servidor Web desde su computadora.
2. Después el sitio Web (que va a estar diseñado en PHP) ejecutará los scripts necesarios para brindarle al usuario la información que necesita.
3. Toda la información se almacenará en una base de datos localizada en un servidor de bases de datos.
4. Al final el usuario obtendrá en su pantalla la información que necesitaba.

### III.2 Análisis costo beneficio

En la gerencia de MOVISAT ya se cuenta con el hardware y software necesario. Así mismo se cuenta con el material requerido para instalar un nuevo nodo de red. Con esto la gerencia además de mejorar su sistema administrativo ahorrará en el desarrollo de éste.

Un beneficio adicional es que ahorrarán dinero al ya no tener tantas pérdidas de bienes, el tiempo que los usuarios inviertan en realizar los trámites será menor y se tendrá un mejor control de las operaciones.

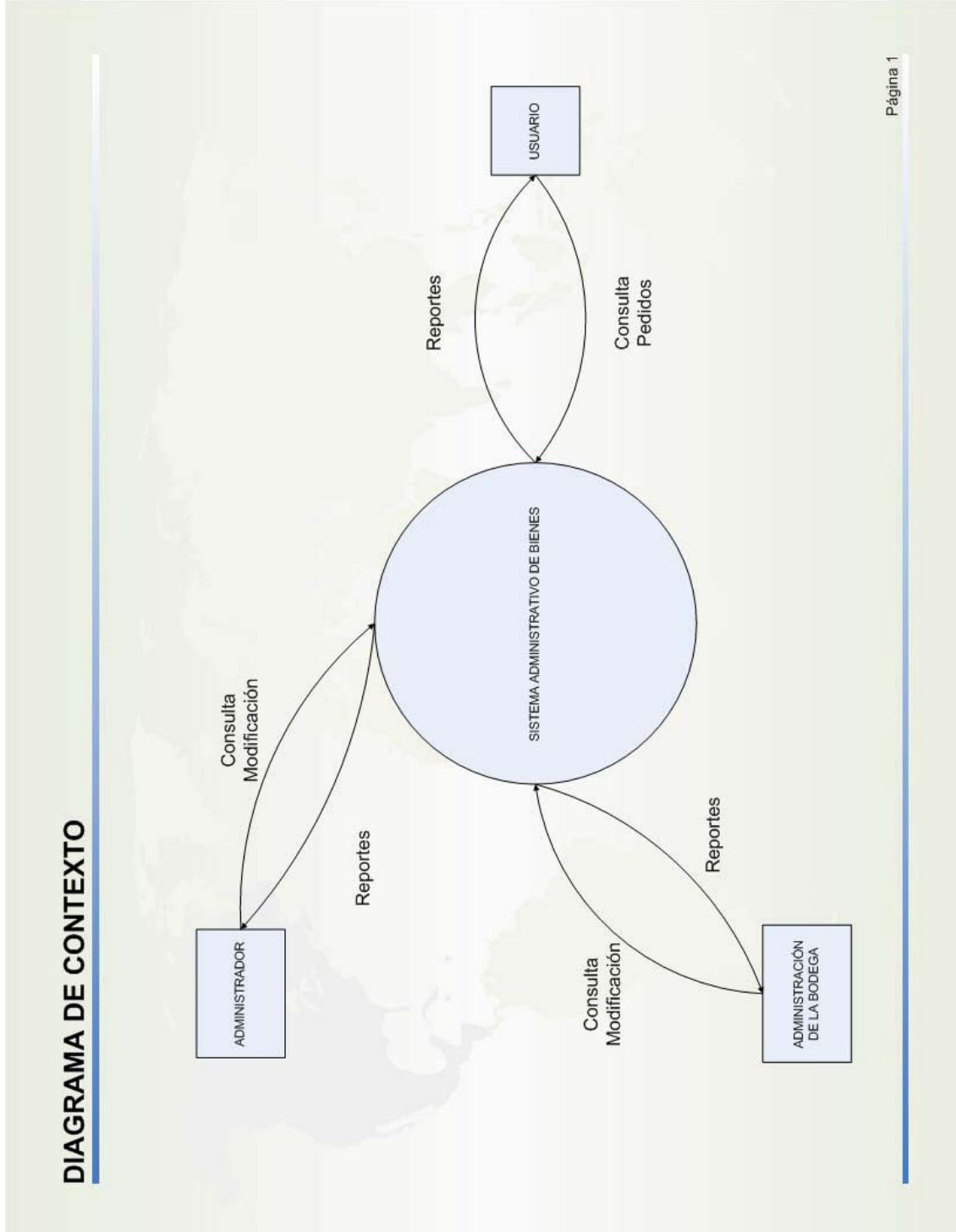
	<b>COSTO</b>	<b>BENEFICIO</b>
Costo del software	\$0	
Costo del hardware	\$0	
Ahorro en dinero de horas		\$22000/año

hombre que ya no se perderán		
Ahorro en dinero de los objetos que ya no se perderán		\$15000/año
<b>Gran total</b>	<b>\$0</b>	<b>\$37000/año</b>

**Como vemos se ahorraran \$37,000 pesos al año y los procesos administrativos serán más rápidos, por lo tanto el sistema informático es la mejor opción.**

