



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

## **“SISTEMA DE CÓMPUTO PARA EL MANEJO DE INFORMACIÓN EN SISTEMAS”**

T E S I S

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

P R E S E N T A

MARICELA MELÉNDEZ ACEVES

DIR. DE TESIS: ING. HÉCTOR RAÚL MEJÍA RAMÍREZ  
CODIRECTOR: ING. JOSÉ ARTURO ORIGEL COUTIÑO





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.

## **OBJETIVO**

Diseñar e Implementar un sistema de Información con las diferentes modalidades del control de Incidencias(Vacaciones, Faltas, Licencias, etc) que genera el personal de Servicios Administrativos Aduanales; con el fin de optimizar el manejo, la actualización y la búsqueda de la información de manera eficiente y oportuna para mantener el control de la información en forma automatizada.

# AGRADECIMIENTOS

## DIOS

Por darme la oportunidad de disfrutar mi vida con quienes me rodean y sobre todo permitirme concluir todo proyecto que me presentas.

## PAPÁS

Agradezco sus valiosos consejos, respeto, amor y confianza.

**Miguel Angel:** Demostrarme cuan esfuerzo extra se requiere para llegar a nuestras metas propuestas.

**Blanca Rosa:** Saber apreciar mejor mis logros.

## HERMANAS

Escucharme y motivarme cuando más las necesité.

**Diana Aurora:** Alentarme con tu diligencia apremiante.

**Norma Beatriz:** Conducir cualquier situación por medio de una gran rectitud de espíritu.

**Cecilia Elena:** Enseñarme a tener templanza en las caídas y éxitos.

## ABUELOS PTERNOS

Con su gran apoyo y cariño incondicional.

**Abue Eulalia:** Mantendré un gran recuerdo de ti, pues sé que en donde te encuentres, tu gran entusiasmo y fuerza moral siempre permanecerá en mí.

## UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO Y FACULTAD DE INGENIERÍA

La gran contribución que tuve de la UNAM y la FI en mi trayectoria estudiantil.

## PERSONAS ESPECIALES

A la gran aportación que me brindaron todas las personas en el desarrollo de la tesis.

A mis profesores y compañeros por brindarme con incontables sugerencias y atenciones el enfrentarme a mi vida profesional.

## MARICELA MELÉNDEZ ACEVES

ACEPTAR EL CURSO DE LOS ACONTECIMIENTOS PROPIOS Y EXTRAÑOS EN UN CONTINUO Y CONSECUENTE ESFUERZO POR SUPERAR ETAPAS O DIFICULTADES PASADAS.



# INDICE

Introducción	3
I.- Marco Teórico	
1.1 Conceptos	4
1.1.1 Definición de Bases de Datos	4
1.1.2 Redes	7
1.1.3 Topología de las redes de influencia local	8
1.1.4 Protocolos	9
1.1.5 World Wide Web(WWW)	10
1.1.6 Intranet	10
1.1.7 Internet	10
1.1.8 Exploradores de internet	11
1.1.9 Cliente/Servidor	11
1.2 Metodologías	
1.2..1 Programación Estructurada	11
1.2.2 Metodología Orientada a Objetos	12
1.2.3 Rational Unified Process(RUP)	14
II.- Análisis del Sistema	
2.1 Identificación del Problema	15
2.1.1 Propuestas de Solución	16
2.2 Estructura Organizacional de las Áreas involucradas	25
2.3 Documento de información de funciones actuales	26
2.4 Estándares de Programación	27
2.4.1 Estándares existentes aplicables al proyecto	29
2.5 Hardware	30
2.6 Software	31
III.- Diseño del Sistema	
3.1 Modelado del sistema	33
3.2 Diagrama de Casos de Uso	33
3.3 Diagrama de Flujo de Datos	38
3.4 Diccionario de Datos	41
3.5 Diagrama Entidad Relación	45
3.6 Normalización	47
IV.- Desarrollo del sistema	
4.1 Construcción del Sistema	51
4.1.1 Creación de la Base de datos	51
4.1.2 Creación de las tablas en la base de datos	53
4.2 Construcción de Pantallas	56
V.- Implantación y Pruebas del sistema	
5.1 Pruebas del sistema	58
5.1.1 Categoría de Pruebas	58
5.1.2 Consideraciones para la ejecución de Pruebas	59
5.1.3 Pruebas aplicadas	59
5.2 Mantenimiento	62
5.3 Capacitación	63
Resultados y Conclusiones	64
Bibliografía	65
Anexos	
Apéndice A Manual del usuario	67
Apéndice B Documentación del código	74

# INTRODUCCIÓN

# INTRODUCCIÓN

Actualmente en Servicios Administrativos Aduanales localizada en Aduana México tiene principalmente el problema de integración y control de la información de su personal, ya que se ha observado que trámites administrativos tienden a aplazarse y además sus actividades laborales repercuten hacia otras administraciones.

Ante esta problemática el presente trabajo de tesis tiene como objetivo desarrollar un sistema que trabaje en ambiente web, que cumpla con los requerimientos solicitados por el administrador o subadministrador del área respectiva para controlar las incidencias generadas por el personal de forma automatizada en sus respectivas administraciones; obteniendo un mejor desempeño en sus actividades de manera que trámites a realizar se lleven a cabo eficientemente.

La propuesta de desarrollo del sistema dentro de Servicios Administrativos Aduanales parte de que el sistema que trabaja en un ambiente web tiende a integrar nuevas tecnologías web en los diversos departamentos que conforman una administración, así como el obtener mejoras en cuanto a la ejecución de actividades por cada usuario.

Cabe destacar que el desarrollo del sistema cubre solo la Subadministración de Recursos Humanos por el enfoque administrativo en el cual se realizan los trámites respectivos de Incidencias del personal que integra Servicios Administrativos Aduanales.

El presente trabajo está integrado por seis capítulos: I Marco Teórico, II Análisis del Sistema, III Diseño del Sistema, IV Desarrollo del Sistema, V Implantación y Pruebas del Sistema, Resultados y Conclusiones.

En el capítulo I se presentan los conceptos relacionados al desarrollo del sistema de información, como: bases de datos, metodologías de desarrollo.

En el capítulo II se describe de forma general la manera en que opera el personal de Servicios Administrativos Aduanales, así como la identificación del problema.

En el capítulo III se presenta el diseño del sistema, modelando diagramas de flujo y diccionario de datos.

En el capítulo IV se integra el desarrollo del sistema, creando la base de datos.

En el capítulo V se implementa el sistema, describiendo las pruebas realizadas para que operara sin errores.

Por último se presentan las conclusiones y resultados obtenidos a los que se llegaron en base a los objetivos y desarrollo del sistema. Así como también se incluye la bibliografía consultada y dos apéndices uno llamado el manual del usuario y otro con la respectiva documentación del código para el sistema.

# **I. MARCO TEÓRICO**

# I. MARCO TEÓRICO

## 1.1 Conceptos

### 1.1.1 Definición de Bases de Datos

La Base de Datos (DataBase), se puede definir como el conjunto de datos organizados que son almacenados, recolectados y explotados por los Sistemas de Información de una empresa o negocio en particular y pueden acceder a ellos de manera sencilla, con la posibilidad de relacionarlos, ordenarlos en base a diferentes criterios.

Las bases de datos son uno de los grupos de aplicaciones de productividad personal más extendidos. Entre las más conocidas pueden citarse dBase, Paradox, Access y Aproach, para entornos PC, y Oracle, ADABAS, DB/2, Informix o Ingres, para sistemas medios y grandes.

Las bases de datos nos proporcionan la infraestructura requerida para el sistema de apoyo a la toma de decisiones y para los sistemas de información estratégicos, ya que estos sistemas explotan la información contenida en las bases de datos de la organización para apoyar el proceso de toma de decisiones o para lograr ventajas competitivas.

Algunos problemas que nos permiten solucionar las bases de datos son:

- Redundancia e inconsistencia de los datos.
- Dificultad para tener acceso a los datos.
- Seguridad

Ventajas de las Bases de Datos

- Minimizar la redundancia
- Aumentar consistencia
- Integrar datos(control de información)
- Facilidad para compartir datos
- Flexibilidad de acceso a los datos.

Para el diseño de Bases de Datos se cuenta con elementos primordiales que la conforman, y se mencionan con la finalidad de diseñar del sistema; ya que sin la ayuda de ellas no sería posible tener la factibilidad y facilidad para su análisis posterior.

**Llave primaria:** Es un atributo el cual definimos como atributo principal, es una forma única de identificar a una entidad. Por ejemplo, el RFC de un empleado se distingue de otro porque los RFC no pueden ser iguales y se tiene que agregar una llave adicional en dado caso de que existieran dos RFC iguales.

**Tabla:** Es la unidad básica de almacenamiento en un sistema de bases de datos

relacionales, en ellas son almacenados los datos de los usuarios y los datos del sistema.

Cada tabla se compone de varias columnas las cuales cuentan con un tipo de datos asociado. La información sobre la estructura de todas las tablas se encuentra en el diccionario de datos.

**Diccionario de Datos:** Es la herramienta que sirve para identificar y clasificar los datos almacenados en la base de datos y está integrada por archivos, registros y campos que contienen información descriptiva de todos los tipos de datos producidos, controlados, intercambiados y mantenidos en la Base de Datos.

**Modelo entidad-relación** (Denominado por sus siglas como E-R): Este modelo representa a la realidad a través de **entidades**, que son objetos que existen y que se distinguen de otros por sus características, por ejemplo: un alumno se distingue de otro por sus características particulares como lo es el nombre, o el número de control asignado al entrar a una institución educativa, así mismo, un empleado, una materia, etc.

Las entidades pueden ser de dos tipos:

Tangibles : Son todos aquellos objetos físicos que podemos ver, tocar o sentir.

Intangibles: Todos aquellos eventos u objetos conceptuales que no podemos ver, aún sabiendo que existen.

Las características de las entidades en base de datos se llaman **atributos**, por ejemplo el nombre, dirección teléfono, etc. son atributos de la entidad usuario; Clave, Departamento, etc., son atributos de la entidad empleado. A su vez una entidad se puede asociar o relacionar con más entidades a través de **relaciones**.

Para representar un modelo E-R gráficamente, se emplean símbolos, los cuales se muestran en la Tabla 3 :



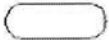

Símbolo	Representa
	Entidad
	Relación
	Atributos
	Ligas

Tabla 3. Símbolos empleados en un modelo E-R

**Modelo relacional.** En el modelo relacional, los datos y las relaciones entre los datos se representan por medio de una serie de tablas, en las cuales los renglones (tuplas) equivalen a los cada uno de los registros que contendrá la base de datos, y las columnas corresponden a las características (atributos) de cada registro localizado en la tupla y cada una esta compuesta por nombres

únicos. Una columna de una tabla representa una relación entre un conjunto de valores.

Existe una correspondencia entre el concepto de tabla y el concepto matemático de relación, del cual recibe su nombre el modelo relacional.

### **Normalización**

Es el proceso que consiste en comprobar que las tablas definidas cumplen unas determinadas condiciones. Se pretende garantizar la no existencia de redundancia y una cierta coherencia en la representación mediante un esquema relacional de las entidades y relaciones del diagrama E-R. Mediante la normalización se pueden solucionar diversos errores en el diseño de la base de datos así como para mejorarlo.

**Usuarios** se pueden definir como toda persona que tenga todo tipo de contacto con el sistema de base de datos desde que este se diseña, elabora, termina y se usa.

Para los Usuarios de la Base de Datos que se manejará se muestran algunas de sus características a continuación:

Existen diferentes puntos de vista y aplicaciones que los usuarios llevan a cabo sobre una Base de Datos, es por ello que el Manejador de Base de Datos debe contar con los mecanismos necesarios que estén de acuerdo con el tipo de usuario involucrado.