### III.3 Diseño del Sistema.

#### III.3.1 Diagrama de Flujo de Datos

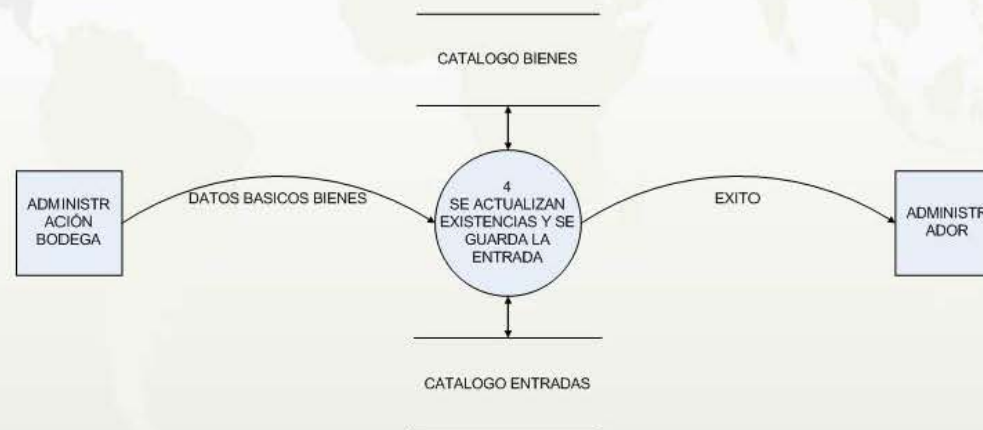


## PROCESO DE SALIDA DE BIENES Y PROCESO DE ENTRADA DE BIENES

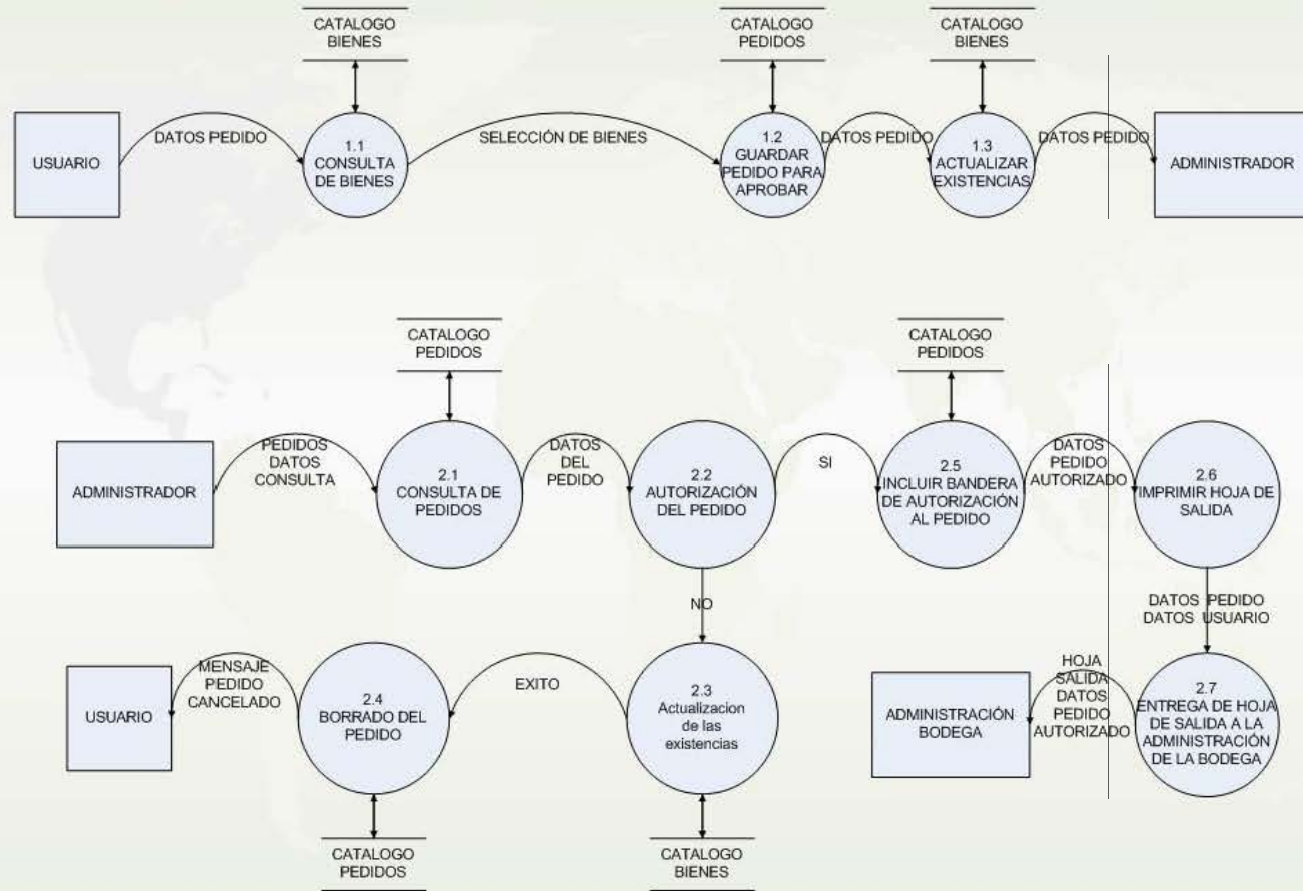
### PROCESO DE SALIDA DE BIENES



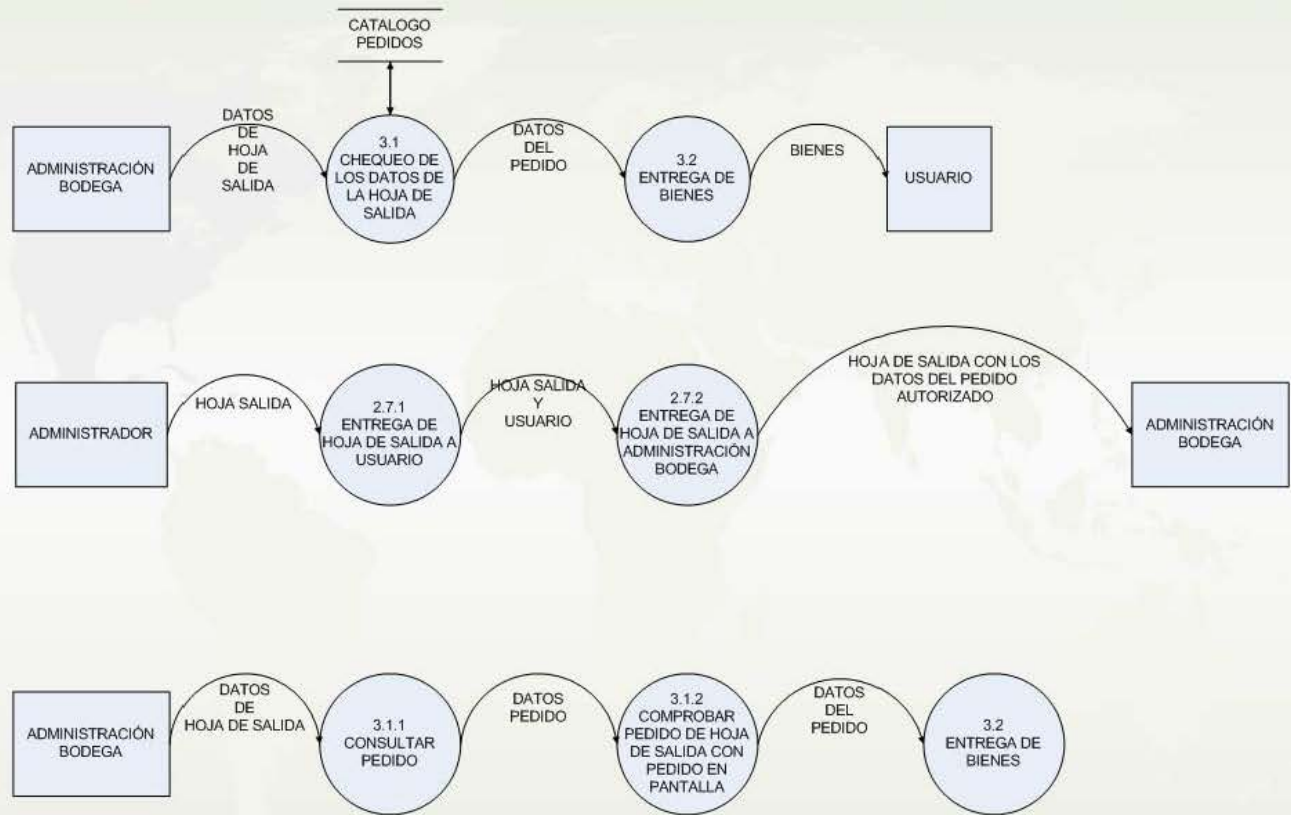
### PROCESO DE ENTRADA DE BIENES



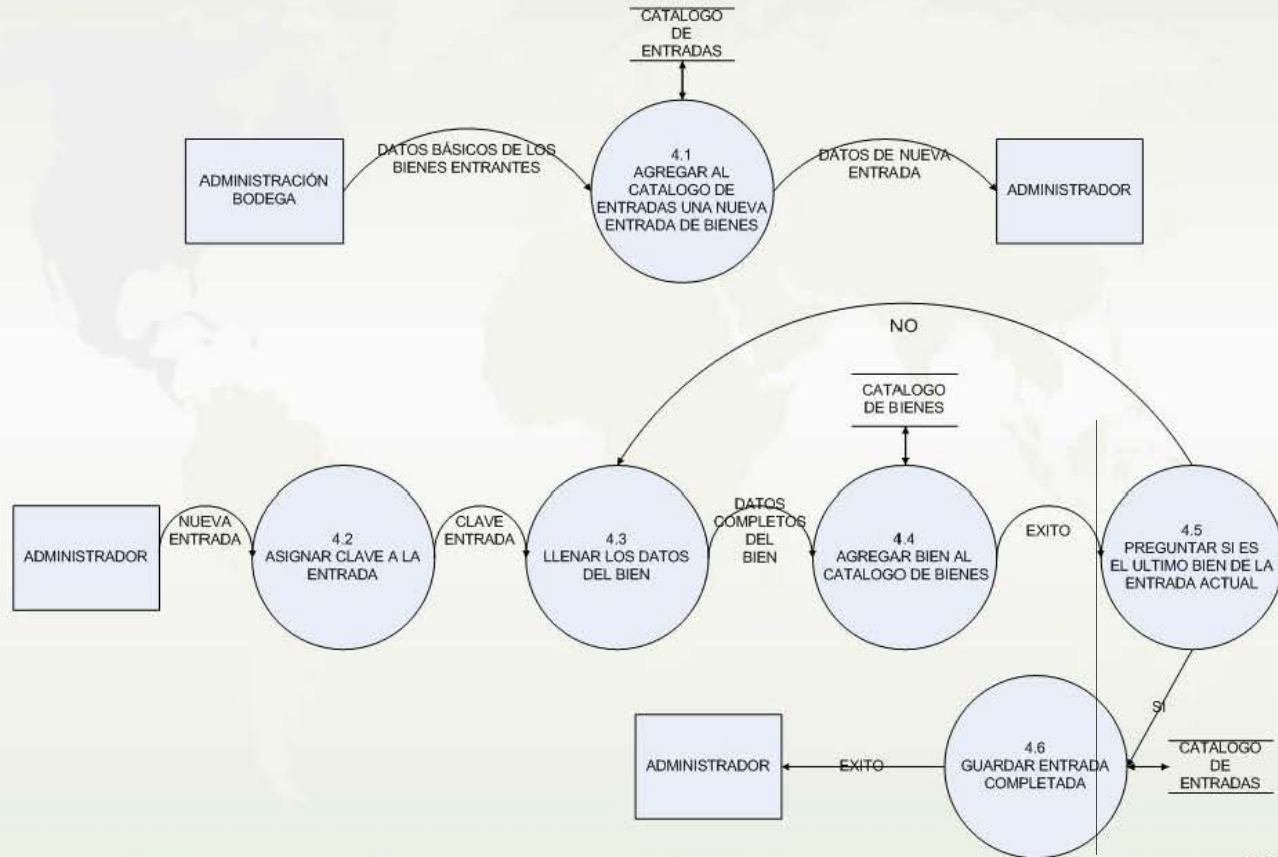
## PROCESO DE SALIDA DE BIENES



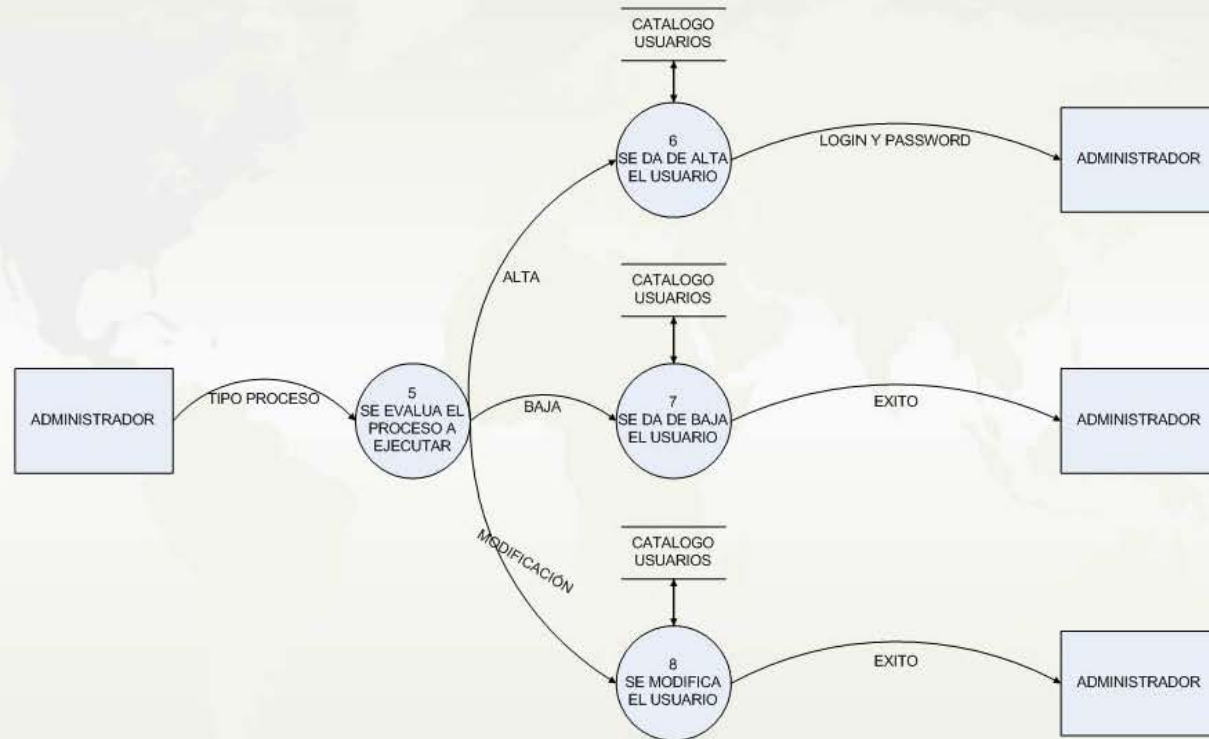
## PROCESO DE SALIDA DE BIENES



## PROCESO DE ENTRADA DE BIENES

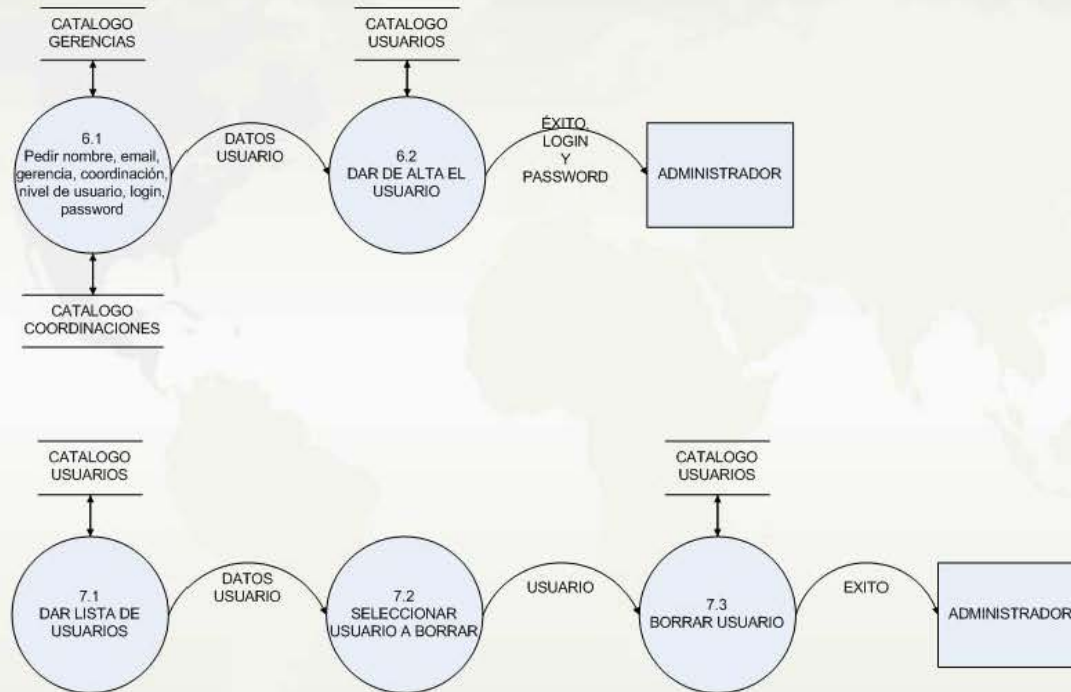


## PROCESO ADMINISTRATIVO DE USUARIOS

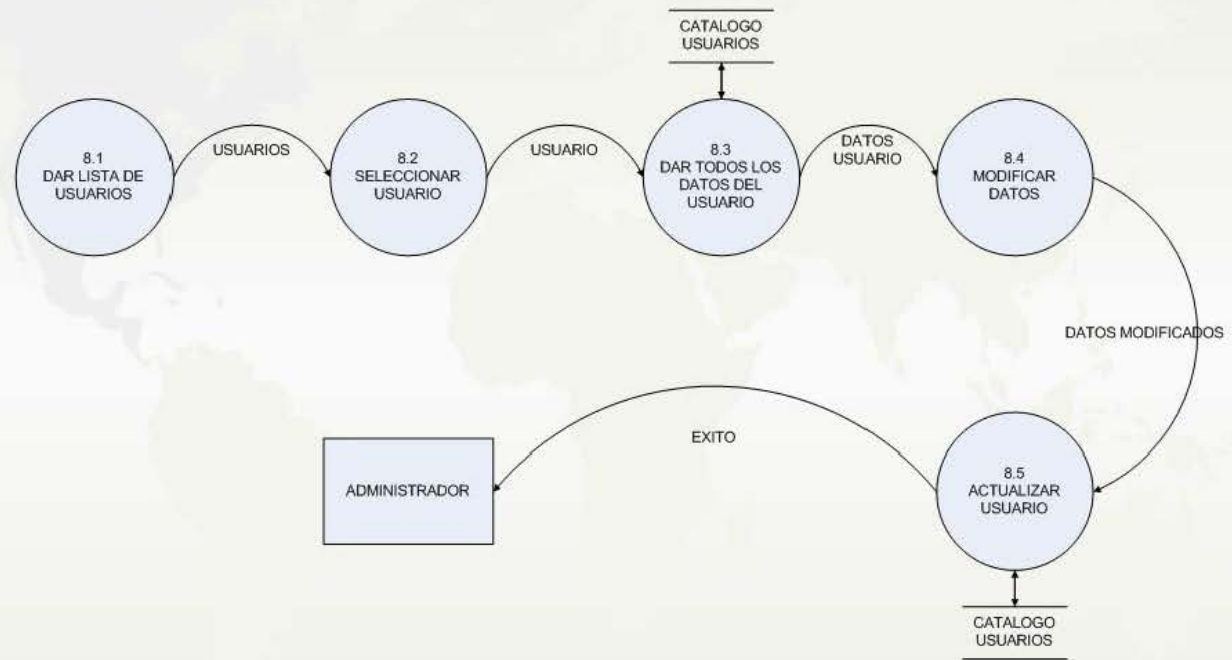




## PROCESO ADMINISTRATIVO DE USUARIOS



## PROCESO ADMINISTRATIVO DE USUARIOS

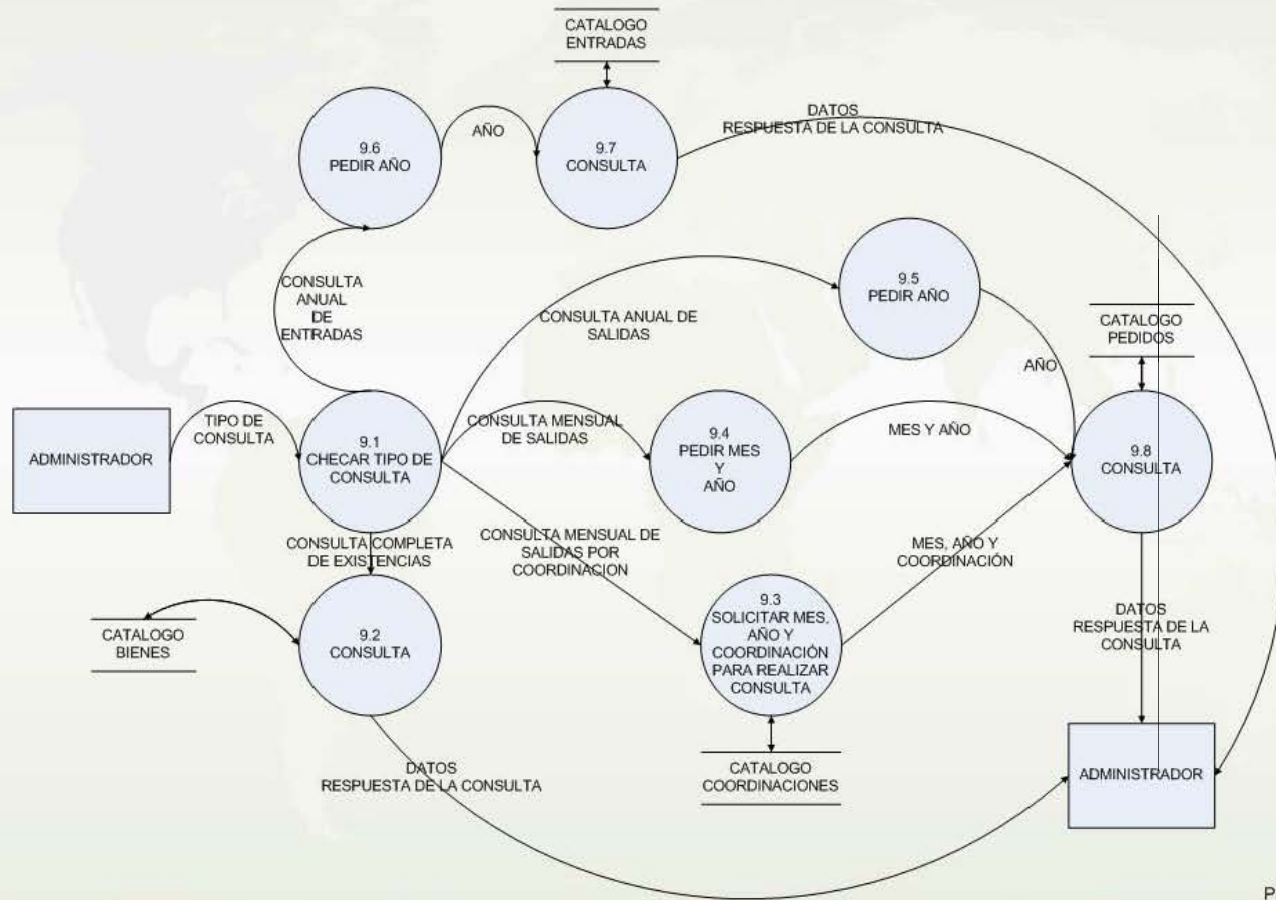


## PROCESO DE CONSULTAS Y PROCESO ADMINISTRATIVO DE PRODUCTOS

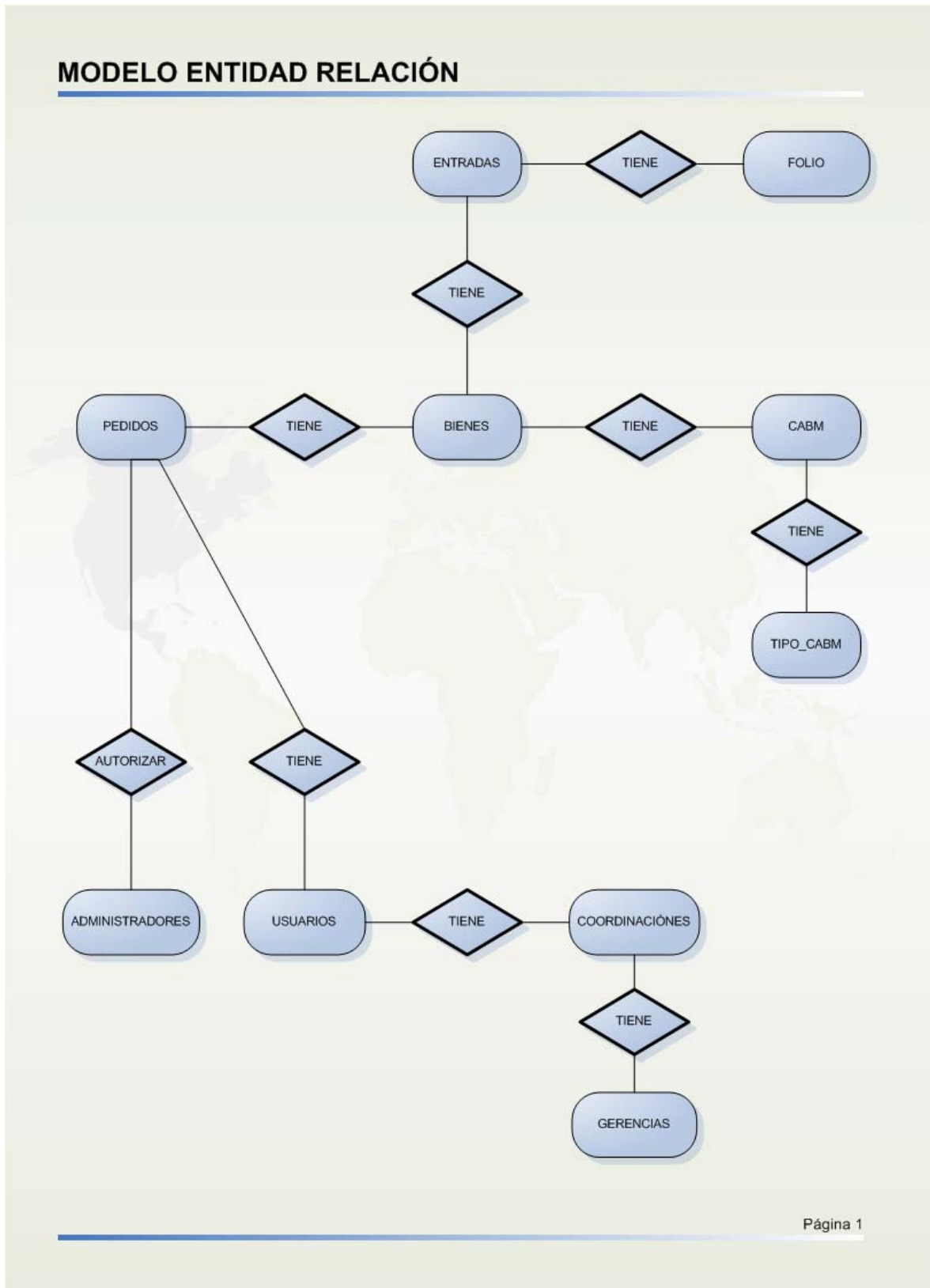
### PROCESO DE CONSULTAS



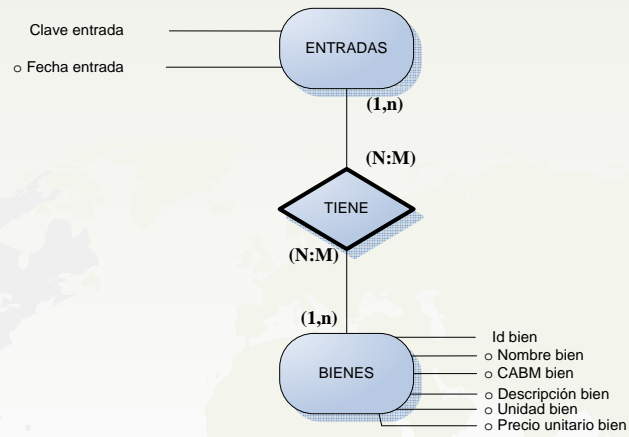
## PROCESO DE CONSULTAS



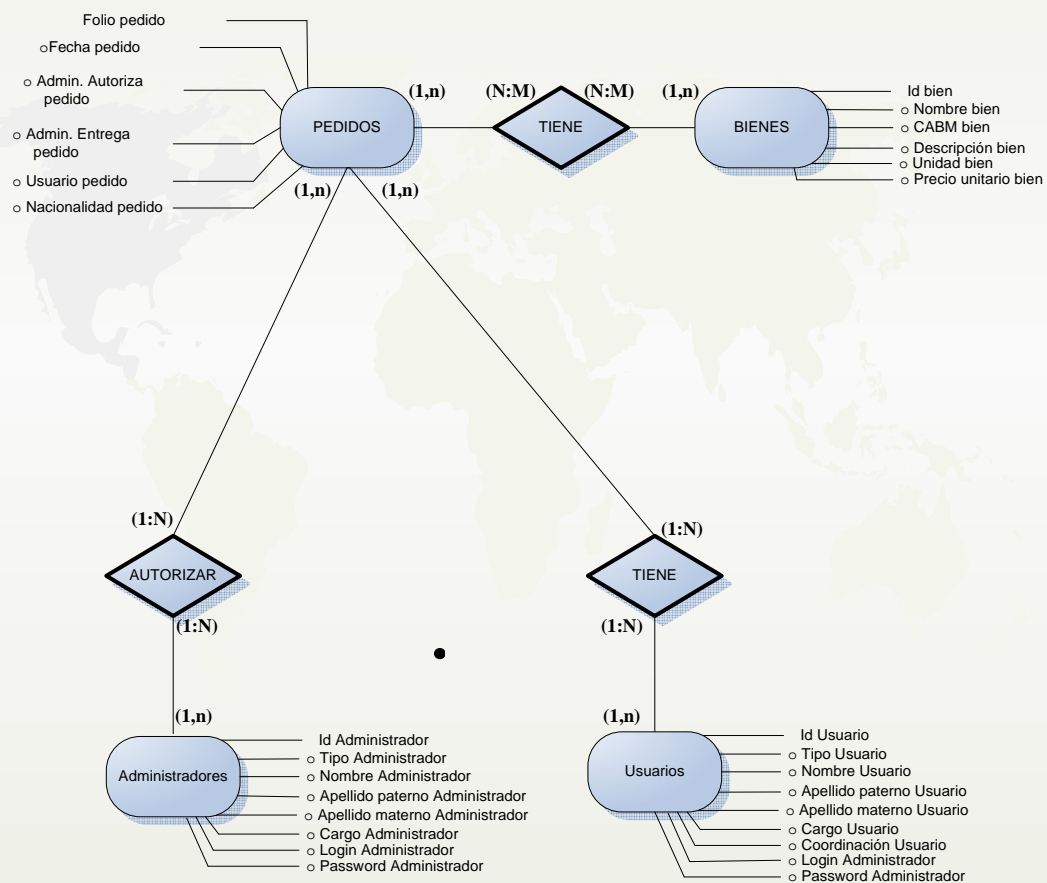
### III.3.2 Modelo Entidad Relación



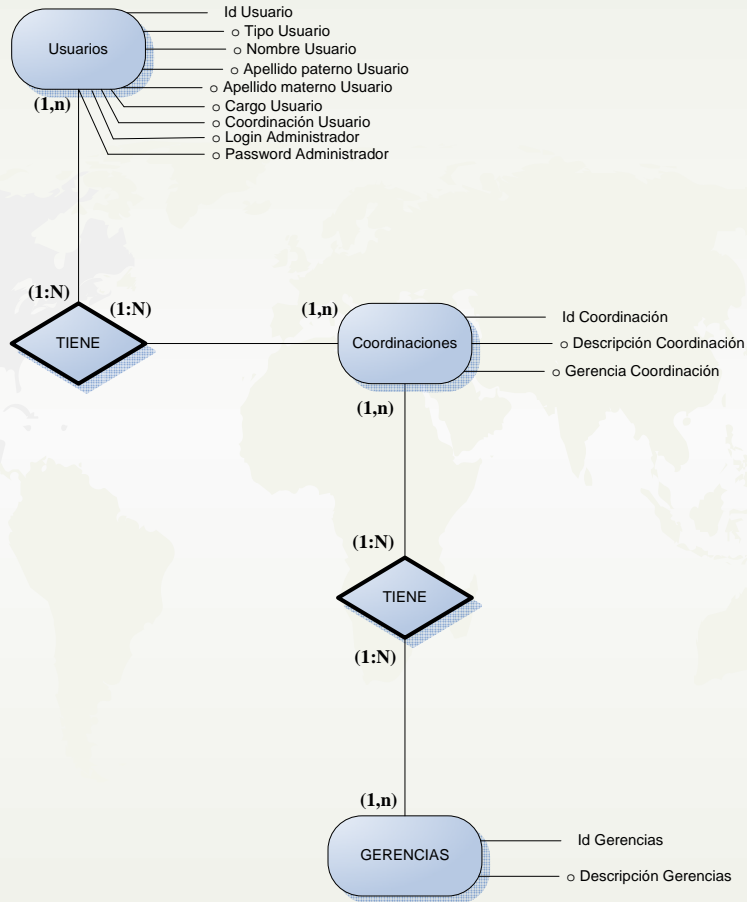
# MODELO ENTIDAD RELACIÓN



# MODELO ENTIDAD RELACIÓN

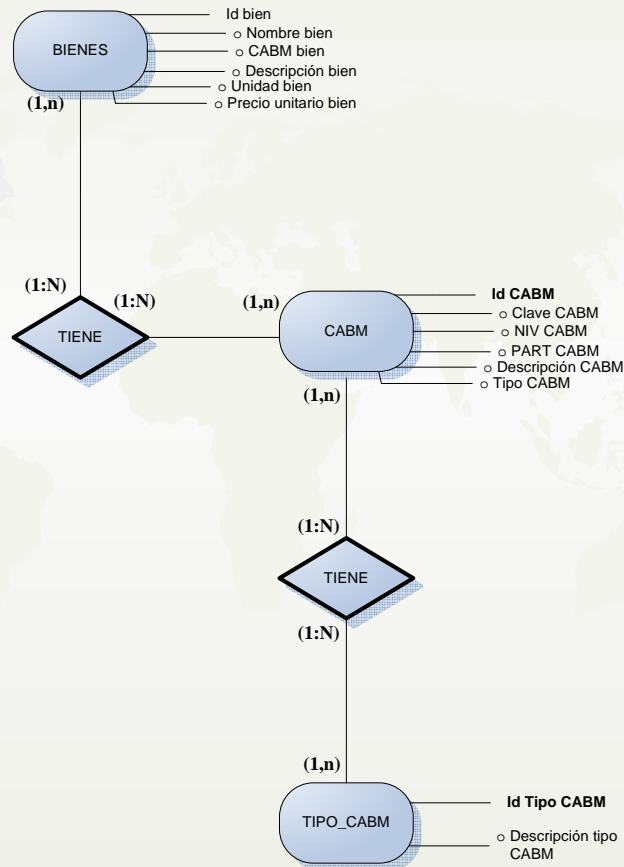


# MODELO ENTIDAD RELACIÓN





# MODELO ENTIDAD RELACIÓN



### III.3.3 Diseño de la base de datos

#### Determinación de las tablas

- ✦ Entradas
- ✦ Bienes
- ✦ CABM
- ✦ TIPO CABM
- ✦ Pedidos
- ✦ Usuarios
- ✦ Coordinaciones
- ✦ Gerencias
- ✦ Administradores
- ✦ Folio
- ✦ Folioe

Determinación de las relaciones entre las tablas.

Entradas	↔	Bienes
CABM	→	Bienes
Usuario	→	Pedidos
Usuario	←	Coordinaciones
Pedidos	↔	Bienes
Administradores	→	Pedidos
Coordinaciones	←	Gerencias
CABM	←	Tipo CABM

#### SIMBOLOGIA

→ Uno a muchos  
↔ Muchos a muchos

Determinación de los posibles campos que requieren las tablas.

Ya que una entrada puede tener varios bienes y varios bienes pueden corresponder a una entrada se crea una tabla intermedia.

#### ENTRADAS

clv\_entrada  
fecha\_entrada

#### ENTRADAS\_BIENES

clv\_entrada  
id\_bien  
cant\_entra\_bien

#### BIENES

id\_bien  
nom\_bien  
cabm\_bien  
descrip\_bien  
unidad\_bien  
precio\_unit\_bien

Se crea la tabla TIPO\_CABM ya que dentro del catalogo del CABM se divide en varios tipos, como son accesorios y material eléctrico, artículos deportivos y de campaña, etc. y éstos a su vez se dividen en más accesorios.

### **TIPO\_CABM**

id\_tipo\_cabm  
descrip\_tipo\_cabm

### **CATALOGO\_CABM**

id\_cabm  
clave\_cabm  
niv\_cabm  
part\_cabm  
descrip\_cabm  
tipo\_cabm

Ya que existe una relación de muchos a muchos entre los pedidos y las existencias de los bienes se crea una tabla intermedia llamada “PEDIDOS\_ BIENES” y el resto de las tablas quedan sin modificación.

### **PEDIDOS\_BIENES**

folio\_pedido  
id\_bien  
cant\_pedido  
partida

### **FOLIO**

id\_folio  
ultimo\_folio

### **PEDIDOS**

folio\_pedido  
fecha\_pedido  
admin.\_autor\_pedido  
admin.\_entreg\_pedido  
usuario\_pedido  
nacion\_pedido

### **FOLIOE**

id\_folio  
ultimo\_folio

### **COORDINACIONES**

id\_coordina  
decrip\_coordina

### **ADMINISTRADORES**

id\_admin.  
tipo\_admin.  
nom\_admin.  
ap\_pat\_admin  
ap\_mat\_admin  
cargo\_admin  
login\_admin  
password\_admin  
prioridad\_admin

### **GERENCIAS**

id\_gerencia  
decrip\_gerencia

### **USUARIOS**

id\_usuario  
nom\_usuario  
ap\_pat\_usuario  
ap\_mat\_usuario  
cargo\_usuario  
coord\_usuario  
login\_usuario  
password\_usuario

Identificar los campos obligatorios, únicos y opcionales, así como identificar el tipo de dato, longitud de los campos y llaves.

### ENTRADAS

PK #\*clv\_entrada  
T(11)

### ENTRADAS\_BIENES

PK #\*clv\_entrada T(11)  
PK \*id\_bien N  
\*cant entra bien N

### BIENES

PK #\*id\_bien N  
\*nom\_bien T(50)  
\*cabm\_bien N  
\*descrip\_bien T(50)  
\*unidad\_bien T(50)  
\*precio\_unit\_bien N  
\*cant exist N

### TIPO\_CABM

PK #\*id\_tipo\_cabm N  
\*descrip\_tipo\_cabm T(70)

### PEDIDOS

PK #\*folio\_pedido T(11)  
\*fecha\_pedido F  
\*admin.\_autor\_pedido N  
\*admin.\_entreg\_pedido N  
\*usuario\_pedido N  
\*nacion\_pedido T(50)  
\*status\_pedido N

### CATALOGO\_CABM

PK #\*id\_cabm N  
\*clave\_cabm T(10)  
\*niv\_cabm N  
\*part\_cabm N  
\*descrip\_cabm T(255)

### PEDIDOS\_BIENES

PK #\*folio\_pedido T(11)  
\*id\_bien N  
\*cant\_pedido N

### USUARIOS

PK #\*id\_usuario N  
\*nom\_usuario T(50)  
\*ap\_pat\_usuario T(50)  
\*ap\_mat\_usuario T(50)  
\*cargo\_usuario T(50)  
\*coord\_usuario N  
\*login\_usuario T(50)  
\*password\_usuario T(50)

### COORDINACIONES

PK #\*id\_coordina N  
\*decrip\_coordina T(50)  
\*gerencia coordina N

### **GERENCIAS**

id\_gerencia N  
decrip\_gerencia T(50)

### **FOLIO**

id\_folio N  
ultimo\_folio T(11)

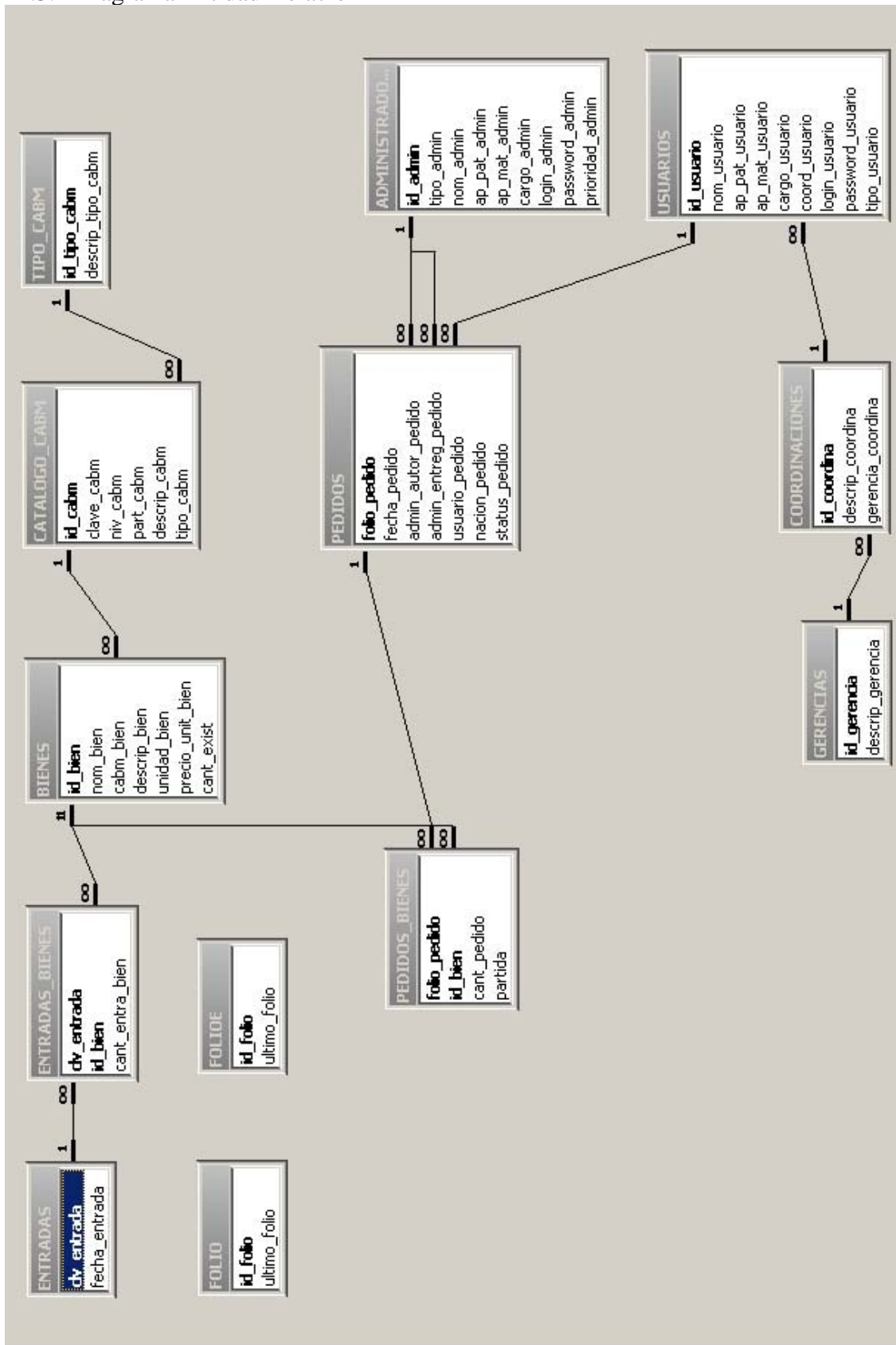
### **FOLIOE**

id\_folio N  
ultimo\_folio T(11)

### **ADMINISTRADORES**

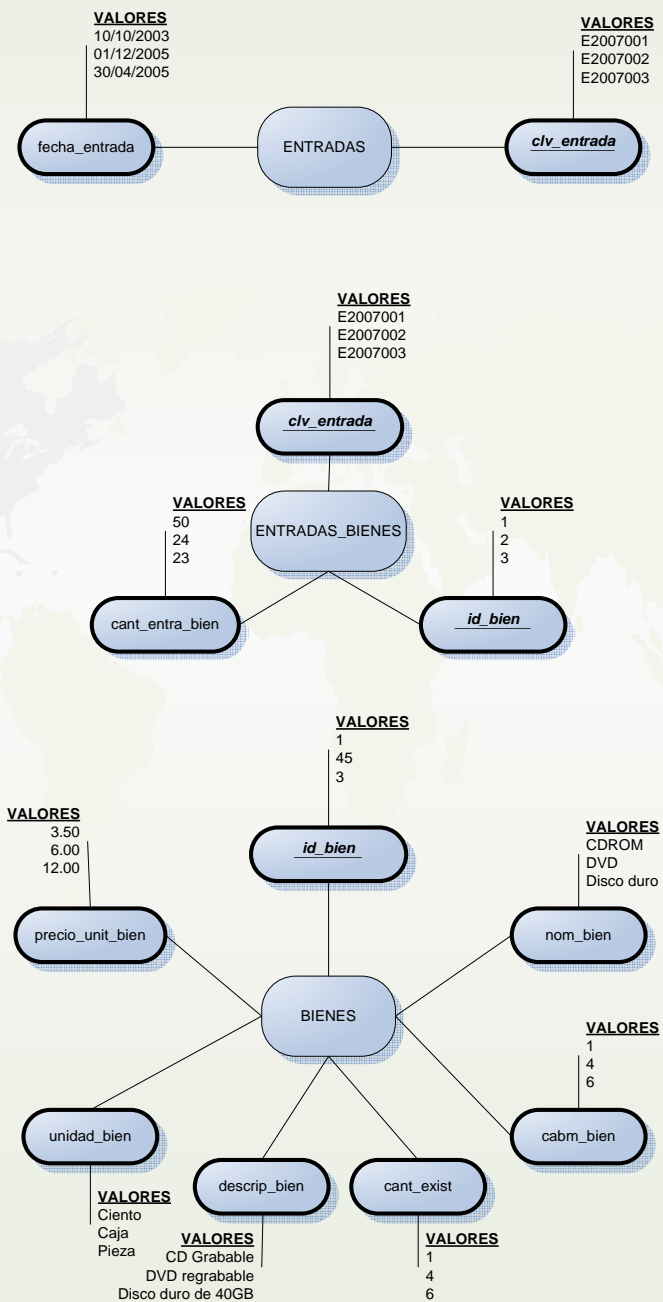
PK #\*id\_admin. N  
\*tipo\_admin. N  
\*nom\_admin. T(50)  
\*ap\_pat\_admin T(50)  
\*ap\_mat\_admin T(50)  
\*cargo\_admin T(50)  
\*login\_admin T(50)  
\*password\_admin T(50)  
\*prioridad\_admin N