Para ello los usuarios que accedan a la base de datos se pueden clasificar como:

- Programadores de Aplicaciones:** Programan aplicaciones de uso más o menos específico por medio de llamadas en DML (Lenguaje de Manipulación de Datos), las cuales están incorporadas en un programa escrito en un lenguaje de programación (Por ejemplo, COBOL, PL/I, Pascal, C, etc.) con el cual se interactúa con el sistema.

- Usuarios Ingenuos:** Los usuarios interactúan con el sistema invocando a uno de los programas de aplicación permanentes que se han escrito anteriormente en el sistema de base de datos, se puede mencionar al usuario como el usuario final que utiliza el sistema de base de datos sin saber nada del diseño interno del mismo .

- Administrador de la Base de Datos:** El DBA (**DataBase Administrator**), es un usuario especial que puede controlar la Base de Datos, modificar la estructura de los datos para mejorar el acceso y optimizar el uso de los recursos(Añadir o borrar índices, dividir en distintos archivos y usar nuevos discos u ordenadores). Dentro de las funciones del DBA (Administrador) se pueden mencionar las siguientes:

- Control Global de la BD:** El DBA es en general el que controla el funcionamiento global de la BD. Puede ser una o varias personas.

- Definir el Esquema de la BD:** A partir del modelo conceptual, construye sentencias del DDL que se traducirán en estructuras de datos e información en el diccionario de datos del SGBD usado.

- Definir Estructuras de Almacenamiento:** Se trata de definir cómo y donde se almacenará cada tipo de datos: Datos de la BD y del diccionario, del sistema de recuperación(*rollback system*), de los índices

(*indexes*), de las aplicaciones o herramientas (*tools*), de información temporal.

La correcta administración de estas estructuras de almacenamiento puede conseguir aumentar mucho la **eficiencia de un SGBD**, lo cual es muy importante.

•**Conceder/Revocar Permisos de Acceso** (*grant/revoke*): Para consultar, insertar, actualizar o borrar datos, pero también para crear objetos (tablas, vistas, índices, usuarios), ejecutar o crear programas, conceder permisos a terceros.

•**Especificación de Restricciones de Integridad**: Para no admitir valores imposibles o repetidos en la BD, restricciones de integridad referencial.

El DBA contribuye con su trabajo al funcionamiento eficaz de todos los sistemas que se ejecutan con la base de datos ; además ofrece asistencia técnica a quienes interactúan con la misma. Dentro de sus responsabilidades están:

- Ajuste de la base de datos para conseguir el rendimiento óptimo.
- Estrategias de copia de seguridad y recuperación.
- Colaboración con el personal de administración del sistema y desarrolladores de aplicaciones.
- **Dimensionamiento de la Base de Datos.** El SGBD deberá garantizar que es capaz de manejar el volumen de datos requerido. Para ello, deberá comprobarse que es adecuado en cada uno de los siguientes puntos:

- Número total de bases de datos que se van a crear.
- Número total de tablas por base de datos.
- Número máximo de filas por tabla.
- Longitud máxima de fila.
- Número máximo de índices por tabla.
- Número máximo de campos por índice.

*Se consideraron las definiciones anteriores, para explicar en los siguientes capítulos el funcionamiento del sistema con mayor detalle.*

## 1.1.2 Redes

Las redes constan de dos o más dispositivos electrónicos unidos que comparten recursos (archivos, CD-ROM, o impresoras) y que son capaces de realizar transacciones electrónicas. Las redes pueden estar unidas por cable, líneas de teléfono, satélites, etc.



Las redes pueden clasificarse en términos de cobertura en:

- Redes LAN (Local Area Network, Redes de Área Local)
- Redes Man (Metropolitan Area Network, Redes de Área Metropolitana)
- Redes WAN (Wide Area Network, Redes de Área Extensa)

Redes LAN (Local Area Network, Redes de Área Local) : Es una red de datos de alta velocidad, tolerante a fallas, que cubre un área geográfica relativamente pequeña. Por lo regular conecta estaciones de trabajo, computadoras personales, impresoras y otros dispositivos en un solo edificio u otra área geográfica limitada.

Tienen muchas ventajas para los usuarios de computadoras, entre ellas el acceso compartido a dispositivos y aplicaciones, el intercambio de archivos entre los usuarios conectados y la comunicación entre usuarios vía correo electrónico.

Los estándares de transmisión en redes LAN son conocidos como protocolos de transmisión siendo un conjunto de normas que rigen la comunicación entre las computadoras de una red. Estas normas especifican qué tipo de cables se utilizarán, qué topología tendrá la red, qué velocidad tendrán las comunicaciones y de que forma se accederá al canal de transmisión.

Redes Man (Metropolitan Area Network, Redes de Área Metropolitana): Las redes de área metropolitana son principalmente una red local muy grande que cubre mayores extensiones como puede ser una ciudad, mediante la interconexión de redes LAN por medio de una línea grande de transmisión llamada backbone.

Redes WAN (Wide Area Network, Redes de Área Extensa): Es una red de comunicación de datos que da servicio a los usuarios localizados en un área geográfica amplia y generalmente utiliza los dispositivos de transmisión que ofrecen las compañías de telecomunicaciones.

*Considerando las anteriores características se puede mencionar que para nuestro fin en el sistema y por tener la comunicación compartida a todos los dispositivos conectados a la red está soportado por el tipo de Red LAN.*

### **1.1.3 Topología de las redes de influencia local**

La arquitectura TCP/IP que las redes de ordenadores surgieron fue por la necesidad de interconectar los diferentes host de una empresa o institución para poder compartir recursos y equipos específicos.

Pero los diferentes componentes que van a formar una red se pueden interconectar o unir de diferentes formas, siendo la forma elegida un factor fundamental que va a determinar el rendimiento y la funcionalidad de la red.

La disposición de los diferentes componentes de una red se conoce con el nombre de topología de la red. La topología idónea para una red concreta va a depender de diferentes factores, como el número de máquinas a interconectar, el tipo de acceso al medio físico que deseemos, etc.

Podemos distinguir tres aspectos diferentes a la hora de considerar una topología:

1. La topología física, que es la disposición real de las máquinas, dispositivos de red y cableado (los medios) en la red.
2. La topología lógica, que es la forma en que las máquinas se comunican a través del medio físico. Los dos tipos más comunes de topologías lógicas son broadcast (Ethernet) y transmisión de tokens (Token Ring).

3. La topología matemática, mapas de nodos y enlaces, a menudo formando patrones.

La topología de broadcast simplemente significa que cada host envía sus datos hacia todos los demás hosts del medio de red. Las estaciones no siguen ningún orden para utilizar la red, sino que cada máquina accede a la red para transmitir datos en el momento en que lo necesita. Esta es la forma en que funciona Ethernet.

En cambio, la transmisión de tokens controla el acceso a la red al transmitir un token eléctrico de forma secuencial a cada host. Cuando un host recibe el token significa que puede enviar datos a través de la red. Si el host no tiene ningún dato para enviar, transmite el token hacia el siguiente host y el proceso se vuelve a repetir.

*En lo que respecta a la topología de Servicios Administrativos Aduanales:*

- *Las Instalaciones de cableado son para Red de Área Local, con Estándar de Transmisión Ethernet conocida como 10 Base-T.*
- *Medio de transmisión: cable UTP; el cual es un cable popular y de calidad.*
- *Además soporta tasas de transferencia de datos de 10Mbps.*

*De este modo los distintos elementos participantes de la red tales como servidores, computadoras de usuarios y demás se comunican entre sí utilizando la arquitectura TCP/IP y principalmente la dirección IP como identificador.*

## 1.1.4 Protocolos

Al enviar un mensaje a través de Internet, éste se divide en fragmentos(paquetes), que viajarán a través de routers desde el ordenador que envía el mensaje hacia el ordenador destino. Para asegurar que este mensaje llegue correctamente, sin fallos ni repeticiones, debe haber algo que controle esta comunicación, un protocolo. El protocolo es el medio de acceso a la información.

El protocolo TCP/IP es la pieza fundamental que constituye la red internet, ya que define una red de conmutación de paquetes, es decir, toda la información que se va a transmitir es fragmentada en paquetes. Cada uno de estos paquetes es enviado a la dirección de la computadora donde debe llegar la información, salta de red en red hasta llegar a su destino. Cada computadora tiene una dirección IP, estas direcciones son identificadores unívocos de cada computadora. Estos paquetes viajan a través de unos "routers" que eligen las rutas más viables que seguirán dichos paquetes hasta llegar a sus correspondientes destinos. Para controlar tanto el soporte físico como lógico de ese tráfico de datos se utiliza el protocolo TCP/IP, que es el corazón de esta red y hace que funcione.

Algunos de los beneficios del protocolo TCP/IP

- Es eficiente, fiable y funciona con poca sobrecarga de datos.
- Es un estándar abierto, totalmente disponible y que es totalmente independiente de cualquier fabricante de software y hardware.
- El protocolo de conectividad de redes y el tráfico de datos implica la transferencia de datos a través de su conexión la cual tolera cierta cantidad de retardo.

*TCP/IP se ha convertido en la base de Internet por lo que se usa dentro de Servicios Administrativos Aduanales es http, que funciona mediante la comunicación que se*

establece mediante un navegador y un Servidor http más conocidos como Web por ser el uso de la World Wide Web el más extendido.

### 1.1.5 World Wide Web(WWW)

WWW es un servicio que, a través de una interfaz gráfica, permite navegar por Internet consultando los documentos que se encuentren en la red.

Estos documentos están unidos entre sí a través de hipervínculos, que son palabras y gráficos que contienen direcciones de Web incrustadas.

(Los hipervínculos se distinguen fácilmente por tener un color distinto que el resto de un texto en página Web).

Los documentos y los hipervínculos forman entre ellos una red o Web de información.

Cada página Web tiene una dirección única, llamada dirección URL(localizador Uniforme de Recursos), que define el nombre del equipo donde se encuentra la página y su ruta.

Una URL se divide en dos partes:

- La primera parte indica el protocolo usado. Para acceder a las páginas de la Web se usa el protocolo "http".
- La segunda parte es el nombre del dominio de la computadora.

*Es utilizado en el presente sistema ya que está basado en la tecnología de hipertexto que le permite ir saltando de un lugar a otro en una página, o de un sitio a otro dentro de la red por medio de un enlace. Este enlace son ciertas palabras gráficas que resaltan o sobresalen dentro de la página que se está observando.*

### 1.1.6 Intranet

Una Intranet es una red configurada con el protocolo TCP/IP que no está conectada a internet, aunque usa los mismos estándares y herramientas de comunicación de Internet para suministrar información a los usuarios de la red privada.

Debido a que en Servicios Administrativos Aduanales se aprovecha **world wide web** , por tener gran popularidad debido a que está construida sobre la base de estándares en lugar de un producto suministrador informático en particular, además de su facilidad para la búsqueda y el acceso a la información.

### 1.1.7 Internet

Es una red mundial de equipos que se comunican usando un lenguaje común. Es la red de redes, un conjunto de redes que conectan millones de ordenadores, personas, programas, bases de datos, etc.

Internet no pertenece a ninguna organización o grupo particular. Es una red global que une alrededor de 20.000 redes a la vez. Existen grupos voluntarios que realizan algunos estándares de las tecnologías de Internet, pero no existe un cuerpo central que gobierne su estructura.

*Como la ayuda de internet se ha convertido en algo útil para todos los usuarios que usan la red; en Servicios Administrativos Aduanales se ha considerado como herramienta de fácil utilización y rápido acceso para el Desarrollo del Sistema.*

## 1.1.8 Exploradores de internet

Un explorador de la web ( World Wide Web,WWW) es una aplicación usada para explorar información a través de Internet, por eso se les conoce también como navegadores.

Los exploradores más conocidos , y que ofrecen una interfaz gráfica a la www son : Netscape Navigator y el Microsoft Internet Explorer.

En los navegadores nos encontramos una barra de direcciones que muestra la dirección de la página Web actual. En donde para ir a una página nueva, hay que escribir la dirección URL de la página y pulsar[Enter].

## 1.1.9 Cliente/Servidor

Es una estructura para conectar sistemas informáticos de una red.