### III.3.4 Diagrama Entidad-Relación

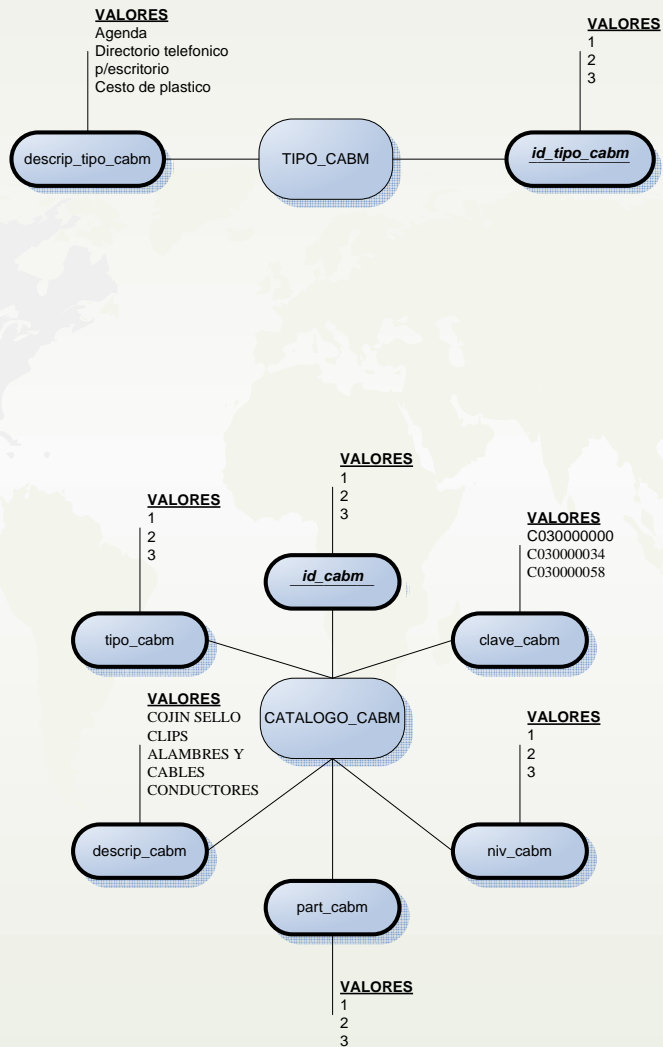


### III.3.5 Identificación de posibles valores que el usuario introducirá al sistema

#### TABLAS: ENTRADAS, ENTRADAS\_BIENES y BIENES

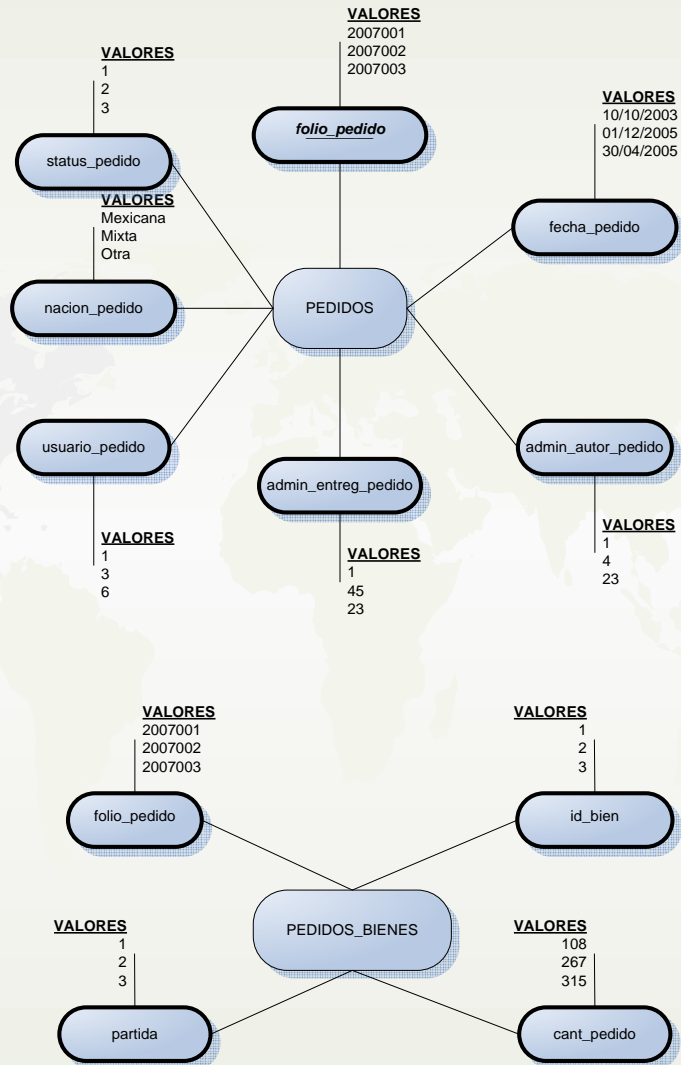


## TABLAS: TIPO\_CABM, EXIST\_BIENES y CATALOGO\_CABM

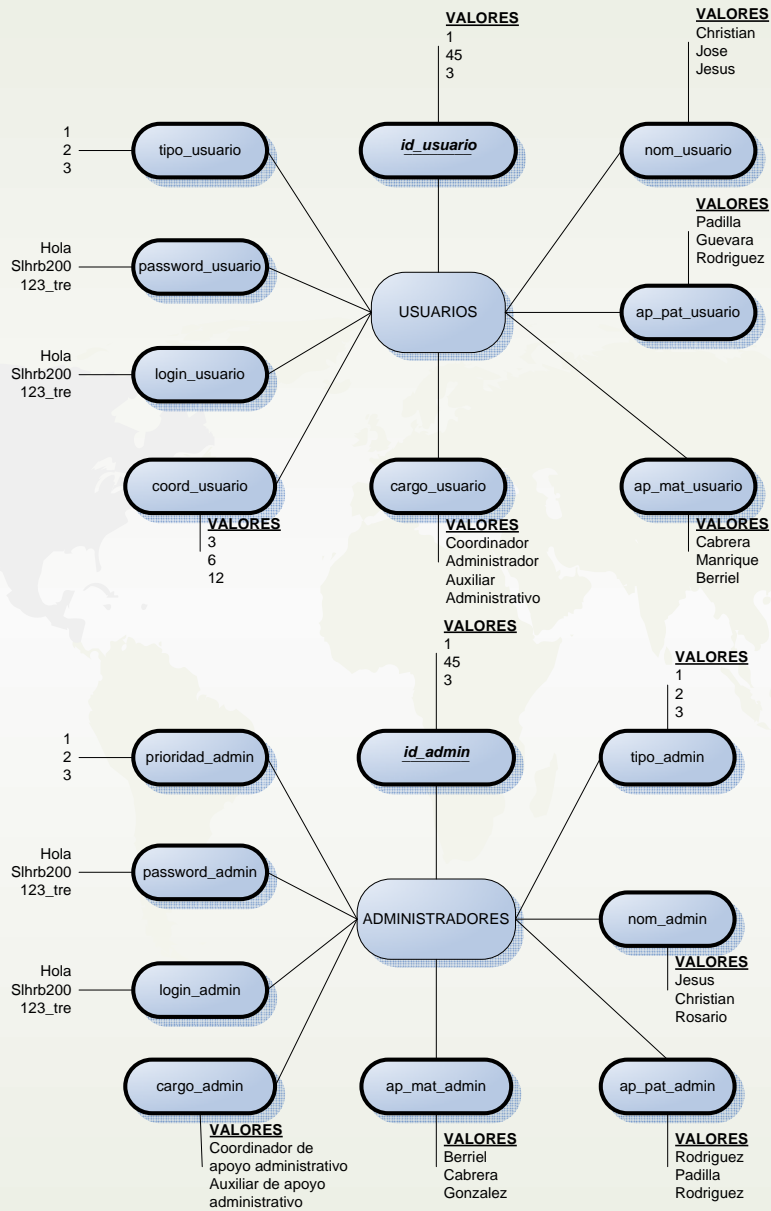




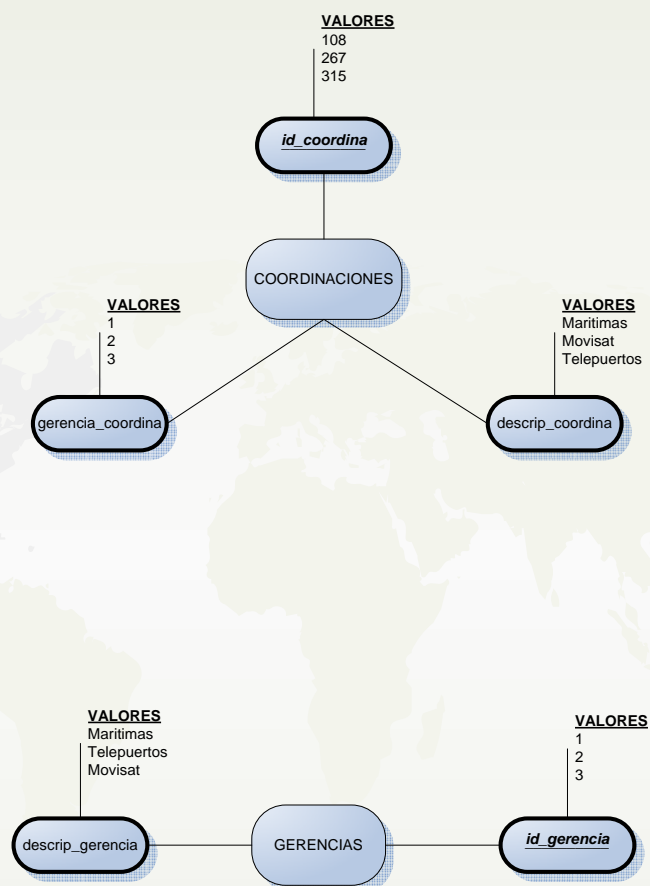
## TABLAS: PEDIDOS y PEDIDOS\_BIENES



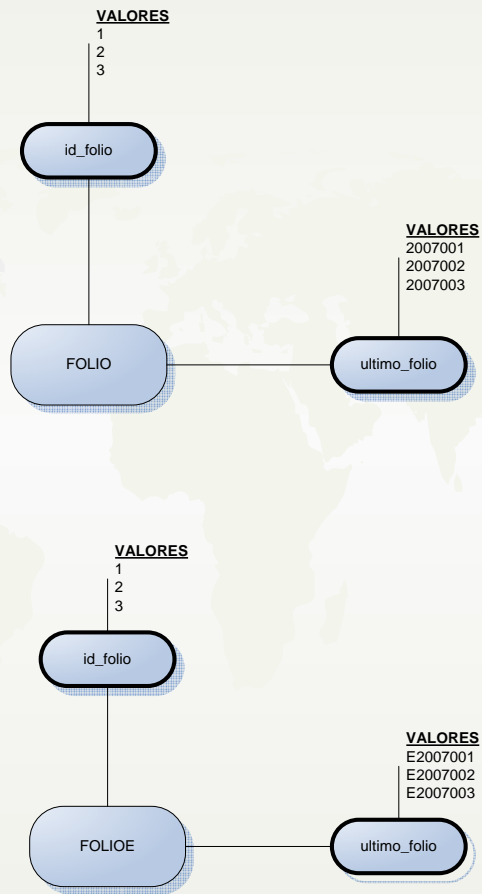
## TABLAS: USUARIOS y ADMINISTRADORES



## TABLAS: COORDINACIONES y GERENCIAS



## TABLAS: FOLIO Y FOLIOE



### III.3.6 Ruta Crítica

Actividades para realizar el sistema:

- A. Entrevistar al administrador y al encargado de bodega (2 días)
- B. Analizar la información de las entrevistas (3 días)
- C. Identificación del problema (1 día)
- D. Identificación de los procesos del sistema actual (1 semana)
- E. Identificación de los recursos con los que cuenta MOVISAT (1 día)
- F. Planteamiento de la solución (3 días)
- G. Determinación de los procesos del nuevo sistema (1 semana)
- H. Diseño de la base de datos (1 mes)
- I. Diseño del sitio Web (1 semana)
- J. Desarrollo de la base de datos (2 semanas)
- K. Desarrollo del sitio Web (1 mes)
- L. Pruebas del nuevo sistema (2 semanas)
- M. Corrección de errores (2 semanas)
- N. Elaboración de manuales (2 semanas)
- O. Capacitación del personal para el uso del sistema (2 semanas)
- P. Implementación del nuevo sistema (1 semana)

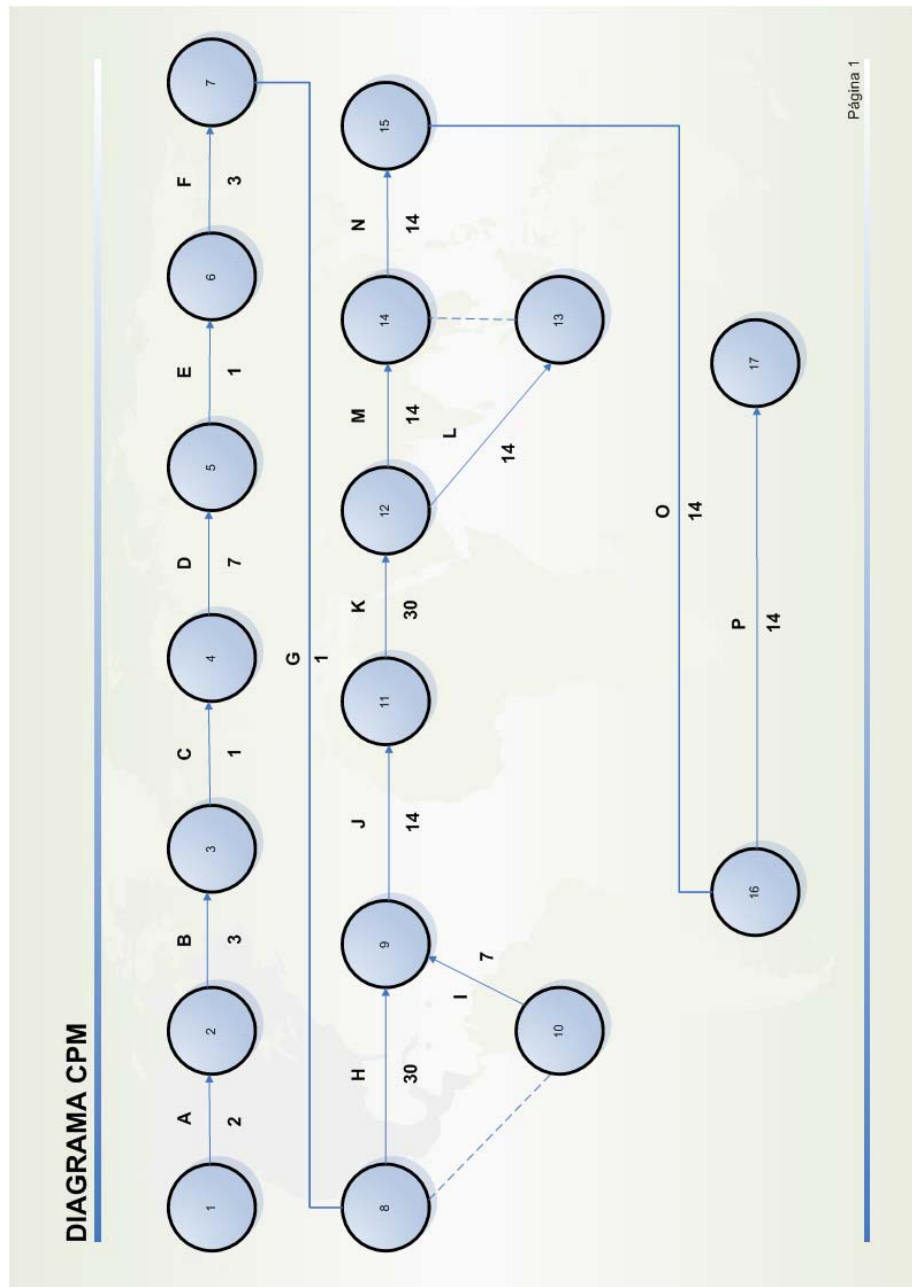
Matriz de secuencias.

Actividad	Antecedente	T. Optimo (o)	T. Normal (m)	T. Pésimo (p)	T. Esperado (Te) $T_e = \frac{o + 4m + p}{6}$
A		1	2	2	2
B	A	2	3	4	3
C	B	1	1	2	1
D	C	5	7	8	7
E	D	1	1	2	1
F	E	2	3	5	3
G	F	5	7	10	7
H	G	27	30	31	30
I	H <sub>ff+23</sub>	5	7	10	7
J	H	12	14	19	14.5
K	I,J	27	30	35	30
L	K	10	14	14	13
M	K	10	14	14	14
N	L,M	10	14	16	14
O	N	12	14	16	14
P	O	5	7	8	7
<b>Tiempo total</b>		120	147	172	<b>147.5</b>

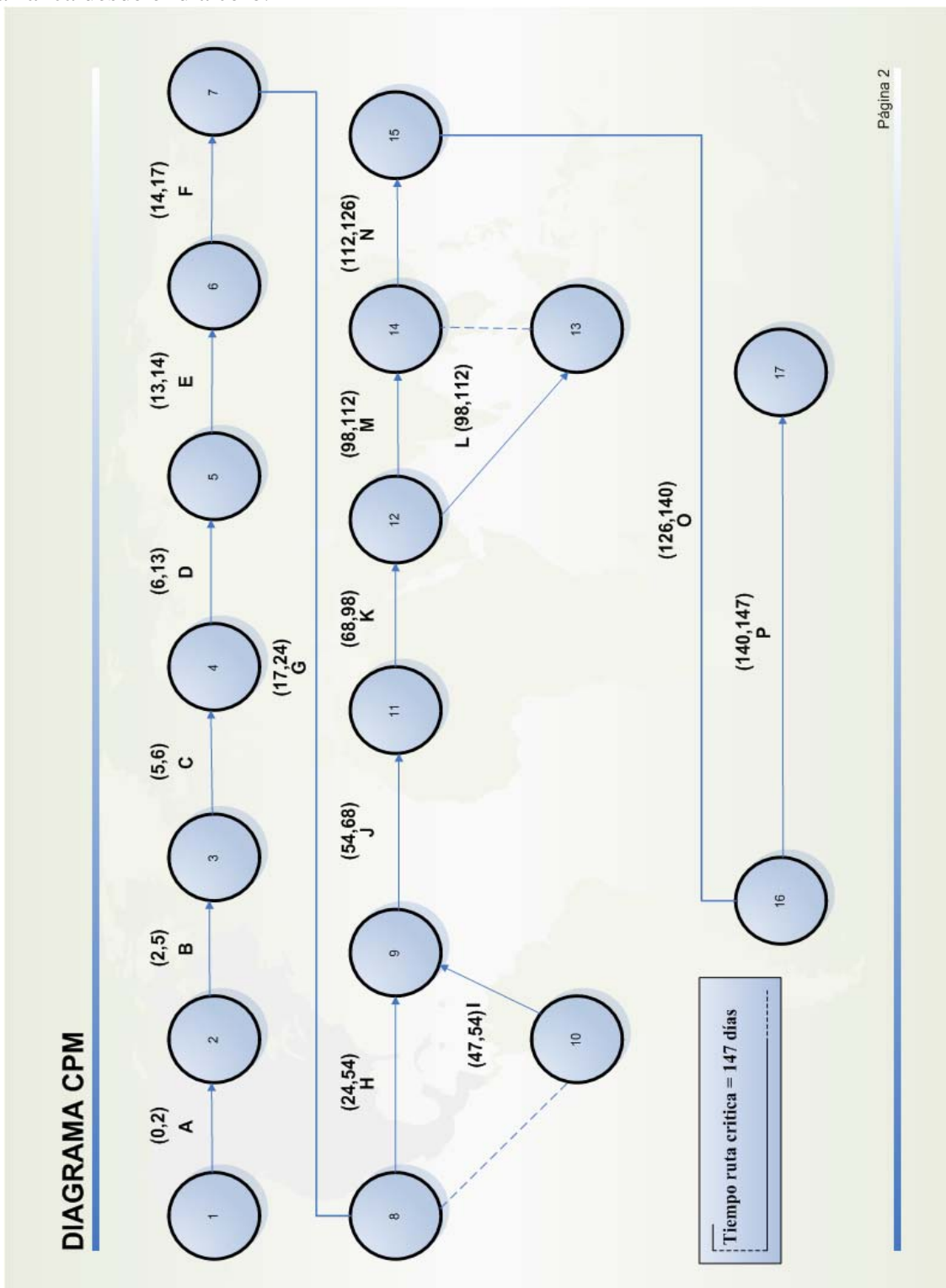
De la tabla anterior se observa que se tiene esperado que el proyecto termine en aproximadamente 147 días y medio, que el tiempo pésimo para terminar es de 172 días y que el tiempo óptimo es de 120 días.

Para saber el tiempo crítico en que se debe terminar el proyecto, realizamos el método de la ruta crítica.

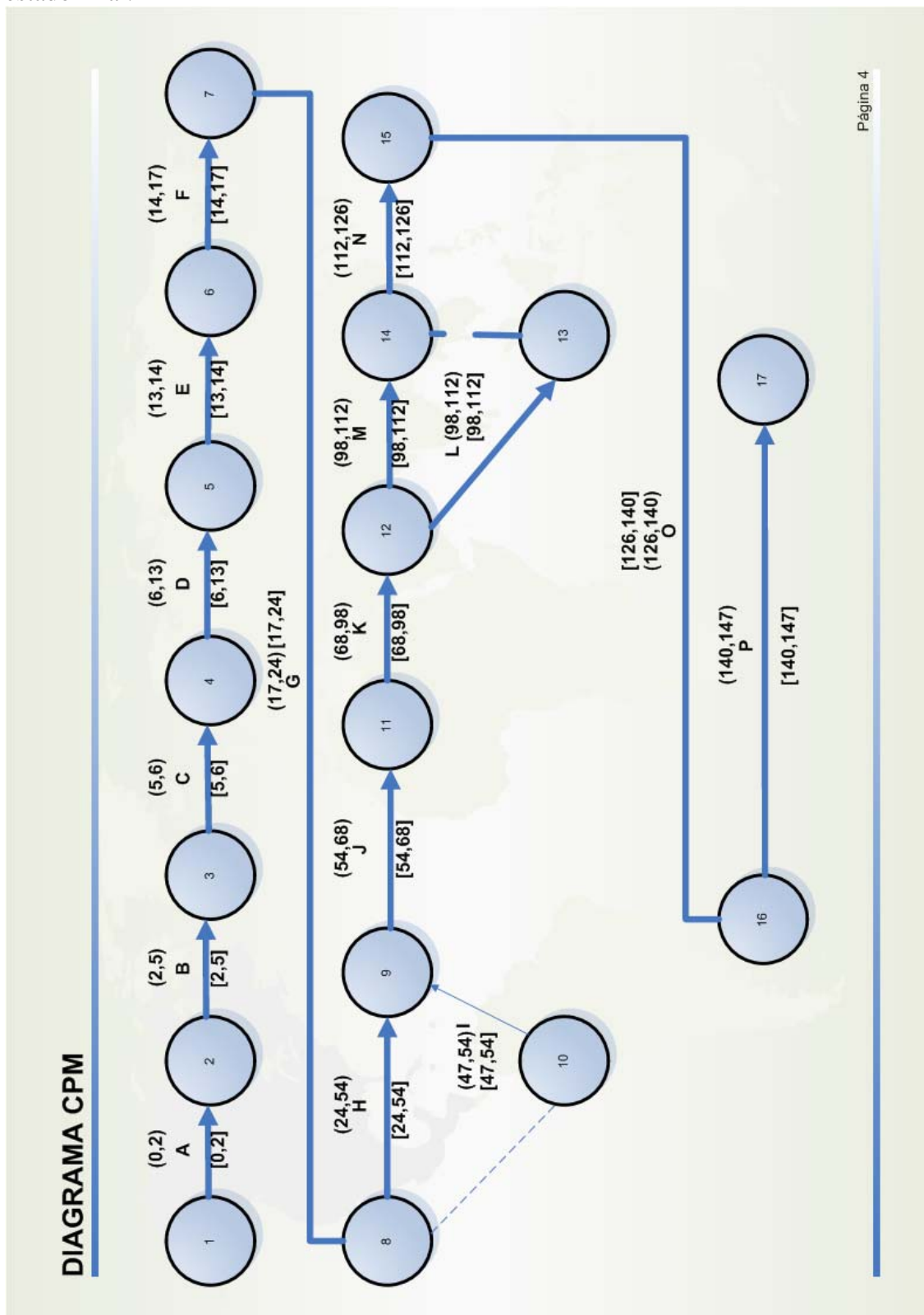
Primero se obtiene el diagrama de red por medio de las tareas. En este diagrama las flechas representan las tareas, los círculos son los estados a los que se llegan después de realizar una tarea y el número debajo de la flecha es la duración de cada tarea:



Después se sacan los intervalos tomando como día de inicio, el día final de la tarea anterior, el intervalo durará el periodo de tiempo especificado en el diagrama de red. El proyecto arranca desde el día cero:



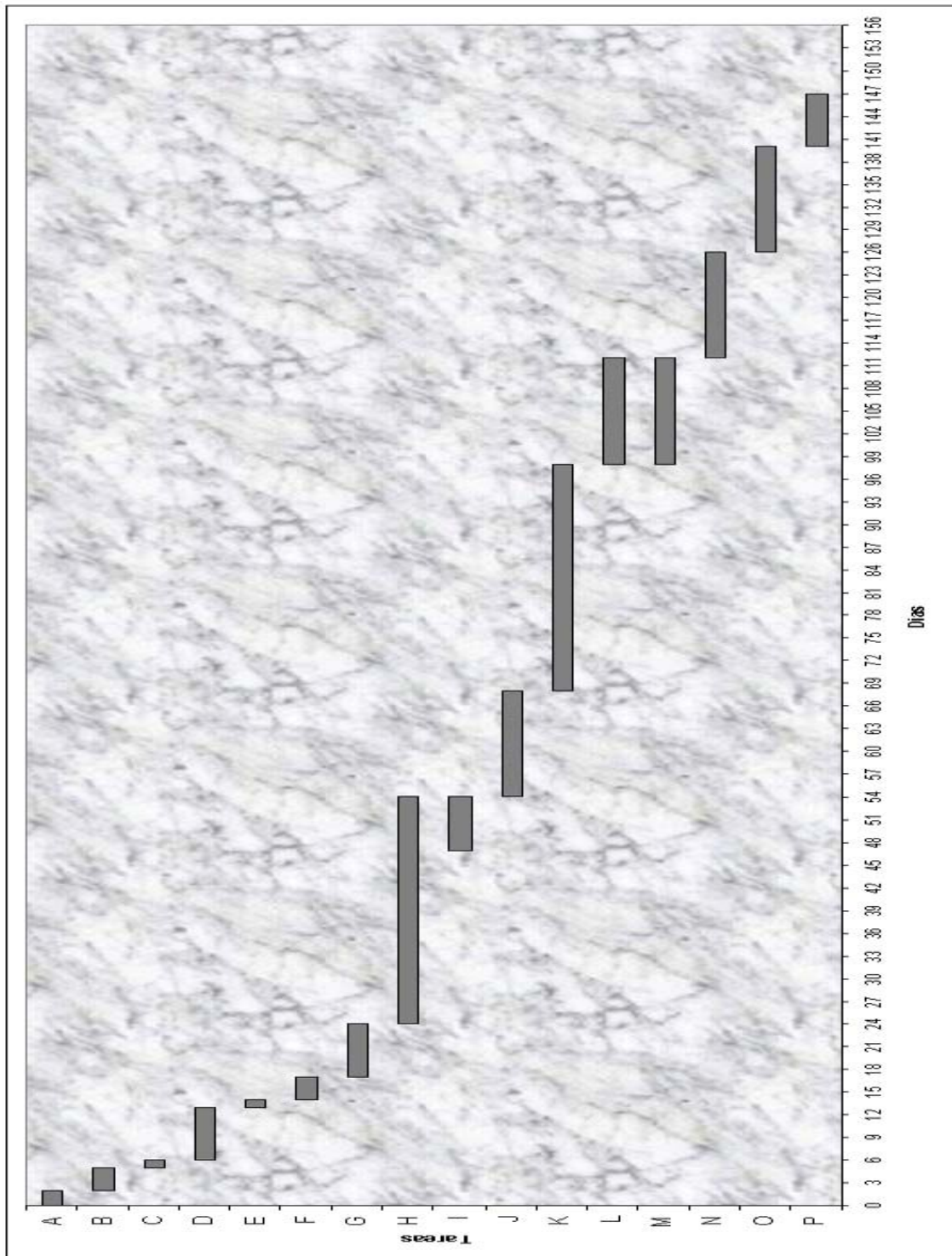
Después se saca la ruta crítica tomando en cuenta el trayecto más largo desde el estado 1 al estado final:





Por lo tanto el tiempo para terminar el proyecto, es a más tardar en 147 días, este resultado nos indica que no se puede atrasar o adelantar alguna tarea, todo se debe realizar en los tiempos contemplados para cada tarea.

Por último hacemos un diagrama de GANTT para ver gráficamente los tiempos de cada actividad, y observar que hay varias tareas que se ejecutan simultáneamente.



Gracias a todo lo visto en este capítulo, tenemos el diseño de nuestro sistema así como los tiempos para el desarrollo de éste. Por lo tanto, pasaremos al siguiente capítulo en el cuál se desarrollará el código fuente de nuestro sistema.

## Capítulo IV

# Desarrollo e implementación del sistema

### IV.1 Codificación

Se comentan las partes más importantes de todo el código fuente, para dar a entender el funcionamiento del sistema.

#### IV.1.1 Portal.php

Esta página es importante ya que por ésta acceden todos los usuarios al sistema. Esta página tiene el código siguiente:

Con esta etiqueta se inicia el código html.

```
<html>
  <head>
```

Después le ponemos título a nuestra página.

```
    <title>Portal</title>
  </head>
```

Le asignamos un fondo de pantalla y empezamos a escribir el cuerpo de nuestra página.

```
  <body background=" ./imagenes/fondop0.jpg" >
    <center>
```

En nuestra página incluimos el logotipo de TELECOMM con un ancho de 325 y un alto de 70, alineado hacia el centro y sin borde alguno

```
      <IMG src="imagenes/logo.jpg" width="325" height="70"
        align="center" border="0">
```

Damos dos saltos de línea e iniciamos un formulario para poder enviar la información a otra página, la información que queremos enviar es el “login” y el “usuario” por lo tanto le ponemos dos cajas de texto para que el usuario introduzca en ellas dichos datos.

```
      <br>
      <br>
      <form method="POST" action="EvaluaUsuario.php">
      <font color="white">USUARIO:</font><input type="text"
        name="usuario" maxlength="50"><br><br>
      <br><font color="white">PASSWORD:</font><input type="password"
        name="password" maxlength="50"><br><br>
```

Agregamos un botón tipo “submit” para enviar los datos a la siguiente página que es EvaluaUsuario.php.

```
}
```

```
<input type="submit" name="enviar" value="Entrar">
```

Por último cerramos el formulario el cuerpo y el código html

```
</form>  
</center>  
</body>  
</html>
```

#### IV.1.2 EvaluaUsuario.php

Como vamos a usar código PHP, necesitamos indicarle al servidor de donde parte y termina nuestro código, para que lo interprete antes de enviarle la página al usuario.

```
<?php
```

Primero preguntamos si hubo un método POST, si lo hubo, obtenemos los datos que nos dio el usuario en la página anterior.

```
if ($_POST){  
    $usuario=$_POST['usuario'];  
    $password=$_POST['password'];
```

Lo siguiente es verificar que los datos que nos dio el usuario sean correctos y validos para poder acceder al sistema, para esto comparamos los datos del usuario con los existentes en la base de datos.

```
require_once "lib/conexion.php";  
$conexion=conectarse();  
mssql_select_db('DBBODEGAS',$conexion);  
$consulta="SELECT id_admin,tipo_admin,login_admin,  
password_admin FROM ADMINISTRADORES  
WHERE login_admin='$usuario' AND password_admin='$password'";  
$respuesta=mssql_query($consulta,$conexion);  
$arreglo=mssql_fetch_array($respuesta,MSSQL_NUM);  
$bandera_encontrado=0;
```

Si el usuario puede acceder al sistema, iniciamos sesión para el usuario y se obtiene su “tipo” el cual nos indica si se trata de un “administrador” o un “responsable de bodega” y se almacena en una variable de sesión

```
if ($arreglo[2]==$usuario && $arreglo[3]==$password)  
{  
    $bandera_encontrado=1;  
    session_start();  
    $_SESSION['tipo']=$arreglo[1];  
    $_SESSION['usuario']=$arreglo[0];  
    $_SESSION['pedido'];  
    $tipo=$_SESSION['tipo'];  
    switch($tipo)  
    {
```

Si el usuario es un administrador se le manda a la página Frame.php la cuál contiene todos los procesos que un administrador puede hacer.

```
case 3: header("Location: Frame.php");
break;
```

Pero si se trata de un responsable de bodega lo mandamos a la página FrameB.php, la cual, al igual que la anterior, contiene procesos propios de un responsable de bodega.

```
case 2: header("Location: FrameB.php");
break;
}}
```

Si los datos que introdujo el usuario no son validos, checamos en la base de datos del sistema si es un una persona nivel 1 el que quiere entrar al sistema.

```
if($bandera_encontrado==0){
$consulta="SELECT id_usuario,login_usuario,password_usuario
FROM USUARIOS WHERE login_usuario='$usuario'
AND password_usuario='$password'";
$respuesta=mssql_query($consulta,$conexion);
$arreglo=mssql_fetch_array($respuesta,MSSQL_NUM);
if ($arreglo[1]==$usuario && $arreglo[2]==$password)
{
$bandera_encontrado=1;
}
```

Si los datos introducidos lo identifican como usuario del sistema, iniciamos sesión, almacenamos en una variable de sesión que se trata de un usuario y lo mandamos a la página FrameU.php para que pueda hacer solo pedidos.

```
session_start();
$_SESSION['tipo']=1;
$_SESSION['usuario']=$arreglo[0];
$_SESSION['pedido'];
header("Location: FrameU.php");
} //cierra if comprobación usuario

} //cierra if de bandera_encontrado
```

Si los datos introducidos no corresponden a un administrador, responsable de bodega o usuario, quiere decir, que es un usuario no autorizado y se le rechaza mandándolo a la página Denegado.php.

```
if($bandera_encontrado==0){
header ("Location: Denegado.php");

} //cierra if donde se manda a denegado

} //Cierra if de que exista el POST
```

?>

### IV.1.3 Frame.php

Esta página divide la pantalla en tres marcos: superior, lateral izquierdo y lateral derecho.

```

<html>
  <head>
    <title>Administradores</title>
  </head>
<frameset rows="20%,*">

```

El marco superior muestra la página título.php que como su nombre lo indica es el título de nuestra pantalla.

```

<frame src="título.php" noresize name="top">
  <frameset cols="20%,*">

```

En el marco lateral izquierdo se carga la página Administradores.php que nos muestra un menú que contiene procesos que solo un administrador puede realizar.

```

    <frame src="Administradores.php" noresize name="left">

```

Por último el marco lateral derecho, este marco ocupa la mayor parte de la pantalla por lo que en ella se muestra todo lo que el administrador necesite.

```

    <frame src="Bienvenida.php" noresize name="main">
  </frameset>
</frameset>
</html>

```

#### IV.1.3.1 Administradores.php

Antes de mostrar el contenido de esta página verificamos que la persona que quiere verla sea realmente un administrador para ello preguntamos por medio del if dos cosas: si se inició una sesión y si el usuario es tipo 3 (administrador).

```

<?php
session_start();
    if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==3)
    {

```

Una vez que se comprobó que es un administrador, le mostramos una serie de botones (menú), los cuales al ser presionados mandan al usuario a la página que realiza el proceso indicado en el botón seleccionado.

```

?>
    <html>
    <head>
    <title>Administrador</title>
    </head>
    <body background="./imagenes/fondop2.gif">
    <center>
    <h3>
    <FONT color="White">Administración</FONT></h3><br><br>

```

Si el administrador quiere realizar una entrada de bienes se le manda a Entradas0.php.

```
<a href="Entradas0.php" target="main">
<IMG src="./imagenes/boton1.jpg" border="0"></a><br>
```

Para realizar un proceso de autorización de pedidos se pulsa este botón y se le manda a la página Autorizapedido.php, que es la encargada de dicho proceso.

```
<a href="Autorizapedido.php" target="main">
<IMG src="./imagenes/boton2.jpg" border="0"></a><br>
```

Si lo que se requiere es administrar las cuentas de usuario, este botón lo envía a la página Administrausuarios.php, la cual realiza esta función.

```
<a href="Administrausuarios.php" target="main">
<IMG src="./imagenes/boton5.jpg" border="0"></a><br>
```

Por último tenemos el botón de reportes el cual nos envía a la página Reportes.php para realizar el reporte deseado.

```
<a href="Reportes.php" target="main">
<IMG src="./imagenes/boton6.jpg" border="0"></a>

</center>
</body>
</html>

<?php
}
```

Si no se inició sesión y el usuario no es un administrador, se le rechaza y es enviado a la página Denegado.php.

```
else
{
header("Location: Denegado.php");
}
?>
```

#### IV.1.3.2 Entradas0.php

Lo especial de esta página es que genera el folio de la entrada, el formato del folio es E+año+consecutivo, el consecutivo se reinicia en uno si se detecta un nuevo año.

```
<?
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==3)
{
require_once "lib/conexion.php";
$conexion=conectarse(); //llamado a la función conexion
mysql_select_db('DBBODEGAS',$conexion);//selección de la base de datos
```

Consultamos el folio anterior, que se encuentra en la tabla FOLIOE, para generar el nuevo folio.

```
$consultafolio="SELECT ultimo_folio FROM FOLIOE WHERE id_folio='1'";
$respuestafolio=mssql_query($consultafolio,$conexion);//devuelve la
respuesta de la consulta

$arreglofolio=mssql_fetch_array($respuestafolio,MSSQL_NUM);
$año="";

$folio=$arreglofolio[0];
for($ped=1;$ped<5;$ped++)
{
    $año=$año .$folio[$ped];
}
settype ($año,"integer");
$añoactual=date(Y);
settype ($añoactual,"integer");
```

Si el año que marca el último folio registrado es anterior al año actual se usa el año actual y el consecutivo se inicializa en 001.

```
if($añoactual>$año )
{
    $folio_ausar="E" .date(Y) ."001";
}
```

En caso contrario al anterior, se obtiene el consecutivo anterior y se incrementa en uno.

```
else{
    $consecutivo="";
    for($ped=5;$ped<8;$ped++)
    {
        $consecutivo=$consecutivo .$folio[$ped];
    }
    settype ($consecutivo,"integer");
    $consecutivo++;

    if($consecutivo<10)
    {
        $folio_ausar="00" .$consecutivo;
        settype($folio_ausar,"string");
    }
    elseif($consecutivo>9&&$consecutivo<100)
    {
        $folio_ausar="0" .$consecutivo;
        settype($folio_ausar,"string");
    }
    else
    {
        $folio_ausar=$consecutivo;
        settype($folio_ausar,"string");
    }
    $folio_ausar="E" .date(Y) .$folio_ausar;
}
```



?>

Aquí le mostramos al usuario el folio generado para la entrada que desea realizar.

```
<HTML>
<HEAD><TITLE>Entrada de bienes</TITLE></HEAD>
<BODY background="./imagenes/fondop3.gif">
<h1 align="center"><FONT color="White">Va a realizar una entrada de
bienes</FONT></h1><br>
<p align="justify"><FONT size="6" style="Arial" color="White"><FONT
color="White">Los bienes se agregaran directamente a la base de datos por
eso se le pide un especial cuidado ya que una vez insertado el bien ya no
se puede modificar

El folio de la entrada que usted va a realizar es:</p><br>
<center>
<b><?php echo "$folio_ausar"; ?></FONT></b>
  <form action="./Entradas.php" method="POST">
    <?php echo "<input type='hidden' name='folioe'
value='$folio_ausar'>"; ?><br>
    <input type="submit" value="CONTINUAR">
  </center>
</form>
</BODY>
</HTML>
<?php
}
else
{
    header("Location: ./Denegado.php");
}
```

#### IV.1.3.2.1 Entradas.php

Esta página obtiene el bien y todos sus datos necesarios para poder agregarlo a la entrada de bienes que se esta realizando.

```
<?php
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==3)
{//si es un usuario que inicio sesión y además es el administrador se le
muestra la pagina sino se le rechaza
if ($_POST){
    $folio=$_POST['folioe'];
}

if ($_GET){
    $tipo=$_GET['tipo'];
    $folio=$_GET['folioe'];
}

else{
    $tipo=1;
}
?>
```

```
<html>
<head>
<title>Entrada de bienes</title>
```

Este script evalúa que los datos numéricos no contengan valores alfanuméricos o no válidos como el signo “-”, si encuentra algo no válido manda un mensaje de error y vacía la caja de texto que contiene la información no válida.

```
<SCRIPT language="JavaScript">
    function valida(cadena)
    {
        var x;

        for (x=0;x<cadena.length;x++)
        {
            if(!(cadena.charAt(x)=="1"||cadena.charAt(x)=="2"||cadena.charAt(x)=="3" |
|cadena.charAt(x)=="4"||cadena.charAt(x)=="5"||cadena.charAt(x)=="6"||cad
ena.charAt(x)=="7"||cadena.charAt(x)=="8"||cadena.charAt(x)=="9"||cadena.
charAt(x)=="0"||cadena.charAt(x)=="."))
            {
                alert("NUMERO INVALIDO favor de corregirlo");
                document.datos.precio.value="";
                break;
            }
        }
    }

    function validac(cadena)
    {
        var y;
        for (y=0;y<cadena.length;y++)
        {
            if(!(cadena.charAt(y)=="1"||cadena.charAt(y)=="2"||cadena.charAt(y)=="3" |
|cadena.charAt(y)=="4"||cadena.charAt(y)=="5"||cadena.charAt(y)=="6"||cad
ena.charAt(y)=="7"||cadena.charAt(y)=="8"||cadena.charAt(y)=="9"||cadena.
charAt(y)=="0"||cadena.charAt(y)=="."))
            {
                alert("NUMERO INVALIDO favor de corregirlo");
                document.datos.cantidad.value="";
                break;
            }
        }
    }
</SCRIPT>
```

```
</head>
<body background="./imagenes/fondop3.gif">
<center>
<h2>Introduzca los valores del bien</h2><br>
```

```
<CENTER><b>Primero introduzca el tipo de CAMB</b></center><br><br>
TIPO CAMB:
```

```
<form name="tipoCABM" action="" method="GET" >
<input type='hidden' name='folio' value="<?php echo $folio; ?>">
```

Aquí cargamos una lista con todos los tipos de CABM que hay, al momento de que el usuario selecciona uno, esta página se autollama y carga todos los CABM en la lista inferior que son del tipo seleccionado.

```
<SELECT name="tipo"
onchange="document.location.href=((this.form.tipo.options[this.form.tipo.
selectedIndex].title)+'?&tipo=')+(this.form.tipo.options[this.form.tipo.
selectedIndex].value)+'&folioe=')+(this.form.folio.value))">
    <?php

        require_once "lib/conexion.php"; //indica que
        requeriremos de esta librería
        $conexion=conectarse();
        mssql_select_db('DBBODEGAS', $conexion); //selección
        de la base de datos
        $consulta="SELECT * FROM TIPO_CABM ORDER BY
        descrip_tipo_cabm";
        $respuesta=mssql_query($consulta, $conexion);
        //devuelve la respuesta de la consulta
        while($arreglo=mssql_fetch_array($respuesta, MSSQL_NUM))
        {
            if ($tipo==$arreglo[0])
            {
                echo "<option value=' " . $arreglo[0] . "' title='./Entradas.php'
                selected>";
                echo "$arreglo[1]";
                echo "</option>";
            }
            else
            {
                echo "<option value=' " . $arreglo[0] . "' title='./Entradas.php'>";
                echo "$arreglo[1]";
                echo "</option>";
            }
        }
    ?>
</SELECT><br>
</form>
<FORM action="./Entradas2.php" name="datos">
CABM:
    <SELECT name="cabm">
    <?php
        require_once "lib/conexion.php"; //indica que requeriremos
        de esta librería
        $conexion=conectarse(); //llamado a la función conexion
        mssql_select_db('DBBODEGAS', $conexion); //selección de la
        base de datos
```

Esta es la lista que carga todos los CABM de cierto tipo seleccionado por el usuario.

```
$consulta="SELECT * FROM CATALOGO_CABM WHERE tipo_cabm='$tipo' ORDER BY
clave_cabm";
```

```

$respuesta=mssql_query($consulta,$conexion); //devuelve la respuesta de
la consulta

while($arreglo=mssql_fetch_array($respuesta,MSSQL_NUM))
{
    echo "<option value='" . $arreglo[0] . "'>";
    echo "$arreglo[1] $arreglo[4]";
    echo "</option>";
}
?>
</SELECT>
<br><br>
<hr>

```

En esta última parte de código se pide al usuario introduzca los datos correspondientes al bien que esta entrando, al final aparecen dos botones: resetear y agregar a la entrada. El de botón “resetear” sirve para dejar en blanco todas las cajas de texto y el botón “agregar a la entrada” sirve para agregar el bien a nuestra entrada de bienes, cuando se ha hecho esto se le pregunta al usuario si quiere seguir introduciendo bienes a su entrada, si su respuesta es afirmativa se vuelve a esta página e introduce el bien, si no, la entrada se da por finalizada y el folio de ésta se guarda en la base de datos como el último folio usado, que para la próxima entrada, se genere otro número de folio consecutivo a éste.

```

<center><b>Ahora introduzca los siguientes datos:</b></center><br>
    Nombre:
<input type="text" name="nombre" value="" maxlength="50"><br><br>
    Descripción:
<input type="text" name="descripcion" value="" maxlength="50"><br><br>
    Unidad:
<input type="text" name="unidad" value="" maxlength="50">
<br>(piezas,metros,cajas, etc.)<br><br>
    Precio unitario: $
<input type="text" name="precio"
onblur="valida(this.form.precio.value);">
<br>(En pesos)<br><br>
    Cantidad:
<input type="text" name="cantidad"
onkeyup="valida(this.form.cantidad.value);">
<br><br>
<?php echo "<input type='hidden' name='folioe' value='$folio'";?>
<input type="reset" value="Resetear">
<input type="submit" name="enviar" value="Agregar a la entrada">
</center>
</FORM>
</body>
</html>

<?php
} //finaliza if de si se inicio sesión y el usuario si es valido
else
{
header("Location: Denegado.php");//NOS MANDA A LA PAGINA
DENEGADO
}
?>

```

### IV.1.3.3 Autorizapedido.php

Esta página carga todos los pedidos que han sido solicitados, añadiéndoles al final de la fila un botón “Ver”, para que el administrador vea los detalles del pedido y así evalúe si autoriza o no, éste.