El sistema cliente es la estación de trabajo y el servidor es una versión de mayor tamaño capaz de almacenar gran cantidad de datos y ejecutar grandes aplicaciones.

Cuando el ordenador se comunica con otro a través de Internet, en dicha comunicación usa el modelo cliente/servidor. Los recursos de internet los proporcionan unos determinados ordenadores, conocidos como servidores, que contienen las aplicaciones, el correo electrónico, etc.

Los clientes son los ordenadores que acceden a estos recursos a través de aplicaciones cliente, que usan el TCP/IP para proporcionar la información en pantalla en el formato que use el ordenador cliente.

Esta arquitectura se basa, en varias plataformas interconectadas: una de las cuales actúa como "servidor" de la BD en la que los datos están físicamente localizados y centraliza las funciones de administración. Las plataformas denominadas "clientes" realizan funciones de manejo de los interfaces de usuario, lógica de aplicación, etc.

*En el caso de Servicios Administrativos Aduanales el cliente es una computadora que pide una acción determinada a otra, la cual es el servidor. Este es la computadora que ejecuta las acciones requeridas por la computadora cliente. Al cliente sólo le interesan los resultados que le dá el servidor no cómo los hace.*

## 1.2 Metodologías

### 1.2.1 Programación Estructurada

La programación estructurada se refiere a escribir un programa de acuerdo a las reglas siguientes:

- Que el programa tenga un diseño modular
- Que los módulos sean diseñados de modo descendente
- Los módulos a ser codificados utilicen las estructuras de control de secuencia, selección y repetición.

En el **diseño modular** la programación se divide en módulos de los cuales cada uno ejecuta una única actividad o tarea y se codifica independientemente de otros módulos. Por ser independientes, los programadores pueden trabajar simultáneamente en diferentes partes del mismo programa, reduciendo con ello el

tiempo de diseño del algoritmo y posteriormente la codificación del programa. Por otra parte el módulo se puede modificar sin afectar a otros módulos siguiendo un método ascendente o descendente hasta llegar a una descomposición final del problema en módulos en forma jerárquica.

El **modo descendente** es el proceso mediante el cual el problema se descompone en una serie de niveles de refinamiento al efectuar una relación entre las sucesivas etapas de estructuración de modo que se relacionan unas con otras mediante entradas y salidas de información. Esto es, que se descompone el problema en etapas de forma que se puede considerar cada estructura para definir interfaces entre los diferentes módulos y programas, como también la estructura de datos y flujo de datos.

## 1.2.2 Metodología Orientada a Objetos

Una metodología de desarrollo de software Orientada a Objetos consta de:

- Conceptos y diagramas
- Etapas y definición de entregas en cada una de ellas
- Actividades y recomendaciones

Entre las metodologías de desarrollo de software Orientada a Objetos basada en UML se tienen:

- Metodología Object Oriented Design por Grady Booch
- Metodología Objectory por Ivar Jacobson
- Metodología Object Modeling Technique por James Rumbaugh et. al.

En la metodología publicada por Booch, Rumbaugh y Jacobson se pueden tener en cuenta las siguientes etapas :

### Etapas

- Análisis de Requerimientos
- Diseño del sistema
- Diseño detallado
- Implementación y pruebas

En cada etapa se resaltan las actividades que le dan cuerpo y los documentos que se esperan al final de cada una de ellas.

### Análisis de Requerimientos

En esta etapa se logra claridad sobre lo que desea el usuario y la forma en la cual se le va a presentar la solución que está buscando.

### Diseño del Sistema

En esta etapa se define una subdivisión en aplicaciones del sistema (si es lo suficientemente grande) y la forma de comunicación con los sistemas ya existentes con los cuales debe interactuar.

## Diseño detallado

En esta etapa se adecúa el análisis a las características específicas del ambiente de implementación y se completan las distintas aplicaciones del sistema con los modelos de control, interfaz o comunicaciones, según sea el caso.

## Implementación y pruebas

Se desarrolla el código de una manera certificada.

A continuación se detallarán las tres metodologías antes descritas:

### 1.- Metodología Object Oriented Design, por Grady Booch

La metodología de Booch usa los siguientes tipos de diagramas para describir las decisiones de análisis y diseño, tácticas y estratégicas, que deben ser hechas en la creación de un sistema orientado por objetos.

**Diagrama de Clases.** Consisten en un conjunto de clases y relaciones entre ellas. Puede contener clases, clases paramétricas, utilidades y metaclasses. Los tipos de relaciones son asociaciones, contenedora, herencia, uso, instanciación y metaclass.

**Especificación de Clases.** Es usado para capturar toda la información importante acerca de una clase en formato texto.

**Diagrama de Categorías.** Muestra clases agrupadas lógicamente bajo varias categorías.

**Diagramas de transición de estados.**

**Diagramas de Objetos.** Muestra objetos en el sistema y su relación lógica. Pueden ser diagramas de escenario, donde se muestra cómo colaboran los objetos en cierta operación; o diagramas de instancia, que muestra la existencia de los objetos y las relaciones estructurales entre ellos.

**Diagramas de Tiempo.** Aumenta un diagrama de objetos con información acerca de eventos externos y tiempo de llegada de los mensajes.

**Diagramas de módulos.** Muestra la localización de objetos y clases en módulos del diseño físico de un sistema. Un diagrama de módulos representa parte o la totalidad de la arquitectura de módulos del sistema.

**Subsistemas.** Un subsistema es una agrupación de módulos, útil en modelos de gran escala.

**Diagramas de procesos.** Muestra la localización de los procesos en los distintos procesadores de un ambiente distribuido.

### 2.- Metodología Ivar Jacobson

Objectory es un proceso organizado para la construcción industrial de software. Este proceso de diseño está guiado por casos de uso, una técnica que basa su centro en el entendimiento de un sistema en la forma en la cual es usado.

**Modelo de Casos de Uso:** Se basa en la descripción de elementos o usuarios externos al sistema (actores) y funcionalidad del sistema (casos de uso).