Una vez mostrado el pedido en Autorizapedido2.php se le muestran dos botones: Autorizar y Rechazar, que nos mandan a Autorizapedido3.php. Si se autoriza el pedido se modifica su status a 2, esto significa que se autorizó pero no ha sido entregado físicamente todavía, pero si se rechaza se agregan los bienes solicitados a las existencias que están registradas en la base de datos y el pedido cambia a status 0, que significa cancelado.

```
<?php
    session_start();
    require_once "lib/conexion.php";
    $conexion=conectarse();
    mssql_select_db('DBBODEGAS',$conexion);
    //selección de la base de datos

    $consulta="SELECT * FROM PEDIDOS WHERE status_pedido=1 ORDER BY
    folio_pedido ASC";
    $respuesta=mssql_query($consulta,$conexion);

?>
<HTML>
<body background="./imagenes/fondop3.gif">
<h1 align="center"><font color="White">Pedidos realizados</font></h1><br>
<TABLE bgcolor="White" border="1" cellpadding="2">
    <TR>
        <TD><b>Folio Pedido</b></TD>
        <TD><b>Fecha Pedido</b></TD>
        <TD><b>Usuario que solicita</b></TD>
        <TD><b>Nacionalidad</b></TD>
        <TD><b>Status</b></TD>
    </TR>

<?php
while($arreglo=mssql_fetch_array($respuesta,MSSQL_NUM))//CON ESTE WHILE
SE CARGA TODO LO QUE HAY EN LA TABLA BIENES
    {
        echo "<tr>";
        echo "<td>$arreglo[0]</td>";
        echo "<td>$arreglo[1]</td>";

        $consulta5="SELECT nom_usuario,ap_pat_usuario,
        ap_mat_usuario FROM USUARIOS
        WHERE id_usuario=$arreglo[4]";
        $respuesta5=mssql_query($consulta5,$conexion);
        //devuelve la respuesta de la consulta

        $arreglo5=mssql_fetch_array($respuesta5,MSSQL_NUM);

        echo "<TD>$arreglo5[1] $arreglo5[2]
```

```

$arreglo5[3]</TD>";

        echo "<td>$arreglo[5]</td>";
        echo "<td>$arreglo[6]</td>";
        ?>
        <td>
        <FORM action="Autorizapedido2.php"
        method="GET">
        <input type="submit" value="VER">
        </td>
        <?
        echo "<input type='hidden' name='folio'
        value='$arreglo[0]'">";
        echo "<input type='hidden' name='usuario'
        value='$arreglo[4]'">";
        echo "</FORM>";
        echo "</tr>";
    }
?>
</TABLE>
</body>
</HTML>

```

#### IV.1.3.4 Administrausuarios.php

Esta página nos muestra dos listas a modo de que en una aparecen las operaciones que se pueden realizar (alta, baja y modificación) y otra que muestra el tipo de cuentas que hay (Administrador, Responsable de bodega y Usuario). Una vez seleccionada la operación, se le manda al usuario a través de las páginas Administrausuarios2.php, Administrausuarios3.php y Administrausuarios4.php para que pueda realizar la operación seleccionada.

```

<?php
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==3)
{ //si es un usuario que inicio sesión y además es el
administrador se le muestra la pagina sino se le rechaza
?>
    <html>
    <head>
    <title>Administrador</title>
    </head>
    <body background=" ./imagenes/fondop3.gif">

    <center>
    <h3><FONT color="White">
Administración Usuarios</FONT></h3><br><br>
    <h3><FONT color="White">
Seleccione la acción a realizar y el tipo de usuario
</FONT></h3><br><br>

    <form action=" ./Administrausuarios2.php"
method="GET">
    <select name="accion">
        <option value="1">Agregar</option>

```



```

INNER JOIN CATALOGO_CABM ON
BIENES.cabm_bien=CATALOGO_CABM.id_cabm";

$respuesta1=mssql_query($consulta1,$conexion);

$arreglo1=mssql_fetch_array($respuesta1,MSSQL_NUM);
?>
<body background="./imagenes/fondop3.gif">
<center><h1><FONT color="White">
Consulta completa de existencias</FONT></h1></center>
<br>
<table bgcolor="White" align="center" width="650" border="1">
<TR>
<TD><b>Nombre Bien</b></TD>
<TD><b>CABM Bien</b></TD>
<TD><b>Descripción Bien</b></TD>
<TD><b>Unidad Bien</b></TD>
<TD><b>Precio Unitario Bien</b></TD>
<TD><b>Cantidad Existente Bien</b></TD>
</TR>
<?php

while($arreglo1=mssql_fetch_array($respuesta1,MSSQL_NUM))
{
?>
<tr>
<TD><?echo $arreglo1[1]?></TD>
<TD><?echo $arreglo1[7]?></TD>
<TD><?echo $arreglo1[3]?></TD>
<TD><?echo $arreglo1[4]?></TD>
<TD><?echo $arreglo1[5]?></TD>
<TD><?echo $arreglo1[6]?></TD>
</tr>

<?php
}
?>

</table>
</body>
<?php
break;

```

Si recibimos un valor 2, quiere decir que debemos de mostrarle el reporte "Consulta mensual de salidas por coordinación", por lo que primero le pedimos el mes, el año y la coordinación para después mostrar dicho reporte.

```

case 2:
?>
<html>
<body background="./imagenes/fondop3.gif">
<center><h1><FONT color="White">
Consulta mensual de salidas por coordinación</FONT>
</h1></center>

<center><h2><FONT color="White">

```



```

Selecciona el mes, el año y la coordinación</FONT></h2></center>
<h3><font color="White">Mes:</font></h3>
<form action="Reportes2.php" method="GET">

<select name="mes">
<option value="1">Enero</option>
<option value="2">Febrero</option>
<option value="3">Marzo</option>
<option value="4">Abril</option>
<option value="5">Mayo</option>
<option value="6">Junio</option>
<option value="7">Julio</option>
<option value="8">Agosto</option>
<option value="9">Septiembre</option>
<option value="10">Octubre</option>
<option value="11">Noviembre</option>
<option value="12">Diciembre</option>
</select><br>

<h3><font color="White">Año:</font></h3>
<INPUT type="text" name="año" maxlength="4"><br>

<h3><font color="White">Coordinación:</font></h3>
<select name="coordinacion">
<?
$consulta2="SELECT * FROM COORDINACIONES";

$respuesta2=mssql_query($consulta2,$conexion);
while ($arreglo2=mssql_fetch_array($respuesta2,MSSQL_NUM))
{
echo "<option value='\$arreglo2[0]'\>\$arreglo2[1]</option>";
}
?>
</select><br><br>
<center>
<input type="submit" name="consulta" value="Consultar">
</center>
</form>
</body>
</html>
<?php
break;

```

Con un valor 3 mostramos un reporte mensual de salidas, donde primero se le pide el mes y el año y se le muestran las salidas del mes y año requeridos.

```

case 3:
?>
<html>
<body background="./imagenes/fondop3.gif">
<center><h1><FONT color="White">
Consulta mensual de salidas</FONT></h1></center>
<center><h2><FONT color="White">
Selecciona el mes y el año</FONT></h2></center>
<h3><font color="White">Mes:</font></h3>
<form action="Reportes3.php" method="GET">

```

```

<select name="mes">
<option value="1">Enero</option>
<option value="2">Febrero</option>
<option value="3">Marzo</option>
<option value="4">Abril</option>
<option value="5">Mayo</option>
<option value="6">Junio</option>
<option value="7">Julio</option>
<option value="8">Agosto</option>
<option value="9">Septiembre</option>
<option value="10">Octubre</option>
<option value="11">Noviembre</option>
<option value="12">Diciembre</option>
</select><br>

<h3><font color="White">Año:</font></h3>
<INPUT type="text" name="año" maxlength="4"><br>

<center>
<input type="submit" name="consulta" value="Consultar">
</center>

</form>
</body>
</html>
<?php
break;

```

Con un valor 4 le mostramos una consulta anual de salidas pidiendo antes el año que se desea ver.

```

case 4:
?>
<html>
<body background="./imagenes/fondop3.gif">
<center><h1><FONT color="White">
Consulta anual de salidas</FONT></h1></center>
<center><h2><FONT color="White">
Introduce el año</FONT></h2></center>

<form action="Reportes4.php" method="GET">

<h3><font color="White">Año:</font></h3>
<INPUT type="text" name="año" maxlength="4"><br>

<center>
<input type="submit" name="consulta" value="Consultar">
</center>

</form>
</body>
</html>
<?php
break;

```

Por último tenemos que si el valor es 5, entonces, le mostramos un reporte sobre todas las entradas que hubo en un año de interés, y como en las anteriores, primero se le pide dicho año.

```
case 5:

?>
<html>
<body background="./imagenes/fondop3.gif">
<center><h1><FONT color="White">
Consulta anual de entradas</FONT></h1></center>
<center><h2><FONT color="White">
Introduce el año</FONT></h2></center>

<form action="Reportes5.php" method="GET">

<h3><font color="White">Año:</font></h3>
<INPUT type="text" name="año" maxlength="4"><br>

<center>
<input type="submit" name="consulta" value="Consultar">
</center>

</form>
</body>
</html>
<?php

break;
}
}
```

Si esta página no fue invocada por un evento GET, entonces, le mostramos al usuario un menú de “tipos de reportes”, para que éste, seleccione el que necesita ver.

```
else
{
?>

<HTML>
<body background="./imagenes/fondop3.gif" alink="Blue"
link="Blue" vlink="Blue">
<center><h1><FONT color="White">
Selecciona que tipo de reporte requieres</font></h1></center>
<table bgcolor="White" align="center">
<TR>
<TD><a href="Reportes.php?&valor=1">
Consulta completa de existencias</a></TD>
</tr>
<tr>
<TD><a href="Reportes.php?&valor=2">
Consulta mensual de salidas por coordinación</a></TD>
</tr>
<tr>
<TD><a href="Reportes.php?&valor=3">
Consulta mensual de salidas</a></TD>
```

```

        </tr>
        <tr>
        <TD><a href="Reportes.php?&valor=4">
        Consulta anual de salidas</a></TD>
        </tr>
        <tr>
        <TD><a href="Reportes.php?&valor=5">
        Consulta anual de entradas</a></TD>
        </TR>
    </table>

    </body>
</HTML>
<?
} //cierra else de get
}
else
{
    header("Location: Denegado.php");
} //cierra else de verificación de tipo de usuario
?>

```

#### IV.1.4 FrameB.php

Esta página funciona de la misma forma que Frame.php, solo que ésta, carga en el marco lateral izquierdo las funciones de un responsable de bodega

```

<html>
  <head>
    <title>Bodega</title>
  </head>
  <frameset rows="20%,*">
    <frame src="titulo.php" noresize name="top">
      <frameset cols="20%,*">
        <frame src="Bodegas.php" noresize name="left">
        <frame src="BienvenidaB.php" noresize name="main">
      </frameset>
    </frameset>
  </html>

```

##### IV.1.4.1 Bodegas.php

Aquí sólo se muestra un solo botón (registrar salida), ya que el responsable de bodega solo se encarga de entregar físicamente un pedido de bienes y de registrar en el sistema que el pedido ha sido entregado.

Como en las páginas anteriores se inicia sesión, se verifica si el usuario es un responsable de bodega, y si lo es, se le muestra el botón para registrar una salida.

```

<?php
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==2)
{

```

```
?>
<html>
<head>
<title>Bodega</title>
</head>
<body background="./imagenes/fondop2.gif">
<center>
<h3><FONT color="White">Bodega</FONT></h3><br><br>
```

Este botón nos manda a la página Autorizasalida.php para registrar de entregado un pedido.

```
<a href="Autorizasalida.php" target="main">
<IMG src="./imagenes/boton1_rs.jpg" border="0"></a><br>
</center>
</body>
</html>
<?php
}
```

Como siempre si el usuario no es válido para ver el contenido de esta página se le rechaza y es enviado a la página Denegado.php.

```
else
{
header("Location: Denegado.php");
}
?>
```

#### IV.1.4.2 Autorizasalida.php

Iniciamos sesión y checamos que la persona sea un responsable de bodega.

```
<?php
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==2)
{
```

Incluimos conexion.php porque en ella yace la función conectarse que se utiliza más abajo, esta función sirve para establecer una conexión con el servidor de bases de datos.

```
require_once "lib/conexion.php";
$conexion=conectarse();
```

Seleccionamos la base de datos de nuestro sistema.

```
mssql_select_db('DBBODEGAS', $conexion);
```

Al responsable de bodega se le muestran todos los pedidos autorizados no entregados para que pueda ver el que necesita entregar.

```
$consulta="SELECT * FROM PEDIDOS WHERE status_pedido=2
ORDER BY folio_pedido ASC";
$respuesta=mssql_query($consulta,$conexion);
```

```

?>
<HTML>
<body background="./imagenes/fondop3.gif">
<h1 align="center"><font color="White">Pedidos realizados</font></h1><br>
<TABLE bgcolor="White" border="1" cellpadding="2">
  <TR>
    <TD><b>Folio Pedido</b></TD>
    <TD><b>Fecha Pedido</b></TD>
    <TD><b>Usuario que solicita</b></TD>
    <TD><b>Nacionalidad</b></TD>
    <TD><b>Status</b></TD>
  </TR>

<?php
while($arreglo=mssql_fetch_array($respuesta,MSSQL_NUM))//CON ESTE WHILE
SE CARGA TODO LO QUE HAY EN LA TABLA BIENES
{
    echo "<tr>";
    echo "<td>$arreglo[0]</td>";
    echo "<td>$arreglo[1]</td>";

    $consulta5="SELECT nom_usuario,ap_pat_usuario,
ap_mat_usuario FROM USUARIOS
WHERE id_usuario=$arreglo[4]";
$respuesta5=mssql_query($consulta5,$conexion);
$arreglo5=mssql_fetch_array($respuesta5,MSSQL_NUM);

    echo "<td>$arreglo5[1] $arreglo5[2] $arreglo5[3]</td>";

    echo "<td>$arreglo[5]</td>";
    echo "<td>$arreglo[6]</td>";
    ?>
    <td>

```

Para que el responsable de bodega vea a detalle lo que debe entregar, se le agrega un botón que dice “VER” a cada pedido no entregado, éste al ser presionado, muestra la página Autorizasalida2.php que contiene los bienes que deben de ser entregados.

```

<FORM action="Autorizasalida2.php" method="GET">
<input type="submit" value="VER">
</td>
<?

```

A la página Autorizasalida2.php se le envía el folio del pedido y la persona que hizo el pedido para poder mostrar los bienes a entregar del pedido.

```

    echo "<input type='hidden' name='folio'
value='$arreglo[0]'">";
    echo "<input type='hidden' name='usuario'
value='$arreglo[4]'">";
    echo "</FORM>";
    echo "</tr>";
}

```

```
?>
        </TABLE>
</body>
</HTML>
<?php
    }//cierra if que pregunta si es un usuario valido
```

Si es una persona no autorizada se le rechaza.

```
    else
    {
        header("Location: Denegado.php");
    }

?>
```

#### IV.1.4.3 Autorizasalida2.php

Esta página muestra los detalles del pedido seleccionado, al final de esta página, aparece el botón “Registrar salida”, que al oprimirlo se cambia el estatus del pedido de “autorizado no entregado” a “autorizado y entregado”.

```
<?php

    session_start();
    if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==2)
    {

        require_once "lib/conexion.php";
        $conexion=conectarse();
        mssql_select_db('DBBODEGAS', $conexion);

        $folio= $_GET["folio"];
        $usuario= $_GET["usuario"];

        $consulta5="SELECT nom_usuario,ap_pat_usuario,ap_mat_usuario,
        coord_usuario FROM USUARIOS WHERE id_usuario=$usuario";
        $respuesta5=mssql_query($consulta5,$conexion); //devuelve la
        respuesta de la consulta
        $arreglo5=mssql_fetch_array($respuesta5,MSSQL_NUM);
        $consulta5a="SELECT descrip_coordina FROM COORDINACIONES
        WHERE id_coordina=$arreglo5[3]";
        $respuesta5a=mssql_query($consulta5a,$conexion);
        $arreglo5a=mssql_fetch_array($respuesta5a,MSSQL_NUM);

?>

<BODY background="./imagenes/fondop3.gif">
<center>
<Table bgcolor="White" border="1" width="600">
    <TR>
        <TD><b>Nombre Usuario</b></TD>
        <?php echo "<TD>$arreglo5[0] $arreglo5[1]
        $arreglo5[2]</TD>" ;?>
    </TR>
    <TR>
        <TD><b>Coordinación</b></TD>
        <?php echo "<TD>$arreglo5a[0]</TD>" ;?>
```

```

</TR>
<TR>
    <td><b>No. Folio</b></td>
    <?php echo "<TD>$folio</TD>" ;?>
</TR>

<?
    $consulta1="SELECT id_bien,cant_pedido
    FROM PEDIDOS_BIENES WHERE folio_pedido=$folio";
    $respuesta1=mssql_query($consulta1,$conexion);

    echo "<table bgcolor='White' border='1' width=600>";
        echo "<TR>";
            echo "<td><b>Nombre bien</b></td>";
            echo "<td><b>Descripción</b></td>";
            echo "<td><b>Unidad</b></td>";
            echo "<td><b>Cantidad Requerida</b></td>";
            echo "</TR>";

    while ($arreglo1=mssql_fetch_array($respuesta1,MSSQL_NUM))
        {
            $consulta2="SELECT * FROM BIENES
            WHERE id_bien=$arreglo1[0]";

            $respuesta2=mssql_query($consulta2,$conexion);
            $arreglo2=mssql_fetch_array($respuesta2,
            MSSQL_NUM);

                echo "<TR>";
                    echo "<td>$arreglo2[1]</td>";
                    echo "<td>$arreglo2[3]</td>";
                    echo "<td>$arreglo2[4]</td>";
                    echo "<td>$arreglo1[1]</td>";
                echo "</TR>";
            }
        echo "</table>";

?>

</table>
<br><br>
<FORM action="Autorizasalida3.php" method="POST">
<input type="submit" name="autorizar" value="Registrar salida">
<input type="hidden" name="valor1" value="1">
<?echo "<input type='hidden' name='folio' value='$folio'>" ;?>
</FORM>

</center>

</BODY>
<?php
} //cierra if que evalúa si el usuario es valido
else
{
header("Location: Denegado.php");
}
?>

```



## IV.1.5 FrameU.php

También funciona igual que Frame.php, la diferencia es, que esta carga la página Usuarios.php que tiene un botón para realizar solo pedidos.

```
<html>
  <head>
    <title>PEDIDOS</title>
  </head>
  <frameset rows="20%,*">
    <frame src="titulo.php" noresize name="top">
      <frameset cols="20%,*">
        <frame src="Usuarios.php" noresize name="left">
          <frame src="BienvenidaU.php" noresize name="main">
        </frameset>
      </frameset>
    </frameset>
  </html>
```

### IV.1.5.1 Usuarios.php

Esta página tiene un único botón, Realizar pedido, ya que un usuario sólo puede realizar pedidos.

```
<?php
session_start();
if(isset($_SESSION['tipo'])&&$_SESSION['tipo']==1)
{
  ?>
  <html>
    <head>
      <title>USUARIOS</title>
    </head>
    <body background="./imagenes/fondop2.gif">
      <center>
        <h3><FONT color="White">PEDIDOS</FONT></h3><br><br>
      </center>
    </body>
  </html>
```

Para realizar su pedido se le envía a la página Pedido.php.

```
  <a href="Pedido.php" target="main">
  <IMG src="./imagenes/boton3.jpg" border="0"></a><br>
  </center>
  </body>
</html>
```

```
<?php
}
```

Si no es un usuario autorizado se le rechaza.

```
  else
  {
    header("Location: Denegado.php");
  }
?>
```

### IV.1.5.2 Pedido2.php

Esta página es muy importante ya que aquí se agregan o se quitan bienes al pedido que el usuario esta realizando.

```
<?php
require_once "lib/encuentra.php";
if($_GET)//pregunta si ocurrió el método GET
{
session_start();
?>
<html>
<head>
<title>Pedidos</title>
```

El siguiente script evalúa si la cantidad que se requiere del bien no exceda su existencia y que la cantidad introducida sea numérica, si no es así, aparece un mensaje pidiéndole que corrija el error.

```
<SCRIPT language="JavaScript">
function numero(a,b){
var y;
var oknum;

for (y=0;y<b.length;y++){

if(!(b.charAt(y)=="1"||b.charAt(y)=="2"||b.charAt(y)=="3"||b.charAt(y)=="
4"||b.charAt(y)=="5"||b.charAt(y)=="6"||b.charAt(y)=="7"||b.charAt(y)=="8
"||b.charAt(y)=="9"||b.charAt(y)=="0"||b.charAt(y)=="."))
{
    alert("NUMERO INVALIDO favor de corregirlo");
    oknum=1;
} //cierra if
} //cierra for

b2=parseInt(b);
a2=parseInt(a);
if(b2>a2){
    alert("las unidades que solicitas exceden a las existentes");
} //cierra if

} //cierra función principal
</SCRIPT>

</head>
<body background="./imagenes/fondop3.gif">
<center>
<h2><FONT color="White">Pedido de bienes</FONT></h2><br><br>
<?php
require_once "lib/conexion.php";//indica que requeriremos de esta
librería

$conexion=conectarse(); //llamado a la función conexion
mysql_select_db('DBBODEGAS',$conexion); //selección de la base de datos
```

```

$consulta="SELECT * FROM BIENES WHERE cant_exist>0 ORDER BY nom_bien
ASC";
$respuesta=mssql_query($consulta,$conexion); //devuelve la respuesta de
la consulta

$operacion=$_GET['operacion'];
if ($operacion==1)
{
    $contador=$_GET['contador'];
    $id_bien=$_GET['id_bien'];
    $uni_pedi=$_GET['uni_pedi'];

    $pedido=$_SESSION['pedido'];
    $pedido[$contador][1]=$id_bien;
    $pedido[$contador][2]=$uni_pedi;

    $_SESSION['pedido']=$pedido;
} //cierra if de operación

echo '<table border="1" bgcolor="White">';
echo '<tr>'; //ESTA FILA SOLO PINTA EL PRIMER RENGLON DE LA TABLA
echo '<td>Bien</td>';
echo '<td>Descripción</td>';
echo '<td>Unidad</td>';
echo '<td>Existencia</td>';
echo '<td>Introduzca el número de unidades a pedir</td>';
echo '<td>Esto ha solicitado</td>';
echo '</tr>';

if ($operacion==1)
{
    $contador=$contador+1;
}

```

Aquí cargamos una tabla que muestra todos los bienes que hay disponibles.

```

while($arreglo=mssql_fetch_array($respuesta,MSSQL_NUM)) //CON ESTE WHILE
SE CARGA TODO LO QUE HAY EN LA TABLA BIENES
{
echo "<tr>";
echo "<td>$arreglo[1]</td>";
echo "<td>$arreglo[3]</td>";
echo "<td>$arreglo[4]</td>";
echo "<td>$arreglo[6]</td>";
}

```

Con esto, al final de cada fila de la tabla, se carga un formulario que contiene datos necesarios del bien para agregarlo al pedido, también, una caja de texto para que el usuario introduzca la cantidad que necesita y un botón para agregar el bien a su pedido.

```

echo "<td><form action='Pedidovacio.php' method='GET'>";
echo "<input type='hidden' name='existencia' value='$arreglo[6]'>";
?>

<input type='text' name='uni_pedi'
onkeyup="numero(this.form.existencia.value,this.form.uni_pedi.value)"><br
>

```

```

<?php
echo "<input type='hidden' name='pagina' value='2'>";
echo "<input type='hidden' name='id_bien' value='\$arreglo[0]'>";
echo "<input type='hidden' name='contador' value='\$contador'>";
echo "<input type='hidden' name='operacion' value='1'>";
echo "<input type='submit' value='Agregar al pedido'></form></td>";

```

La función encontrar (definida en encuentra.php), agrega el botón “Quitar del pedido” a los bienes que el usuario ya agregó a su pedido, para que el usuario en caso de arrepentimiento quite el bien de su pedido.

```

encontrar(\$arreglo[0]);

echo "</tr>";
} //cierra while

echo '</table>';

```

Al final, agregamos el botón “Realizar pedido” que manda al usuario a la siguiente página que le muestra su pedido y le pide su nacionalidad para que por fin pueda finalizar su pedido.

```

echo "<form action='Pedido3.php' method='POST'>";
echo "<input type='submit' value='Realizar pedido'>";
echo '</form>';
echo '</center>';
echo '</body>';
echo '</html>';

} //cierra if de GET
?>

```

## IV.2 Pruebas

Se probó el sistema completo, realizando todos los procesos de éste e introduciendo todo tipo de valores donde se pide se introduzcan éstos. Al final de las pruebas se obtuvo lo siguiente:

1. En lugares donde el usuario solo puede introducir valores numéricos, se podían poner también caracteres alfanuméricos, provocando errores al insertar los datos a la base de datos.
2. El usuario podía introducir cantidades negativas.
3. El usuario podía pedir un número de bienes mayor al de las existencias.
4. En campos donde la base de datos solo acepta determinado número de carácter el usuario podía introducir los caracteres que quisiera.
5. El usuario podía poner años como 20789981.

Éstos fueron los problemas detectados y que fueron corregidos.

### **IV.3 Implantación**

El sistema se implantó en paralelo ya que el nuevo y el viejo sistema funcionaron durante un periodo al mismo tiempo.

Primero se instaló todo el equipo necesario, después todo el software necesario. Una vez instalado el software y hardware se habilitó el servidor de páginas Web junto con el servidor de bases de datos de nuestro sistema.

Se llevaron a cabo pruebas de conexión entre los equipos que usaran los usuarios y el servidor de páginas Web, y del servidor de páginas Web hacia el servidor de bases de datos.

Se probó que el sistema estuviese corriendo de forma correcta, para después hacer un simulacro con todos los usuarios involucrados.

Al final entró en función el nuevo sistema en paralelo con el viejo, esperando que después de un mes sólo se trabaje con el nuevo.

Ya solo nos queda dar los manuales de nuestro sistema para cualquier duda, para eso pasaremos al siguiente capítulo.

## Capítulo V Manual de Usuario y Administrador

El sistema cuenta con tres tipos de manual:

- a) Manual de Responsable de Bodega
- b) Manual de Usuario
- c) Manual de Administrador

Manual de Responsable de Bodega: Éste nos enseña a manejar la parte del sistema que se encarga de registrar las salidas físicas de los bienes.

Manual de Usuario: Va dirigido a los usuarios que desean saber como hacer pedidos de bienes.

Manual de administrador: Aquí se describe como dar de alta a un usuario en nuestro sistema, como autorizar un pedido y a realizar una entrada de bienes entre otras cosas.

Cada manual va acompañado de un anexo, donde se muestran las ventanas que van a aparecer en nuestro explorador, cuando realicemos algún proceso descrito en éste.

El formato de los manuales se eligió así, para dar hincapié a los procesos de cada manual.

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT
	Manual de Responsable de bodega

Fecha	26-II-2007		
Página	1	de	1
Sustituye a			
Fecha			
Página			

REGISTRO DE LA SALIDA DE UN PEDIDO					
No. Operación	Responsables	Descripción			
1	Responsable de bodega	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.			
2	Responsable de bodega	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.			
3	Responsable de bodega	Hacer clic sobre el botón “Registrar salida” (ver Fig. 2 anexo A). Aparecerá un listado con todos los pedidos autorizados no entregados.			
4	Responsable de bodega	Hacer clic sobre el botón “Ver” (ver Fig. 3 anexo A) del pedido que se quiera registrar su salida. Aparecerá una página con todos los detalles del pedido (ver Fig. 4 anexo A).			
5	Responsable de bodega	Deberá entregar solo los bienes descritos en el pedido.			
6	Responsable de bodega	Al final de la entrega se deberá oprimir el botón “Registrar salida” para registrar en el sistema que el pedido ha sido entregado (ver Fig. 5 anexo A).			
<table border="1" style="width: 100%;"> <tr> <td style="width: 33%;">Elaboró: Christian Padilla</td> <td style="width: 33%;">Revisó</td> <td style="width: 33%;">Autorizó</td> </tr> </table>			Elaboró: Christian Padilla	Revisó	Autorizó
Elaboró: Christian Padilla	Revisó	Autorizó			

TELECOMM-GCOM-01

## ANEXO A

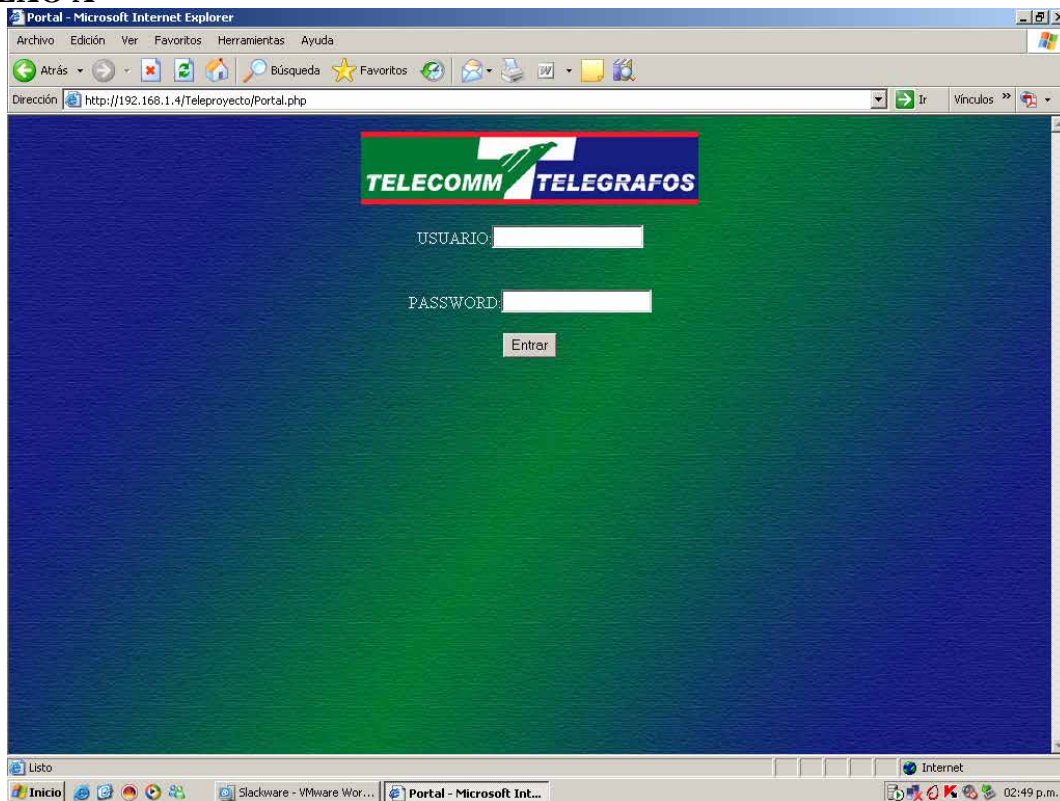


Fig. 1



Fig. 2





Fig. 3

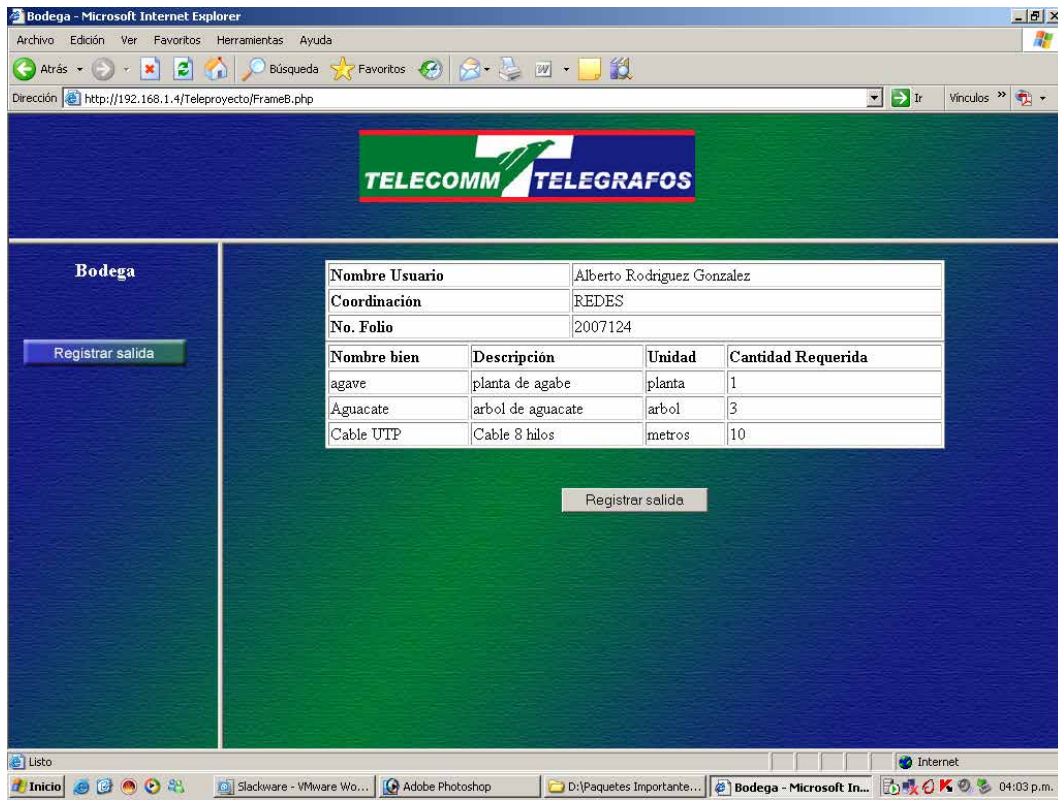



Fig. 4




Fig. 5

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		Fecha	26-II-2007		
	Manual de Usuario		Página	1	de	2
			Sustituye a			
			Fecha			
			Página			

<b>REALIZAR UN PEDIDO</b>					
No. Operación	Responsables	Descripción			
1	Usuario	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.			
2	Usuario	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.			
3	Usuario	Hacer clic sobre el botón “Pedido de bienes” (ver Fig. 2 anexo A) para realizar un pedido. Aparecerá un listado con todos los bienes que están en existencia cada bien aparece con un botón que dice “Agregar al pedido”.			
4	Usuario	Se debe introducir en la caja de texto, del bien que se quiere agregar, la cantidad de unidades que usted requiere. Si introduce valores que no sean numéricos aparecerá un aviso de que está cometiendo un error o si introduce una cantidad mayor a la existente también se le avisará que corrija este error. Usted puede repetir el mismo proceso hasta tener todos los bienes que usted necesita (ver Fig. 3 anexo A).			
5	Usuario	Si por error o por otra causa usted necesita quitar un bien de su pedido deberá presionar el botón “Quitar del pedido” para quitar el bien de su pedido (ver Fig. 3 anexo A).			
6	Usuario	Una vez introducidos los bienes a su pedido deberá pulsar el botón “realizar pedido” (ver Fig. 4 anexo A). Ésto lo mandará a otra página donde se muestra su pedido, la fecha, su folio del pedido entre otras cosas.			
<table border="1" style="width: 100%;"> <tr> <td style="width: 33%;">Elaboró: Christian Padilla</td> <td style="width: 33%;">Revisó</td> <td style="width: 33%;">Autorizó</td> </tr> </table>			Elaboró: Christian Padilla	Revisó	Autorizó
Elaboró: Christian Padilla	Revisó	Autorizó			

**TELECOMM-GCOM-02**

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		
	Manual de Usuario		
Fecha	26-II-2007		
Página	2	de	2
Sustituye a			
Fecha			
Página			

### REALIZAR UN PEDIDO

No. Operación	Responsables	Descripción
7	Usuario	Deberá checar que todos los datos sean correctos y después seleccionar la nacionalidad de su pedido por medio de la lista de nacionalidades (ver Fig. 5 anexo A).
8	Usuario	Una vez seleccionada la nacionalidad se oprime el botón "Solicitar Pedido". Se debe tener cuidado en este paso pues una vez oprimido este botón ya no hay marcha atrás (ver Fig. 5 anexo A).
9	Usuario	Por último, deberá esperar respuesta del administrador de si autoriza o no su pedido (ver Fig. 6 anexo A).
<p>Elaboró: Christian Padilla      Revisó      Autorizó</p>		

**TELECOMM-GCOM-03**



## ANEXO A

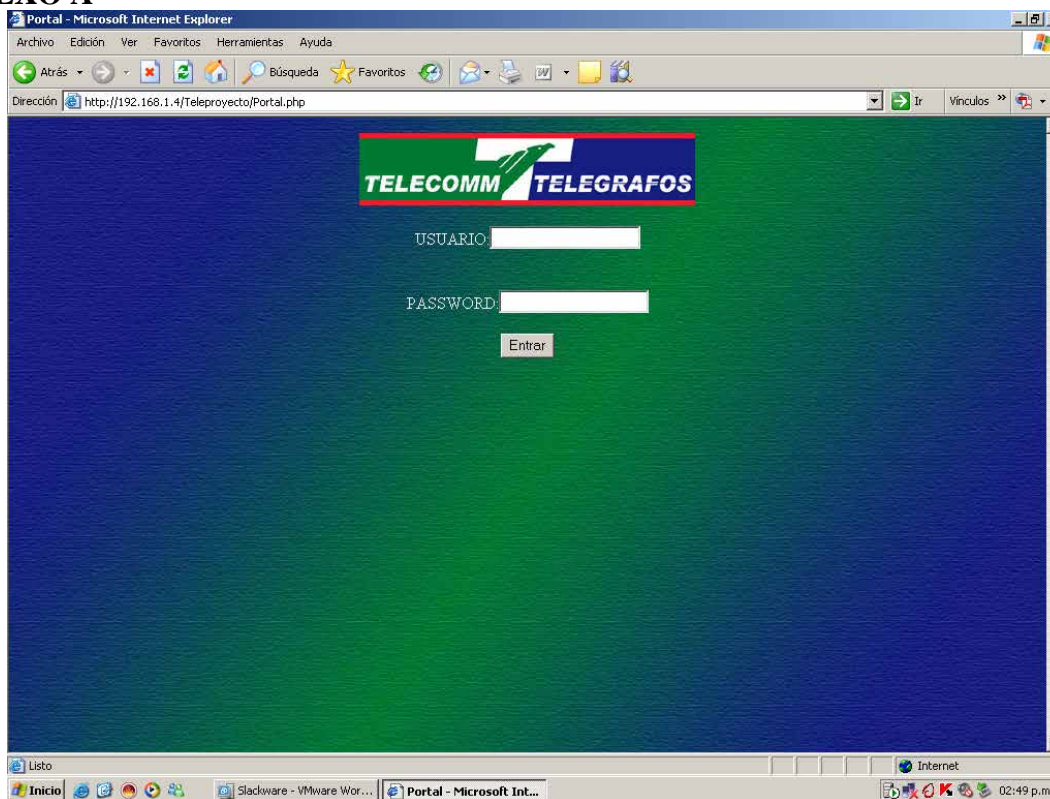


Fig. 1



Fig. 2

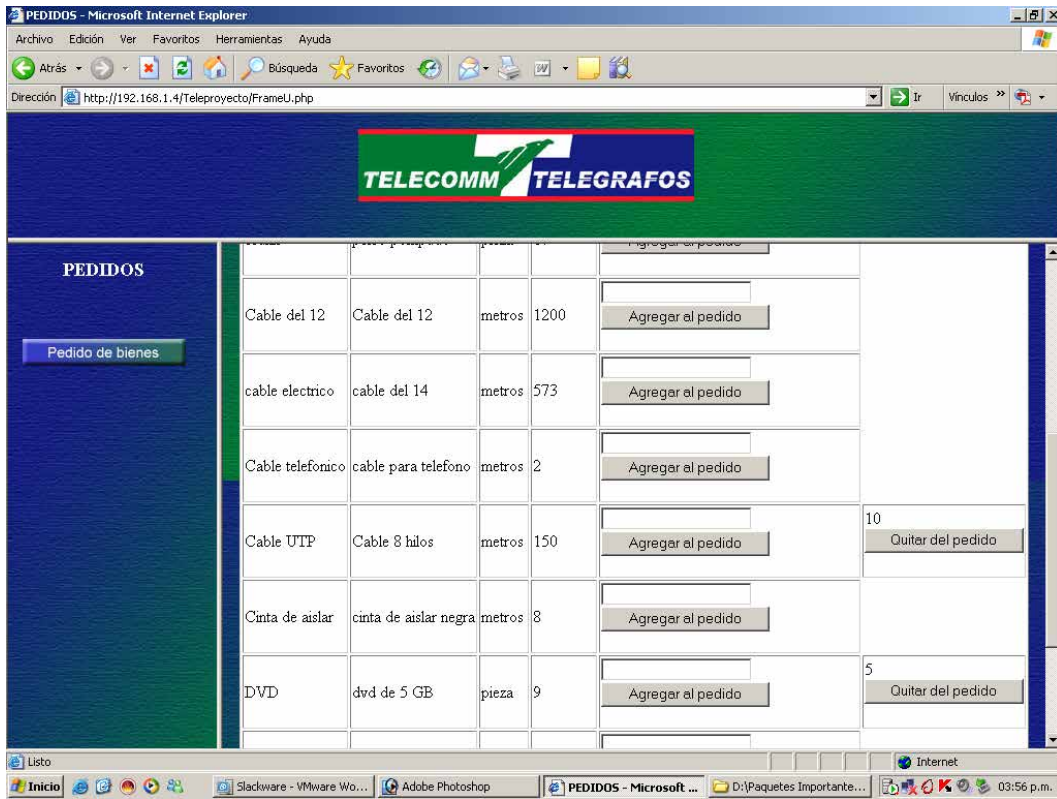


Fig. 3

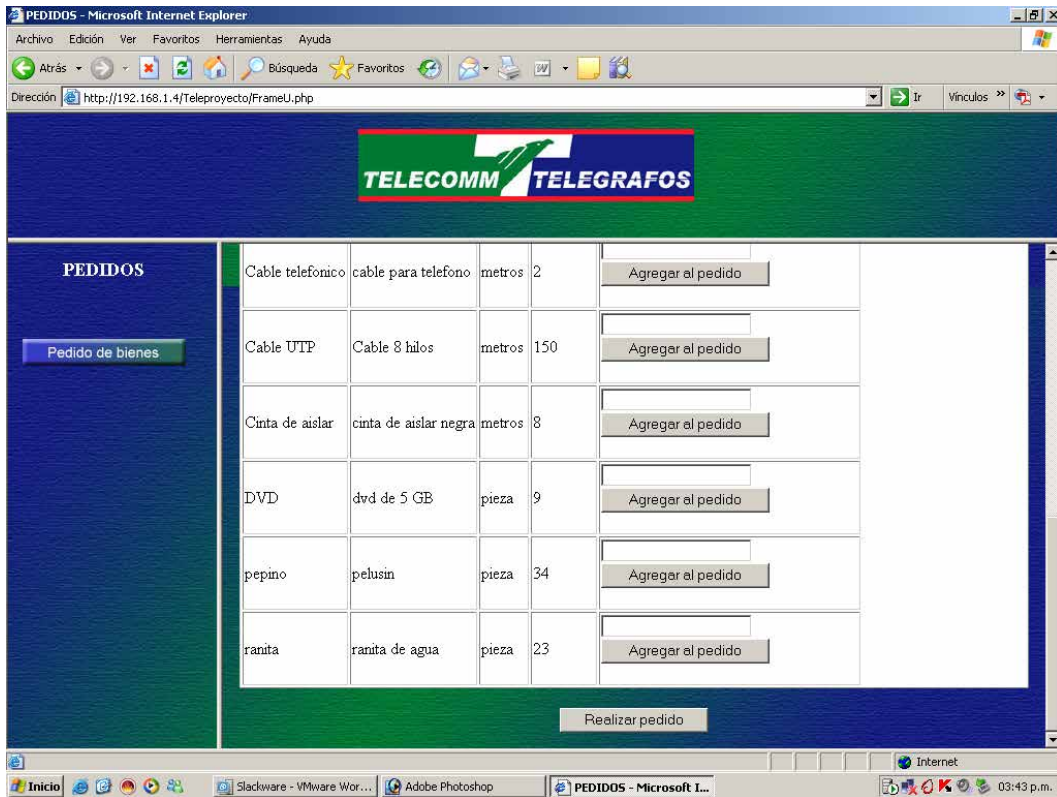


Fig. 4



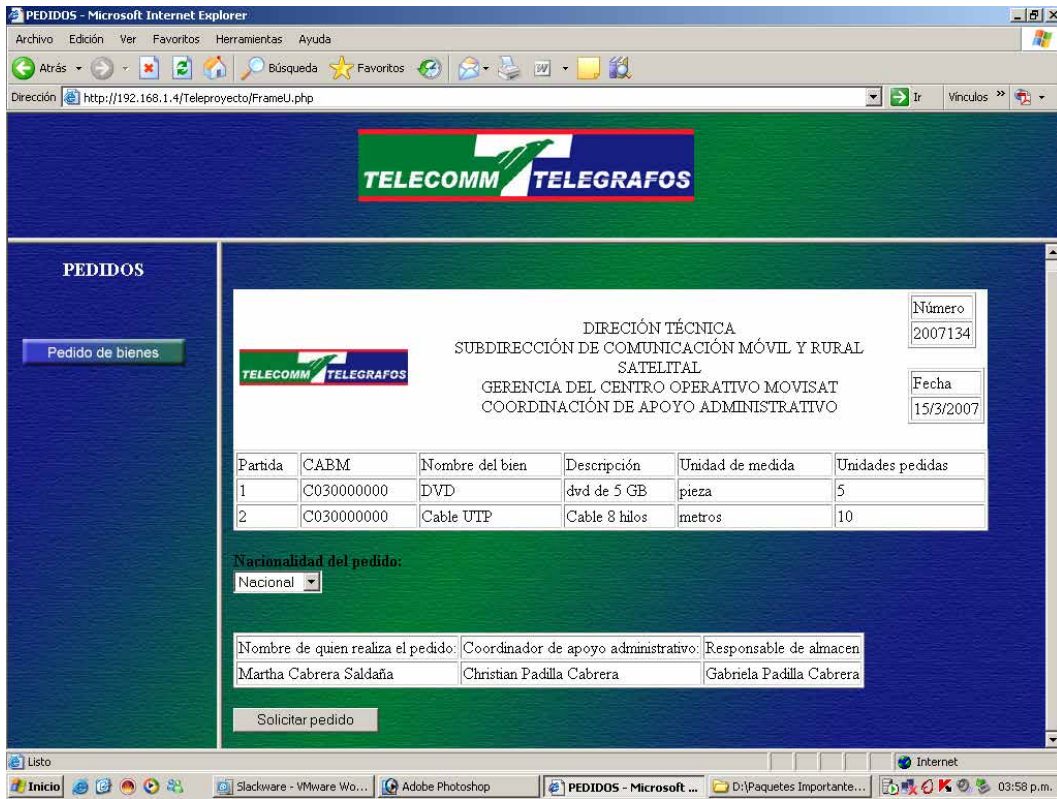



Fig. 5




Fig. 6

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT	Fecha 26-II-2007
	Manual de Administrador	Página 1 de 9
Sustituye a		
Fecha		
Página		

<b>ENTRADA DE BIENES</b>		
No. Operación	Responsables	Descripción
1	Administrador	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.
2	Administrador	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.
3	Administrador	Hacer clic sobre el botón “Entrada de bienes” (ver Fig. 2 anexo A), aparecerá una página que le mostrará el número de folio de la entrada que va a hacer. Este número lo generará automáticamente el sistema.
4	Administrador	Hacer clic sobre el botón “Continuar” (ver Fig. 3 anexo A). Aparecerá ahora otra página que tiene dos listas (para seleccionar el CABM) y varias cajas de texto para agregar el bien y todas sus características así también la cantidad de unidades que entran (ver Fig. 4 anexo A).
5	Administrador	Deberá hacer clic sobre la primera lista (ver Fig. 4 anexo A) aquí usted seleccionará (haciendo nuevamente clic sobre la opción deseada) el “tipo de CABM” que le corresponde al bien.
6	Administrador	Una vez seleccionado el tipo de CABM la lista de abajo (lista de CABM) se cargará con todos los CABM’s del tipo seleccionado. Aquí deberá seleccionar el CABM específico del bien haciendo clic sobre la opción requerida
Elaboró: Christian Padilla      Revisó      Autorizó		

**TELECOMM-GCOM-04**



	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		
	Manual de Administrador		
	Fecha	26-II-2007	
	Página	2	de 9
	Sustituye a		
Fecha			
Página			

<b>ENTRADA DE BIENES</b>					
No. Operación	Responsables	Descripción			
7	Administrador	Una vez seleccionado el CABM, se deberán llenar todos los campos del bien (nombre, descripción, unidad, etc.). Si usted introduce valores alfanuméricos en campos que son numéricos aparecerá un letrero, avisándole que esta cometiendo dicho error, el cual deberá corregir. Asegúrese de no dejar algún campo en blanco.			
8	Administrador	Una vez llenados todos los campos presione el botón “Agregar a la entrada” (ver Fig. 4 anexo A) para agregar el bien a la entrada. Sea cuidadoso en este paso ya que una vez presionado el botón, el sistema habrá guardado el bien y no hay marcha atrás. Este botón lo mandará a otra página que le mostrará el bien que acabo de agregar y dos opciones (botones):”Insertar otro bien” y “Terminar las entradas” (ver Fig. 5 anexo A).			
9	Administrador	Deberá decidir si quiere seguir agregando bienes a la entrada o si quiere finalizar la entrada (ver Fig. 5 anexo A).			
10.1	Administrador	Si decide seguir agregando bienes a la entrada presiones el botón “Insertar otro bien” y repita los pasos 5,6,7,8 y 9.			
10.2	Administrador	Si decide terminar de introducir bienes a la entrada, entonces, oprima el botón “Terminar las entradas” (ver Fig. 5 anexo A). Deberá tener cuidado ya que una vez presionado el botón, finalizará la entrada y ya no podrá agregar o quitar bienes a esa entrada.			
<table border="1" style="width: 100%;"> <tr> <td style="width: 33%;">Elaboró: Christian Padilla</td> <td style="width: 33%;">Revisó</td> <td style="width: 33%;">Autorizó</td> </tr> </table>			Elaboró: Christian Padilla	Revisó	Autorizó
Elaboró: Christian Padilla	Revisó	Autorizó			

**TELECOMM-GCOM-05**

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT
	Manual de Administrador

Fecha	26-II-2007		
Página	3	de	9
Sustituye a			
Fecha			
Página			

### AUTORIZACIÓN DE PEDIDOS

No. Operación	Responsables	Descripción
1	Administrador	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.
2	Administrador	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.
3	Administrador	Oprimir el botón “Autorización de pedidos” (ver Fig. 2 anexo A), el cual mostrará una página con un listado de todos los pedidos que han sido solicitados, dicho listado contiene el folio, la fecha, la nacionalidad y el usuario que realizó dicho pedido (entre otras cosas).
4	Administrador	Seleccionar el pedido a evaluar con el botón “Ver” (ver Fig. 7 anexo A) éste nos mandará a una página que nos muestra a detalle el pedido.
5	Administrador	Evaluar si el pedido se autoriza o no.
6.1	Administrador	Si se autoriza el pedido presionar sobre el botón “Autorizar” (ver Fig. 8 anexo A) y automáticamente el sistema autoriza el pedido y manda un e-mail al usuario que lo solicitó de que ya fue autorizado su pedido y que lo vaya a recoger lo antes posible a la bodega.

Elaboró: Christian Padilla	Revisó	Autorizó
----------------------------	--------	----------

TELECOMM-GCOM-06


	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT
	Manual de Administrador

Fecha	26-II-2007		
Página	4	de	9
Sustituye a			
Fecha			
Página			

### AUTORIZACIÓN DE PEDIDOS

No. Operación	Responsables	Descripción
6.2	Administrador	Si queremos rechazar el pedido presionamos el botón "Rechazar" (ver Fig. 8 anexo A) y el sistema actualizará la existencia de los bienes sumando lo que hubo en el pedido a las existencias actuales y enviará un e-mail al usuario para que sepa que su pedido fue rechazado
Elaboró: Christian Padilla	Revisó	Autorizó

TELECOMM-GCOM-07

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT	Fecha 26-II-2007
	Manual de Administrador	Página 5 de 9
Sustituye a		
Fecha		
Página		

### ADMINISTRACIÓN DE USUARIOS

No. Operación	Responsables	Descripción
1	Administrador	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.
2	Administrador	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.
3	Administrador	<p>Seleccionamos la opción “Administración de usuarios” (ver Fig. 2 anexo A) que se encuentra del lado izquierdo.</p> <p>Aparecerá una página con dos listas (ver Fig. 9 anexo A):</p> <p>a) La primera nos indica el tipo de operación que deseamos realizar (agregar, modificar o borrar).</p> <p>b) La segunda el tipo de usuario (Administrador, Responsable de bodega y usuario).</p>
4	Administrador	<p>Selecciona el tipo de operación y el tipo de usuario deseado. Ejemplo:</p> <p>Si queremos agregar un usuario seleccionamos de la primer lista “Agregar” y de la segunda “Usuario”</p>
5	Administrador	Para ejecutar la operación seleccionada oprima “Continuar” (ver Fig. 9 anexo A).
6.1	Administrador	Si seleccionó agregar Administrador aparecerá una página que le pide todos los datos de la persona que usted desea agregar. Deberá llenar todos los campos y seleccionar el tipo de administrador (Primario o secundario). El sistema solo admite un Administrador Primario ya que es quien aparecerá en los pedidos que se realicen, pero puede agregar los administradores secundarios que desee. Al final pulse “Agregar nuevo administrador” (ver Fig. 10 anexo A).