**Modelo de objetos:** Representa la estructura estática de objetos. Puede contener objetos entidad, de interfaz y de control, entre otros tipos; y relaciones de herencia, conocido (una referencia estática) y comunicación.

Diagrama de interacción. Muestran la secuencia de eventos entre paquetes u objetos necesarios para realizar un caso de uso.

Diagrama de estado. Muestra los estados internos de un objeto complejo.

Algunos de los conceptos más importantes de esta metodología son:

- Objeto Entidad. Representa información del sistema que debe sobrevivir cierto período de tiempo, por ejemplo, un caso de uso.
- Objeto de Interfaz. Modela información y comportamiento que es dependiente de la interfaz actual del sistema
- Objeto de Control. modela funcionalidad que no corresponde a ningún objeto en particular y que se presenta en algunos casos de uso. Estos objetos generalmente operan sobre varios objetos entidad, realizan algún algoritmo y retornan algún resultado a un objeto de interfaz.
- Paquete. Módulo que contiene código, traducible a un módulo en el lenguaje de implementación.
- Unidad. En pruebas, desde una clase hasta un subsistema

### **3.- Metodología Object Modeling Technique (OMT), por James Rumbaugh et al.**

OMT hace un cubrimiento de las etapas de análisis, diseño e implementación, dejando sin cubrir el modelamiento estratégico.

**Modelo de Objetos.** Se define como un diagrama de objetos mas un diccionario de datos. El diagrama de objetos muestra clases y sus relaciones (generalización, agregación, asociación, instanciación). El diccionario de datos es el detalle de las clases en el diagrama de objetos.

**Modelo dinámico.** Se define como un conjunto de diagramas de estado mas un diagrama de Flujo de eventos Global.

## **1.2.3 Rational Unified Process(RUP)**

Éste método ayuda a crear y mantener modelos con un fácil acceso a una base de conocimiento y herramientas en actividades críticas de desarrollo de software que actualmente es aplicable en el desarrollo de proyectos, así como diversas instituciones.

Para ello se cuenta con fases descriptivas sobre cómo utilizar de forma efectiva dicho método:

- Inicio: Para definir el alcance del proyecto
- Elaboración: Se especifican las características y el plan del proyecto.
- Construcción: Donde se construye un modelo completo de diseño
- Transición: Considerando los respectivos manuales de usuario y la documentación de desarrollo.

Además es necesario considerar que al usar dicho método se utiliza también herramientas de UML(Unified Modeling Language), como guía para utilizar de manera efectiva el desarrollo del sistema.

## **II. ANÁLISIS DEL SISTEMA**



## II .- ANÁLISIS DEL SISTEMA

En el análisis se profundizan las necesidades del usuario para la comprensión del problema y se documentan los requerimientos para el desarrollo del sistema.

Se deben especificar todas aquellas características que permitan definir los objetivos del sistema, es decir, la información que se obtendrá, los datos que se suministrarán, las funciones y el rendimiento esperado.

Para delimitar el sistema se tienen que tener presentes los siguientes objetivos:

- Identificar las necesidades de los usuarios
- Asignar las funciones al Hardware, Software, Bases de Datos, Elementos del sistema.

A continuación se detallarán cada uno de los objetivos anteriormente mencionados:

### **Identificación de Necesidades**

Entre las necesidades que actualmente enfrenta Servicios Administrativos Aduanales se tienen las siguientes:

- No se cuenta con una interfaz que permita crear o borrar bases de datos, respaldar las ya existentes o recuperar alguna base respaldada.
- Los usuarios no cuentan con permisos para hacer uso de la información en una Base de Datos.

### **Asignación de las funciones**

En el sistema es preciso que se manejen diferentes tipos de usuarios; los cuales puedan tener acceso al sistema con determinadas condiciones. Para ello se tienen:

- Administrador del Sistema: Que es la persona encargada de dar de alta las bases de datos, borrar una o más bases de datos del sistema, y borrar físicamente usuarios de la base de datos.
- Administrador de la Base de Datos: Quien dá de alta nuevos usuarios en la Base de Datos, y otorga permisos a cada uno de los usuarios pertenecientes a sus bases.
- Usuario del Sistema: Es quien puede a través de la interfaz para el usuario consultar y enriquecer la Base de Datos, dependiendo de los permisos que le fueron otorgados.

## 2.1 Identificación del Problema

El registro y control de la información en Servicios Administrativos Aduanales no cuenta en la actualidad con algún sistema informático con el cual pueda manejar el control de incidencias que genera su personal; y todo proceso se realiza de forma manual a través de la integración de expedientes que se clasifican y almacenan físicamente en un área que conforma su archivo.

Para ello es importante mencionar que en el área de Recursos Humanos de Servicios Administrativos Aduanales se realiza la captura, el control y análisis de Incidencias del personal por medio de hojas de cálculo Excel que maneja de una manera poco práctica, y no es óptima, ya que cuando se requiere realizar una consulta detallada el tiempo para localizar la información y poder generar un informe con los datos solicitados se extiende de manera considerable y es poco confiable.

Ante esta problemática el presente trabajo tiene como objetivo primordial desarrollar un sistema que trabaje en ambiente Web, permitiendo a los usuarios llevar a cabo un control en el registro de Incidencias de tal forma que solucione los principales problemas de comunicación entre los distintos departamentos.

La propuesta de desarrollo de un sistema dentro de Servicios Administrativos Aduanales parte de la premisa de un sistema que trabaja en ambiente Web, ya que cuenta con la ayuda de internet y el uso de la red interna a través de intranet (como los medios útiles para lograr la difusión interna a fin de tener la comunicación de todo tipo de información entre el personal y las diferentes áreas), ayudando a tener la capacidad de interactuar con los usuarios, difundir la información general y específica de las Incidencias, así como realizar sus modificaciones utilizando de una manera eficiente la información en el sistema informático que almacene y optimice el manejo de la misma, evitando con ello duplicidad de información y consolidación de los datos relacionados.