Elaboró: Christian Padilla	Revisó	Autorizó
----------------------------	--------	----------

TELECOMM-GCOM-08

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		Fecha	26-II-2007		
	Manual de Administrador		Página	6	de	9
			Sustituye a			
			Fecha			
			Página			

ADMINISTRACIÓN DE USUARIOS		
No. Operación	Responsables	Descripción
6.2	Administrador	Si seleccionó agregar responsable de bodega aparecerá una página que le pide todos los datos de la persona que usted desea agregar. Deberá llenar todos los campos y seleccionar el tipo de responsable de bodega (Primario o secundario). El sistema solo admite un responsable de bodega primario ya que es quien aparecerá en los pedidos que se realicen, pero puede agregar los responsables de bodega secundarios que desee. Al final pulse “Agregar nuevo administrador” (ver Fig. 11 anexo A).
6.3	Administrador	Si seleccionó agregar usuario aparecerá una página que le pide todos los datos de la persona que usted desea agregar. Deberá llenar todos los campos. Al final pulse “Agregar nuevo Usuario” (ver Fig. 12 anexo A).
6.4	Administrador	Si seleccionó modificar administrador aparecerá una página que muestra un listado de los administradores que hay (ver Fig. 13 anexo A), pulse “modificar” para modificar el administrador. “Modificar” te manda a otra página que te muestra toda la información del administrador a manera de que puedes modificar los valores existentes.
6.4.1	Administrador	Deberá de modificar los campos deseados y dejar intactos los campos que desea sigan con los mismos valores y por último presione “Modificar” (ver Fig. 14 anexo A) para que la cuenta se modifique.
6.5	Administrador	Si seleccionó modificar responsable de bodega aparecerá una página que muestra un listado de los responsables de bodega que hay, pulse “modificar” (ver Fig. 15 anexo A) para modificar el responsable de bodega. “Modificar” te manda a otra página que te muestra toda la información del responsable de bodega a manera de que puedes modificar los valores existentes.
Elaboró: Christian Padilla		
Revisó		Autorizó

TELECOMM-GCOM-09


	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT	Fecha 26-II-2007
	Manual de Administrador	Página 7 de 9
Sustituye a		
Fecha		
Página		

### ADMINISTRACIÓN DE USUARIOS

No. Operación	Responsables	Descripción
6.5.1	Administrador	Deberá de modificar los campos deseados y dejar intactos los campos que desea sigan con los mismos valores y por último presione “Modificar” para que la cuenta se modifique (ver Fig. 16 anexo A).
6.6	Administrador	Si seleccionó modificar usuario aparecerá una página que muestra un listado de los usuarios que hay, pulse “modificar” (ver Fig. 17 anexo A) para modificar el usuario. Este botón te manda a otra página que te muestra toda la información del usuario a manera de que puedes modificar los valores existentes.
6.6.1	Administrador	Deberá de modificar los campos deseados y dejar intactos los campos que desea sigan con los mismos valores y por último presione “Modificar” para que la cuenta se modifique (ver Fig. 18 anexo A).
6.7	Administrador	Si seleccionó borrar (no importa si fue administrador, responsable de bodega o usuario) aparecerá un listado con todas las cuentas del tipo de usuario que usted escogió (Usuario, administrador, etc.) pulse borrar a la cuenta que desea borrar y será borrada del sistema (ver Fig. 19 anexo A).


Elaboró: Christian Padilla	Revisó	Autorizó
----------------------------	--------	----------

**TELECOMM-GCOM-10**

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		
	Manual de Administrador		
	Fecha	26-II-2007	
	Página	8	de 9
	Sustituye a		
Fecha			
Página			

<b>REPORTES</b>					
No. Operación	Responsables	Descripción			
1	Administrador	Abrimos el portal (ver Fig. 1 anexo A) e insertamos la dirección del portal en el navegador Web.			
2	Administrador	Acceder introduciendo el “login” y el “password” que se le asignó. Después se oprime la tecla “Enter” o se da clic en el botón “Entrar”.			
3	Administrador	Pulsar el botón “Reportes” (ver Fig. 2 anexo A). Con esto aparecerá una página con 5 opciones a elegir (ver Fig. 20 anexo A): 1. Consulta completa de existencias. 2. Consulta mensual de salidas por coordinación. 3. Consulta mensual de salidas. 4. Consulta anual de salidas. 5. Consulta anual de entradas.			
4.1	Administrador	Si quiere realizar una consulta de existencias pulse la opción “Consulta completa de existencias” y aparecerá una tabla con todos los bienes con sus respectivas características y existencias (ver Fig. 21 anexo A).			
4.2	Administrador	Si quiere saber todo lo que se le entregó a una coordinación “x”, en el mes “y”, presione el botón “Consulta mensual de salidas por coordinación”. Aparecerá una página que le pedirá el mes, el año y la coordinación (ver Fig. 22 anexo A).			
4.2.1	Administrador	Llene los campos de mes, año y coordinación, y después oprima “Consultar” (ver Fig. 22 anexo A). Con esto le aparecerá un listado (ver Fig. 23 anexo A) con todos los pedidos que se hicieron desde la gerencia y mes seleccionados.			
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">Elaboró: Christian Padilla</td> <td style="width: 33%;">Revisó</td> <td style="width: 33%;">Autorizó</td> </tr> </table>			Elaboró: Christian Padilla	Revisó	Autorizó
Elaboró: Christian Padilla	Revisó	Autorizó			

**TELECOMM-GCOM-11**

	TELECOMM GERENCIA DEL CENTRO OPERATIVO MOVISAT		
	Manual de Administrador		
	Fecha	26-II-2007	
	Página	9	de 9
	Sustituye a		
Fecha			
Página			

REPORTES		
No. Operación	Responsables	Descripción
4.3	Administrador	Si lo que quiere es saber que fue lo que salió en un mes “x”, sin que importe la gerencia, entonces pulse “Consulta mensual de salidas”. Aparecerá un página que le pedirá el año y el mes (ver Fig. 24 anexo A).
4.3.1	Administrador	Introduzca el año y seleccione el mes deseado y a continuación presione “Consultar” (ver Fig. 24 anexo A). Aparecerán todos los pedidos hechos en el mes y año seleccionados (ver Fig. 25 anexo A).
4.4	Administrador	Para saber sobre todo lo que salió en un año presione “Consulta anual de salidas”. Aparecerá una página pidiéndole el año que desea consultar (ver Fig. 26 anexo A).
4.4.1	Administrador	Introduzca el año deseado e inmediatamente oprima “Consulta anual de salidas” (ver Fig. 26 anexo A). Aparecerá un listado con todos los pedidos realizados en el año seleccionado (ver Fig. 27 anexo A).
4.5	Administrador	Si desea saber sobre todo lo que ha <b>entrado</b> en un año presione el botón “Consulta anual de entradas”. Aparecerá una página que le pide el año que desea ver (ver Fig. 28 anexo A).
4.5.1	Administrador	Introduzca el año a ver y a continuación presione “consultar” (ver Fig. 28 anexo A). Aparecerá un listado con todas las entradas realizadas en el año deseado (ver Fig. 29 anexo A).
Elaboró: Christian Padilla      Revisó _____      Autorizó _____		