Una vez identificado el problema, se mencionarán los requerimientos del usuario con relación al sistema:

- Debe existir un control de seguridad en los accesos a la información.
- Que se pueda acceder al sistema desde cualquier equipo que cumpla requerimientos mínimos que permitan ejecutar y transferir archivos sin problemas de horarios.
- Asegurar la confidencialidad de los documentos entre los diferentes departamentos, verificando que los documentos confidenciales sean vistos solamente por la persona a quien van dirigidos.

## 2.1.1 Propuestas de Solución

A continuación se mencionan las diferentes opciones de solución consideradas para el funcionamiento, así como el desarrollo del Sistema en Servicios Administrativos Aduanales; para ello se tomaron en cuenta los requerimientos, ventajas y principales características citadas en tres etapas de la siguiente forma:

En la Primera Etapa se muestran los Sistemas Manejadores de Bases de Datos posibles a ser usados: SQL SERVER 2000, INFORMIX ONLINE 7.2, ORACLE 9i.

En la Segunda Etapa las opciones de Sistemas Operativos, tales como: WINDOWS NT SERVER y UNIX.

Y en la Tercera Etapa las Herramientas para crear Páginas Web, tales como: ASP (ACTIVE SERVER PAGES), PHP (HYPERTEXT PREPROCESSOR, PROCESADOR DE HIPERTEXTO), JSP (JAVA SERVER PAGES, SERVIDOR DE PAGINAS JAVA).

### Opciones de solución:

- **Primero** se mostrarán las opciones de Sistemas Manejadores de Bases de Datos (DBMS, Data Base Management System) :

Los SGBD (Sistema Gestor de Bases de Datos) o DBMS (Data Base Management System). Son un conjunto de programas que se encargan de administrar, controlar, acceder y manejar la creación y todos los accesos a las bases de datos.

Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos.

En sí, un sistema manejador de base de datos es el corazón de la base de datos ya que se encarga del control total de los posibles aspectos que la puedan afectar.

Algunas de las funciones principales de un SGBD son:

- Crear y organizar la Base de datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser accedidos rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos. Esto a través de las sentencias en DML al comando de el sistema de archivos. Así el manejador de base de datos es el responsable del verdadero almacenamiento de los datos.
- Respaldo y recuperación. Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.
- Control de concurrencia. Consiste en controlar la interacción entre los usuarios concurrentes para no afectar la inconsistencia de los datos.
- Seguridad e integridad. Consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

La figura 1 muestra el DBMS como interfase entre la base de datos física y las peticiones del usuario. El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal.

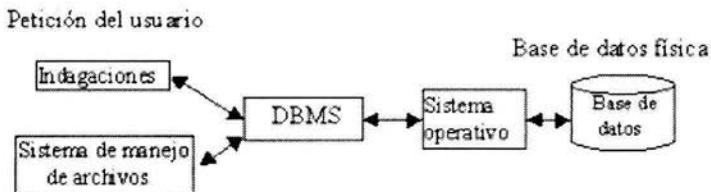


Figura 1. Peticiones del usuario al DBMS

Existen varios manejadores de Bases de Datos como son: SQL Server, Informix, Oracle.

Para elegir el manejador más adecuado para el desarrollo del sistema es necesario conocer las características de cada uno, por lo que se muestran a continuación:

### **SQL (Structured Query language, Lenguaje de Consultas Estructurado)**

En sistemas de administración de bases de datos, lenguaje de consultas desarrollado por IBM que se ha vuelto el estándar para consultas de bases de datos en una red cliente/servidor. Los cuatro comandos básicos(SELECT, UPDATE, DELETE e INSERT) corresponden a las cuatro funciones básicas de la manipulación de datos(recuperación, modificación, eliminación e inserción). Las consultas SQL se aproximan a la estructura de una consulta de lenguaje natural en inglés. Los resultados de una consulta se muestran en una tabla de datos que consta de columnas(correspondientes a los campos de datos) y filas (correspondientes a los registros de datos).

SQL es un lenguaje tanto para la definición de los datos como para la manipulación de ellos; contando también con las siguientes características:

- Es un lenguaje estándar reconocido por ANSI e ISO.
- Se encuentra implementado en la mayoría de los DBMS.
- Usa una combinación de construcciones del álgebra relacional y del cálculo relacional.
- Es utilizado desde muchos programas de aplicación que forman parte de un RDBMS (Relational Data Base Management System, Sistemas Manejadores de Bases de Datos Relacionales)
- Es utilizado para la obtención, modificación y definición de los datos, así como para la Administración de la Base de Datos.

Se debe tener en cuenta que el lenguaje SQL es el único modo estandarizado de acceso a los datos en Bases de Datos Relacionales, por lo que es especialmente deseable que esté soportada la versión normalizada de dicho lenguaje.

También es muy importante que el SGBD proporcione los controladores y utilidades ODBC (Open Data Base Connectivity, Conectividad Abierta de Bases de Datos) necesarios para acceder a la base de datos por esta vía.

### **SQL SERVER 2000**

SQL Server 2000 es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con un interface intuitivo y con herramientas de desarrollo integradas como Visual Studio 6.0 , además podemos desarrollar cualquier aplicación que manipule componentes de SQL Server, es decir, realizar la aplicación para crear bases de datos, tablas, etc., y todo lo que se puede hacer desde el administrador del SQL Server y no solo en Visual C++ sino también en Visual Basic, y además en ASP .

Otras de las características que se pueden mencionar son las siguientes:

- Cuenta con una interfaz gráfica que reduce la complejidad de las tareas de administración y gestión de la base de datos.
- Está orientado al desarrollo: Visual Basic, Visual C++, Visual J++, y muchas

otras herramientas son compatibles con Microsoft SQL Server.

- Su plataforma de desarrollo es fácil y abierta, por estar integrada con tecnologías como: ActiveX, herramientas de gestión y desarrollo para Internet, y Microsoft Office.

#### **Desventaja**

En cuanto a su administración o una mala instalación, puede hacer que la base de datos mal creada o diseñada nos haga la vida imposible en nuestras aplicaciones ya que se puede tener un mal rendimiento; por lo que debemos tener cuidado y aprender a usarlo correctamente. También es importante el hardware, ya que es recomendable en el sistema de 256 MB o 512 MB para su buen funcionamiento y una cantidad suficiente de espacio en disco para que pueda trabajar con las bases de datos.