TELECOMM-GCOM-12



## ANEXO A

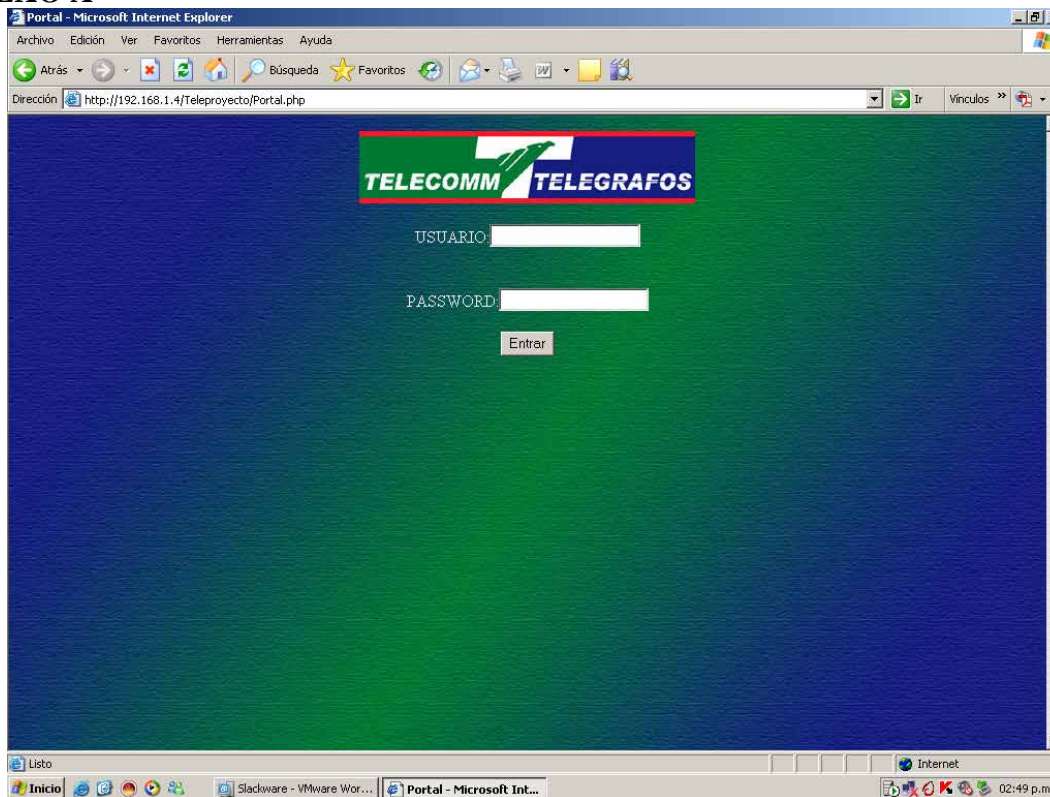


Fig. 1

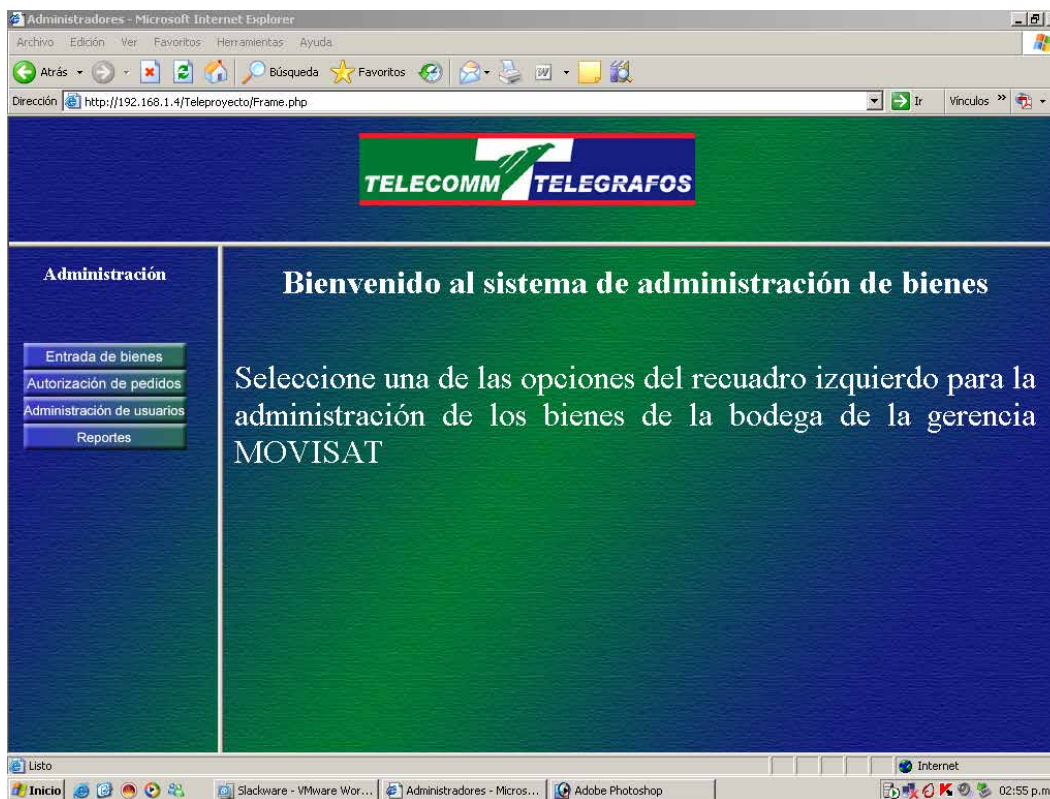


Fig. 2

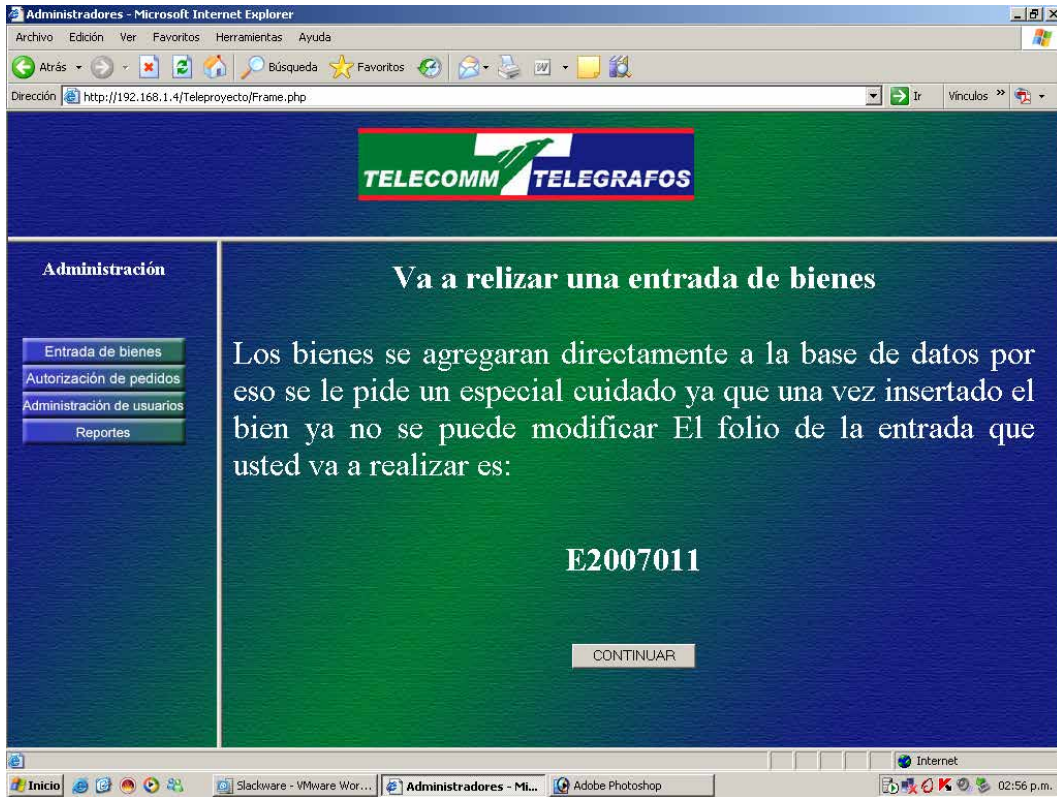


Fig. 3

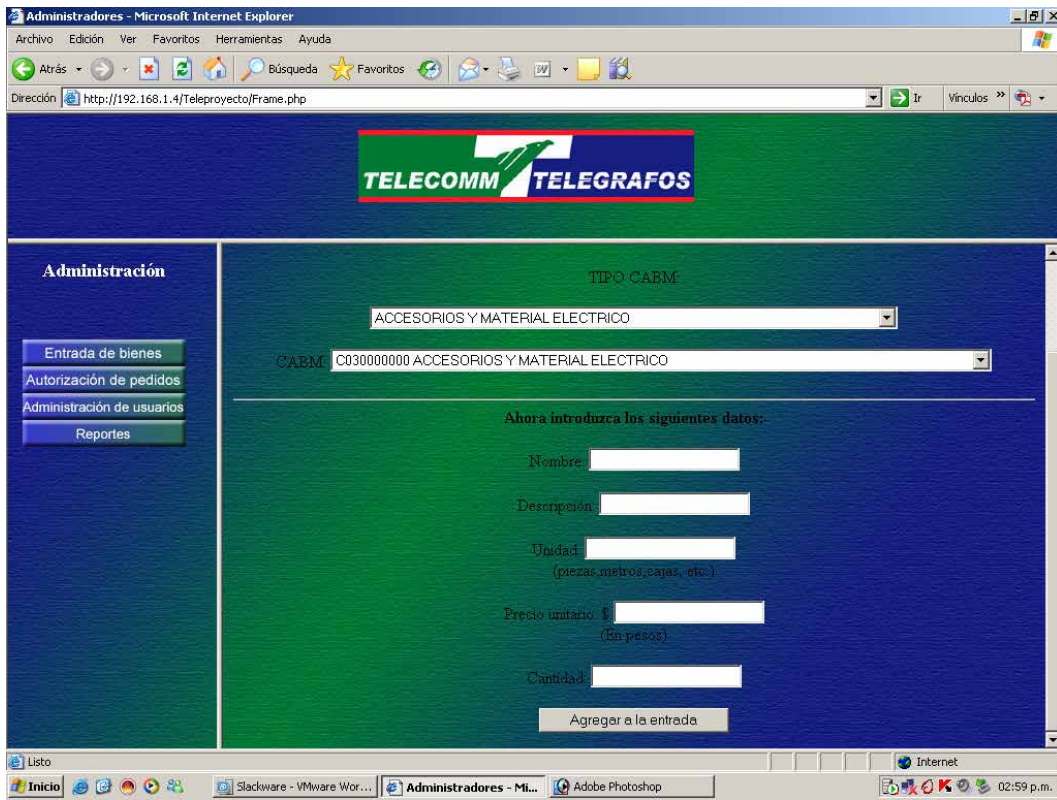


Fig. 4





Fig. 5



Fig. 6



Fig. 7

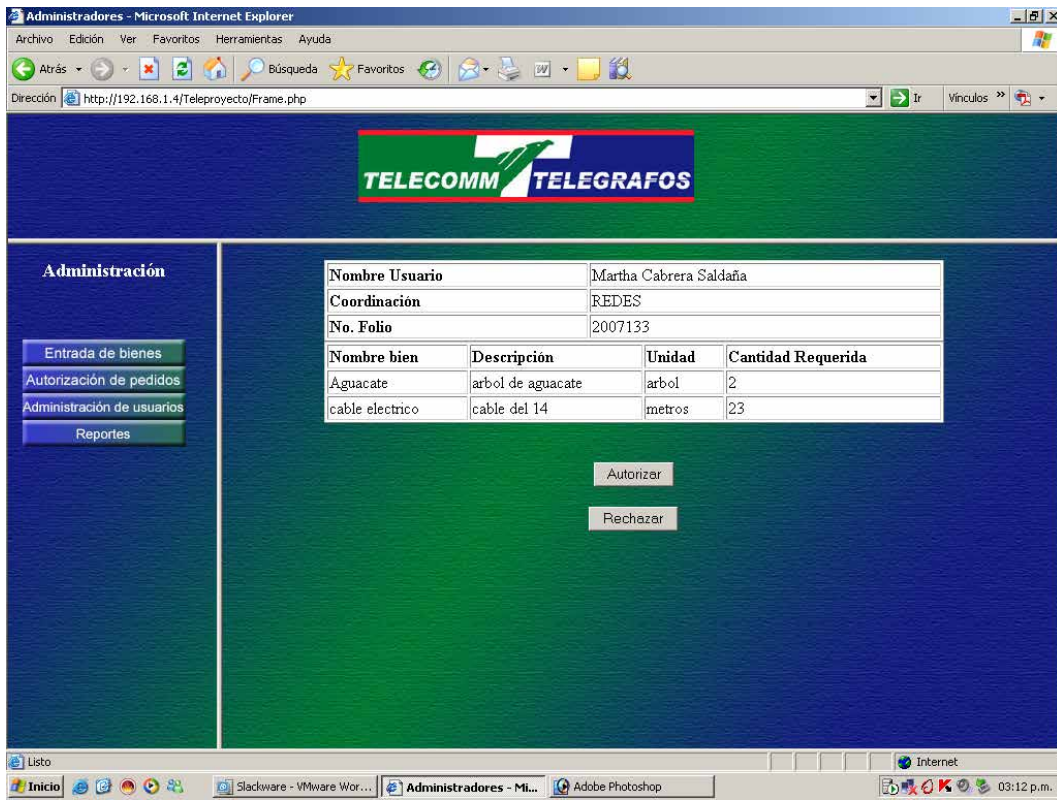


Fig. 8



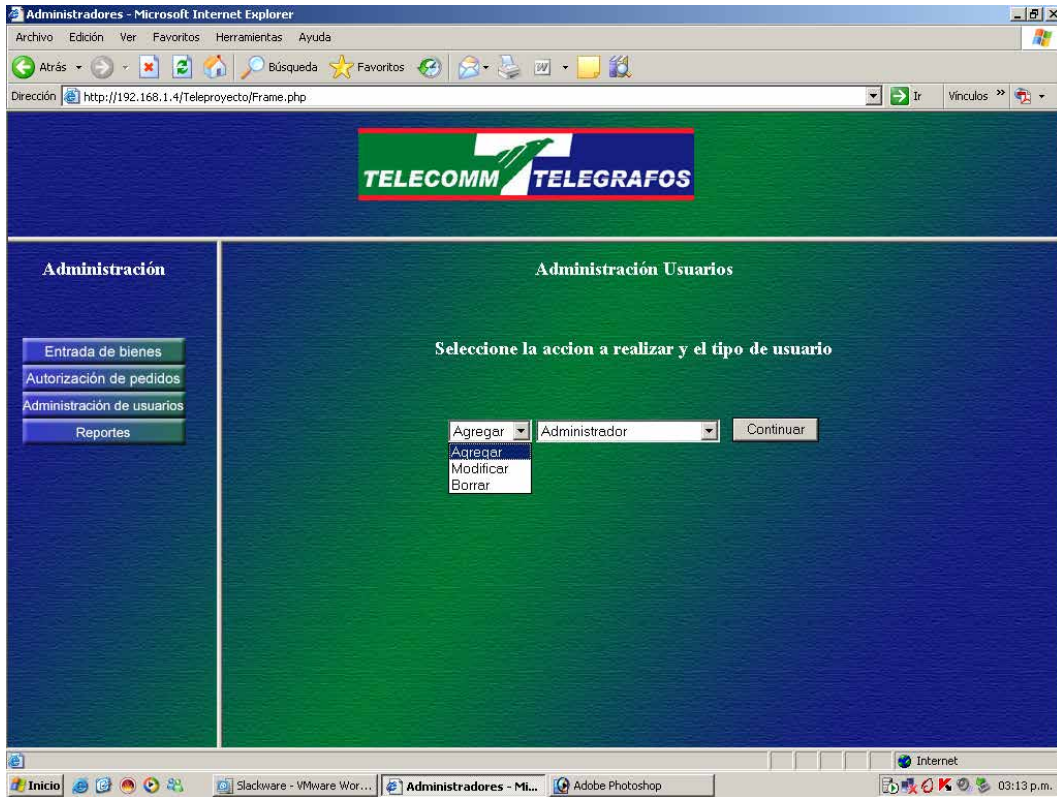


Fig. 9



Fig. 10



Fig. 11

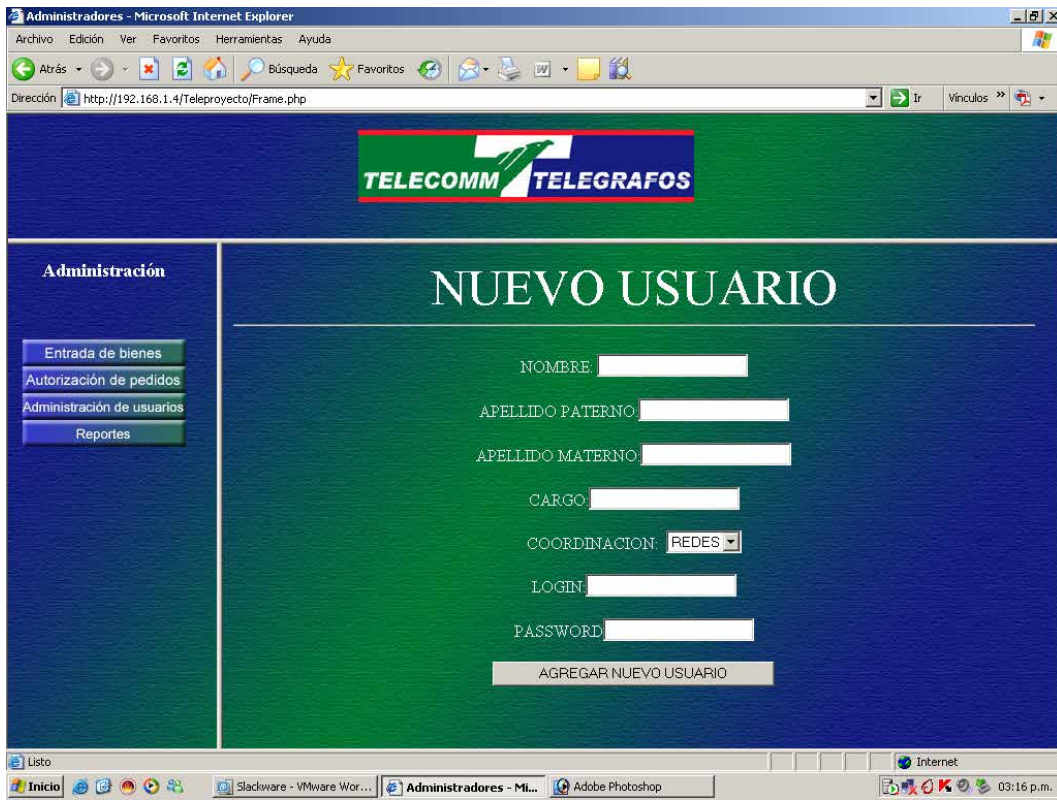


Fig. 12



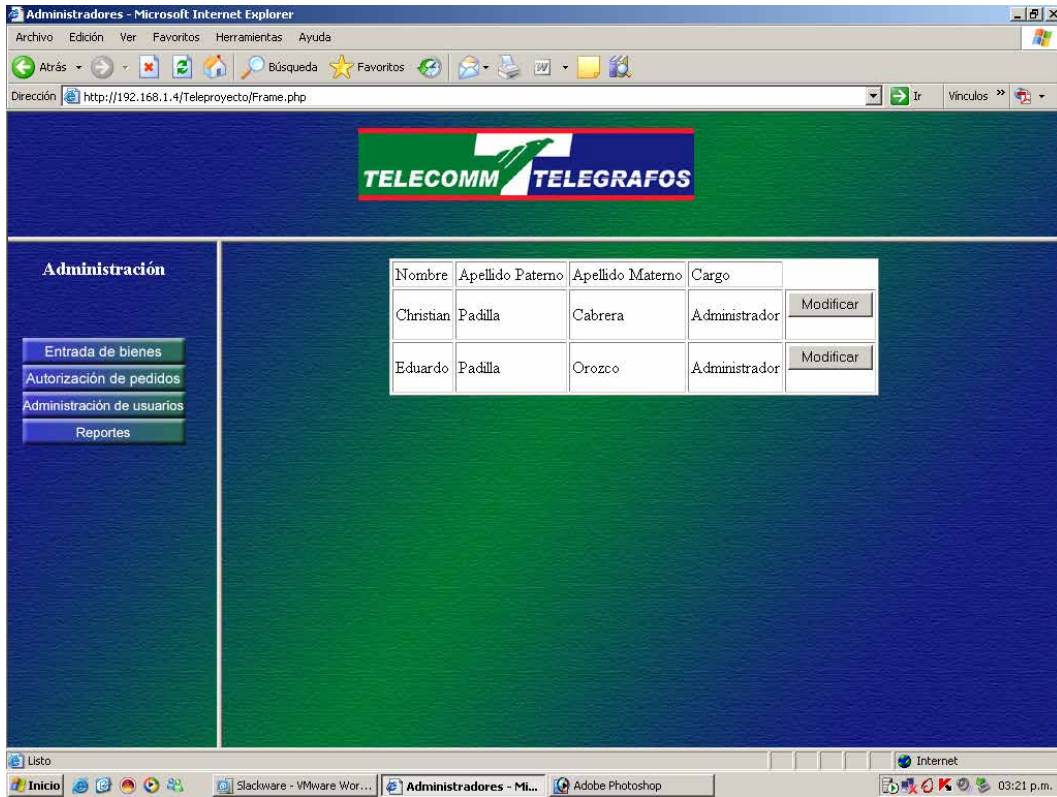


Fig. 13



Fig. 14

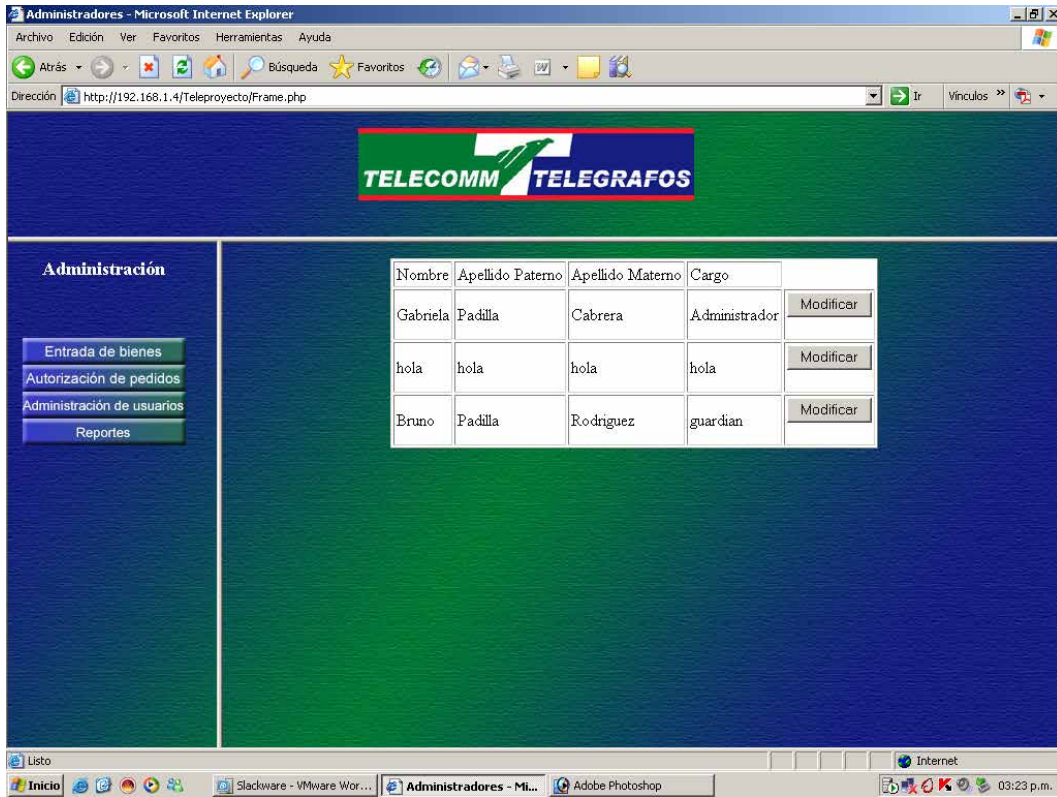


Fig. 15



Fig. 16



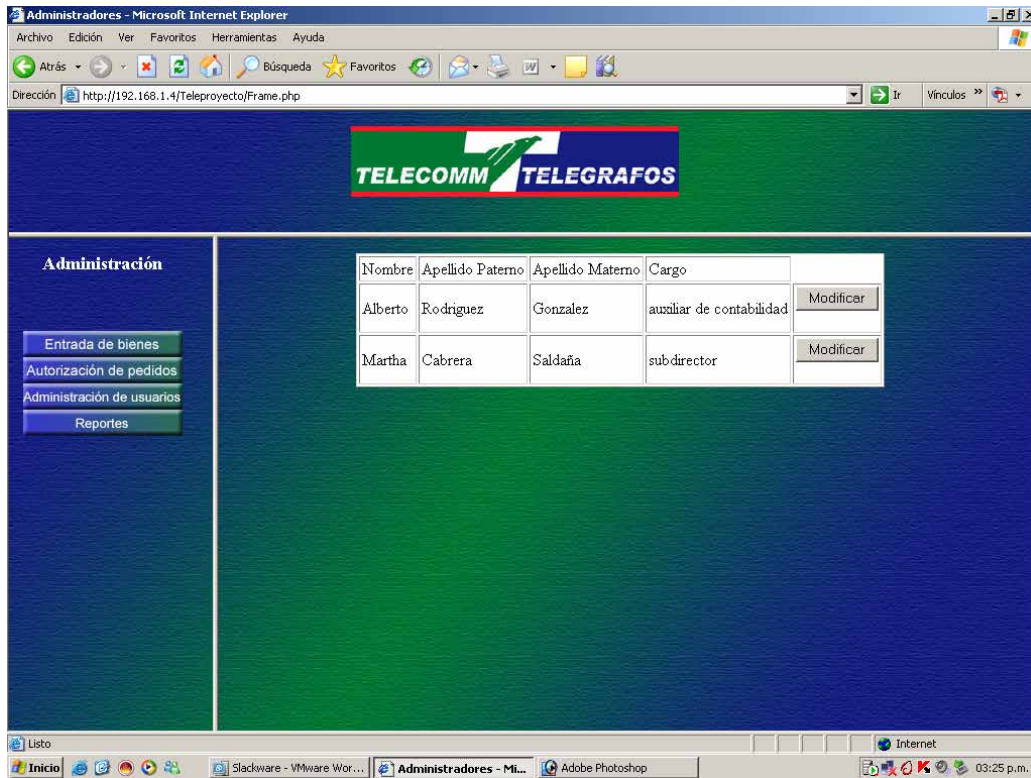


Fig. 17



Fig. 18

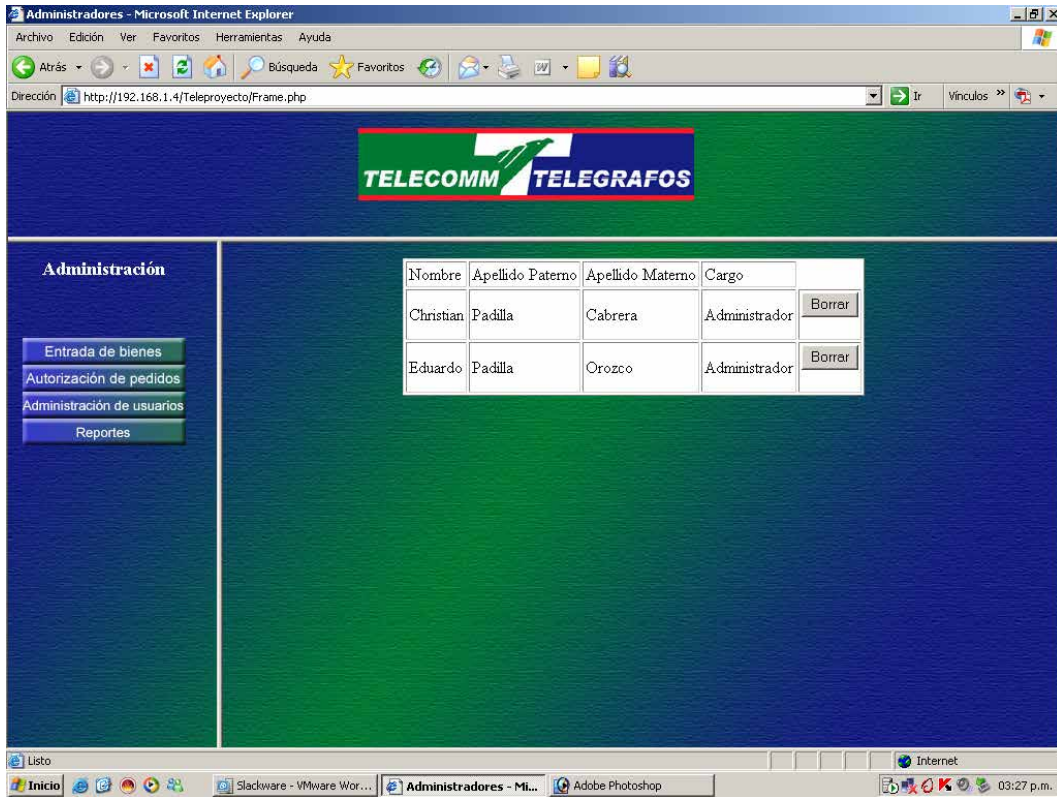


Fig. 19



Fig. 20





Fig. 21



Fig. 22

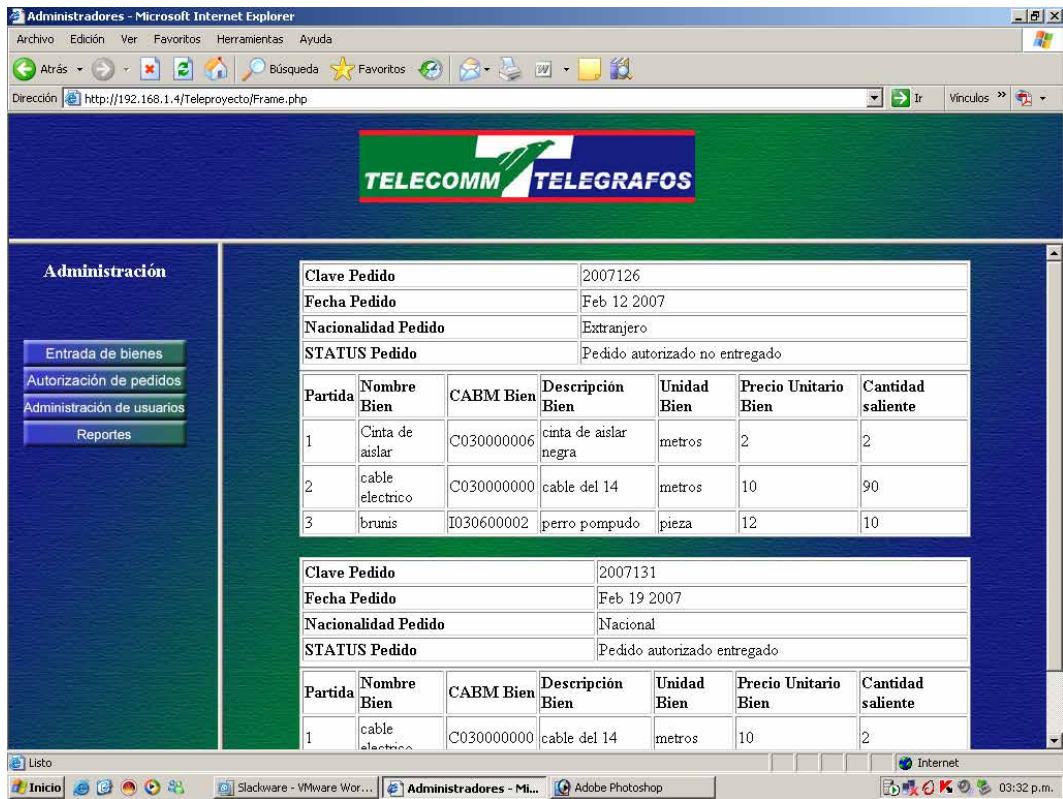


Fig. 23



Fig. 24



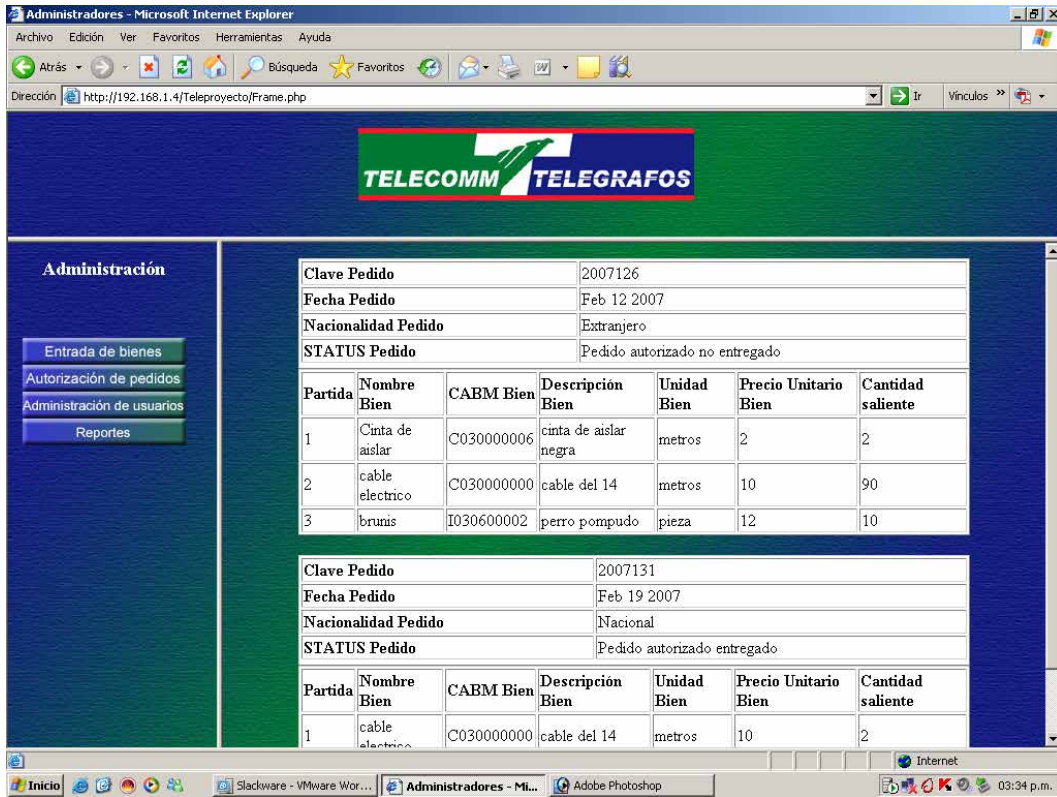


Fig. 25




Fig. 26

Administradores - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos

Dirección http://192.168.1.4/Teleproyecto/Frame.php Ir Vinculos



**Administración**

Entrada de bienes

Autorización de pedidos

Administración de usuarios

Reportes

Clave Pedido	2007124
Fecha Pedido	Dec 02 2007
Nacionalidad Pedido	Nacional
STATUS Pedido	Pedido autorizado no entregado

Partida	Nombre Bien	CABM Bien	Descripción Bien	Unidad Bien	Precio Unitario Bien	Cantidad saliente
1	agave	C570200002	planta de agabe	planta	12	1
2	Aguacate	C570200004	arbol de aguacate	arbol	123	3
3	Cable UTP	C030000000	Cable 8 hilos	metros	12	10

Clave Pedido	2007125
Fecha Pedido	Dec 02 2007
Nacionalidad Pedido	Nacional
STATUS Pedido	Pedido autorizado entregado

Partida	Nombre Bien	CABM Bien	Descripción Bien	Unidad Bien	Precio Unitario Bien	Cantidad saliente
1	cable electrico	C030000000	cable del 14	metros	10	10

Inicio Slackware - VMware Wor... Administradores - Mi... Adobe Photoshop Internet 03:36 p.m.


Fig. 27

Administradores - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos

Dirección http://192.168.1.4/Teleproyecto/Frame.php Ir Vinculos



**Administración**

Entrada de bienes

Autorización de pedidos

Administración de usuarios

Reportes

## Consulta anual de entradas

Introduce el año

Año:

2007

Consultar

Inicio Slackware - VMware Wor... Administradores - Mi... Adobe Photoshop Internet 03:37 p.m.


Fig. 28

Administradores - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos Ir Vinculos

Dirección http://192.168.1.4/Teleproyecto/Frame.php



**Administración**

Entrada de bienes

Autorización de pedidos

Administración de usuarios

Reportes

## Consulta anual de entradas

<b>Clave entrada</b>		E2007005			
<b>Fecha entrada</b>		Nov 02 2007			
Nombre Bien	CABM Bien	Descripción Bien	Unidad Bien	Precio Unitario Bien	Cantidad Entrante
Cable UTP	C030000000	Cable 8 hilos	metros	12	200
Aguacate	C570200004	arbol de aguacate	arbol	123	23
agave	C570200002	planta de agabe	planta	12	2
DVD	C030000000	dvd de 5 GB	pieza	23	12

<b>Clave entrada</b>		E2007006			
<b>Fecha entrada</b>		Nov 02 2007			
Nombre Bien	CABM Bien	Descripción Bien	Unidad Bien	Precio Unitario Bien	Cantidad Entrante
cable electrico	C030000000	cable del 14	metros	10	700

Inicio Slackware - VMware Wor... Administradores - Mi... Adobe Photoshop 03:38 p.m.

Fig. 29

## CONCLUSIONES

El nuevo sistema mejoró los tiempos en que se llevan acabo los pedidos, por lo cuál, se ahorra bastante tiempo que puede ser utilizado para otra actividad. Se brindó movilidad a todos los usuarios del sistema, ya que ahora desde cualquier PC (que se encuentre dentro de la intranet de MOVISAT), uno puede realizar pedidos, verificar existencias, autorizar pedidos y realizar entradas entre otras cosas.

Ahora se tiene un mejor control de las existencias de los bienes, lo que ayuda a disminuir la pérdida y robo de éstos, en conclusión se solucionaron los problemas que había con el antiguo sistema.

En lo personal adquirí bastante experiencia en el desarrollo de sistemas, ya que el estar haciendo entrevistas, calculando tiempos, diseñando los procesos, programando, implementando, etc., me di cuenta de todo lo que conlleva realizar un buen sistema que de solución a los problemas y necesidades de una empresa, y también cabe señalar que el próximo sistema que tenga que diseñar, lo podré hacer de una forma más rápida y eficiente.

También concluí que para desarrollar un sistema, no sólo se necesita saber programar, sino también, es necesario involucrarse un poco con los procesos, información y personal de la empresa que tenga que ver con lo que estamos haciendo, y en lo particular, el diseñar una buena base de datos, conlleva una serie de procesos que contienen ciertos pasos a seguir y que además, uno como ingeniero en computación necesita tener conocimientos generales sobre las demás áreas, tales como administración, medicina, economía, etc. debido a que los sistemas pueden manejar diferente tipo información y procesos que no son de nuestro saber.



## GLOSARIO

**Apache:** Es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows y otras, que implementa el protocolo HTTP y la noción de sitio virtual.

**Archivo:** Sinónimo de fichero y esencia del almacenamiento informático. Delimitado por una cabecera y una marca de final, lugar donde puede estar ubicada cualquier tipo de información, ya sea texto, programas, imágenes, sonidos, etc.

**Base de datos:** Es un conjunto de tablas dedicadas a guardar información relacionada entre sí, con referencias entre ellas de manera que se complementen con el principio de no duplicidad de datos.

**CABM (Clave Administrativa de Bienes Muebles):** Es una clave que identifica a cada bien en particular.

**Comando:** Es una orden que se da, generalmente al sistema operativo de la computadora, para que realice la tarea que esa orden conlleva.

**Consola:** Es el interprete de comandos de los sistemas de la familia UNIX. Básicamente funcionan dando una línea de comandos al usuario y ejecutando los comandos que se le piden.

**C.T.O.** Centro de Telecomunicaciones y Operaciones.

**Driver:** Aunque puede referirse a dispositivos físicos, se utiliza normalmente para indicar los controladores de periféricos, es decir, el software que permite al Sistema Operativo reconocer y utilizar lo que tiene conectado el ordenador.

**Entrada:** En este trabajo se refiere a un conjunto de bienes que entran físicamente a la bodega de TELECOMM (MOVISAT).

**Hardware** Se denomina así al conjunto de componentes físicos dentro de la informática (un teclado, una placa, por ej.).

**HTML:** Es un lenguaje que se emplea para crear páginas Web. Un código escrito en este lenguaje es, básicamente, un texto que el navegador mostrará en formato de página Web.

**HTTP:** Protocolo de transferencia de hipertexto es el lenguaje que utilizan el cliente y el servidor para comunicarse entre si.

**Lenguaje de Programación** Son secuencias de instrucciones, más o menos parecidas al lenguaje humano, que permiten, una vez traducidas o compiladas, ejecutar las órdenes incluidas en el programa.

**Librerías** Conjunto de programas con un formato determinado para ser utilizados por los desarrolladores de aplicaciones, con el objetivo principal de evitar la repetición de procesos.

**Licencia GNU:** La licencia pública general (GNU) está destinada a garantizar la libertad de compartir y cambiar el software gratis.

**Memoria** Tanto en ordenadores como en periféricos, lugar donde se almacenan datos o programas mientras se están utilizando.

**Multitarea** Realización simultánea de varios procesos en un ordenador, teniendo en cuenta que éste y su Sistema Operativo lo permitan.

**Multiusuario** Utilización simultánea de un ordenador por varios usuarios, o dicho de otra manera, ordenador y Sistema Operativo con capacidad para permitir el trabajar desde distintas consolas con la misma CPU.

**Partida:** Es un número consecutivo que se le asigna a cada bien agregado a un pedido como por ejemplo:  
1 cable, 2 pinzas y 3 cd's.

**PHP:** Es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web. PHP es un acrónimo recurrente que significa "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, *Personal Home Page Tools*), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios Web.

**Programa** Instrucciones que varían según el lenguaje que se utiliza, pero cuyo fin es el de controlar las acciones que tiene que llevar a cabo el ordenador y sus periféricos.

**Protocolo** Es un término de comunicaciones y su función es fijar unas reglas de funcionamiento, a todos los niveles, a las que han de atenerse los distintos sistemas informáticos para poder comprenderse.

**Script** Son programas que se ejecutan en un servidor Web dedicados a procesar las peticiones que le llegan de los navegadores.

**Seguridad:** Es un mecanismo que da la garantía de un buen funcionamiento y que evita, en caso de fallo de otro mecanismo, que se produzca un accidente o daño.

**Sentencia** El conjunto de órdenes que tienen un sentido, de tal forma que el ordenador sepa ejecutar una acción. Un ejemplo, en pseudocódigo: Si I es mayor que X haz FuncionZ.

**Servidor** Se denomina así al ordenador que se encarga de suministrar lo necesario a una red, dependiendo de cual sea la finalidad de ésta.

**Sistema Operativo** Es un conjunto de programas que sirve de enlace entre el ordenador y el programador usuario. Son los responsables de gestionar los recursos del ordenador, discos duros, memorias, control de periféricos como pantallas, teclados, etc. De alguna forma unifican y estandarizan el funcionamiento de los ordenadores.

**Sitio (Site)** Se refiere a un lugar o dirección Web cualquiera con una URL.

**Software libre:** Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios.

**SQL** Structured Query Language. Lenguaje o sentencias para el manejo y consulta de bases de datos.

**SQL Server 2000:** Es un sistema de gestión de bases de datos relacional, bajo licencia de Microsoft.

**TELECOMM:** Telecomunicaciones de México.

**Web:** Puede referirse a "una Web" como una página, sitio o conjunto de sitios que proveen información por los medios descritos, o a "la Web", que es la enorme e interconectada red disponible prácticamente en todos los sitios de Internet

## BIBLIOGRAFÍA

- Bandel David y Napier Robert, “Edición Especial Linux”, 6º edición., Madrid 2001, Prentice Hall
- Bowen Rich & Coar Ken, “Servidor Apache al Descubierto”, Madrid 2000, Prentice Hall
- Castaño Adoración de Miguel y Piattini Mario, “Fundamentos y modelos de bases de datos”, 2<sup>da</sup> Edición, México 1999, Alfaomega-Ra-Ma
- Castaño Adoración de Miguel, Piattini Mario y Martínez Esperanza Marcos, “Diseño de bases de datos relacionales”, México 2000, Alfaomega-Ra-Ma
- Dee-Ann Le Blanc, “La Biblia de administración de sistemas Linux”, México, Anaya
- Dr Shwarte Jachim, “El gran libro de HTML como publicar en Internet”, México, Marcombo
- García Jiménez Francisco, “Guía de campo de Linux” México, RA-MA
- Guiérrez Abraham & Bravo Ginés, “PHP5 a través de ejemplos”, México 2005, Alfaomega RA-MA
- Lerdorf Ramus and McGredy Ric, “Programming PHP”, USA 2002, O'REILLY
- López Quijado José, “Domine HTML y DHTML” México, Alfaomega Ram-Ma
- Microsoft® Encarta® 2007. © 1993-2006 Microsoft Corporation. Reservados todos los derechos
- Minera Francisco José, “PHP Y MYSQL integración total”, Argentina 2005, MP Ediciones
- Pott Oliver, “Gran libro de HTML y XML Referencia t Práctica”, México, Marcombo-Boixareu Editores
- Williams E. Hugo and Lane David, “Web Database Applications with PHP and MYSQL”, USA 2004, O'REILLY

## PÁGINAS WEB

- <http://es.wikipedia.org/wiki/Gnu>
- <http://es.wikipedia.org/wiki/Php>
- <http://www.microsoft.com/spain/sql/2000/productinfo/whatsnew.aspx>
- <http://www.microsoft.com/spain/windowsserver2003/default.msp>