### **INFORMIX ONLINE 7.2**

On Line( En línea), se refiere a cualquier documento, archivo o servicio de la red. Informix ofrece varias opciones para conectar datos relacionales en páginas Web. En la cual proporciona un Web Interface Kit , que puede ser utilizado para establecer conexiones simples de Web a DBMS .

Una opción más sofisticada es Web Connectivity Framework, que permite a un Web site mantener información de estado de la sesión en la misma forma en que lo hace una aplicación de base de datos.

La estrategia seguida por Informix se basa en su alianza con Netscape, integrando los productos de Informix y Netscape con licencia para usuarios que desean utilizar el entorno Web pero no se puede utilizar el mismo DBMS para aplicaciones no-Web.

Algunas de sus características son las siguientes:

- Puede trabajar bajo la plataforma cliente/servidor mediante el producto de conectividad Informix Net.
- Opera bajo el sistema operativo UNIX, proporcionando alto desempeño y alta disponibilidad de información.
- Soporta bases de datos distribuidas mediante Informix Star y el manejador de bases de datos OnLine, mediante el cual se puede consultar y modificar varias bases de datos Informix en diferentes máquinas, como si fuera una sola base de datos centralizada.
- Cuenta con un conjunto de tablas para registrar el diccionario de datos.

### **ORACLE 9i**

Esta familia de productos soporta procesamiento online y permite establecer sesiones persistentes; en la cual se conserva el estado de interacción entre las pantallas del navegador, servidores Web y servidores de bases de datos con las bases de datos.

Para ello se mencionan las siguientes características:

- Se ejecuta sobre redes, software y hardware de cualquier otra base de datos relacional.
- Cuenta con opciones de seguridad para el manejador de la base de datos.

- Está diseñado para proveer facilidades de almacenamiento y recuperación de información de diferentes formatos.
- Es soportado por un gran número de lenguajes, con lo cual se pueden desarrollar de forma rápida aplicaciones que impliquen un gran volumen de información.

A continuación se presenta una tabla(1) comparativa entre las principales características de los manejadores de bases de datos analizadas anteriormente.

		<b>MANEJA DE BASES DE DATOS</b>		
		<b>SQL Server 2000</b>	<b>Informix Online 7.2</b>	<b>Oracle 9i</b>
<b>CARAC- TERÍS- TICAS</b>	<b>Vistas actualizables(con opción de verificación)</b>	Sí	Sí	Sí
	<b>Tipos de datos definidos por el usuario</b>	Sí	No	Sí
	<b>Gestión de Base de Datos</b>	Sí	Sí	Sí

**Tabla 1 Manejadores de Bases de Datos.**

*Considerando los resultados anteriores de la tabla podemos observar que SQL Server 2000 y Oracle 9i son útiles para la aplicación en el sistema, pero ya que en Servicios Administrativos Aduanales solo cuenta con solo el primero, en el desarrollo del sistema solo aplica para SQL Server 2000.*

- **Segundo** se mostrarán las opciones de Sistemas Operativos, no sin antes dar una introducción a los mismos.

### **Sistemas operativos**

La importancia de los sistemas operativos nace históricamente desde los 50's, cuando se hizo evidente que el operar una computadora por medio de tableros enchufables en la primera generación y luego por medio del trabajo en lote en la segunda generación se podía mejorar notoriamente, pues el operador realizaba siempre una secuencia de pasos repetitivos, lo cual es una de las características contempladas en la definición de lo que es un programa.

Es decir, se comenzó a ver que las tareas mismas del operador podían plasmarse en un programa, el cual a través del tiempo y por su enorme complejidad se le llamó "Sistema Operativo".

Así, tenemos entre los primeros sistemas operativos al Fortran Monitor System (FMS) e IBSYS [Tan92].

Posteriormente, en la tercera generación de computadoras nace uno de los primeros sistemas operativos con la filosofía de administrar una familia de computadoras: el OS/360 de IBM. Fue este un proyecto tan novedoso y ambicioso que enfrentó por primera vez una serie de problemas conflictivos debido a que anteriormente las computadoras eran creadas para dos propósitos en general: el comercial y el científico.

Así, al tratar de crear un solo sistema operativo para computadoras que podían dedicarse a un propósito, al otro o ambos, puso en evidencia la problemática del trabajo en equipos de análisis, diseño e implantación de sistemas grandes.

Surge también en la tercera generación de computadoras el concepto de la multiprogramación, porque debido al alto costo de las computadoras era necesario idear un esquema de trabajo que mantuviese a la unidad central de procesamiento más tiempo ocupada.

En la cuarta generación de las computadoras(en donde la electrónica avanza hacia la integración a gran escala, pudiendo crear circuitos con miles de transistores en un centímetro cuadrado de silicón y ya es posible hablar de las computadoras personales y las estaciones de trabajo), se hacen populares el MS-DOS y UNIX en estas máquinas.

Para mediados de los 80's, comienza el auge de las redes de computadoras y la necesidad de sistemas operativos en red y sistemas operativos distribuidos. La red mundial Internet se va haciendo accesible a toda clase de instituciones y se comienzan a dar muchas soluciones al querer hacer convivir recursos residentes en computadoras con sistemas operativos diferentes.

Para los 90's el paradigma de la programación orientada a objetos cobra auge, así como el manejo de objetos desde los sistemas operativos. Las aplicaciones intentan crearse para ser ejecutadas en una plataforma específica y poder ver sus resultados en la pantalla o monitor de otra diferente (por ejemplo, ejecutar una simulación en una máquina con UNIX y ver los resultados en otra con DOS ).

### **WINDOWS NT SERVER**

Lo que podemos mencionar acerca de Windows NT Server 4.0 es que al ser una buena opción servidor de Intranet, en la que las estaciones de trabajo ejecutan Microsoft Windows NT u otras versiones de Windows, utiliza las poderosas PCs basadas en Intel y las empresas pueden crear sistemas cliente/servidor a través de Windows NT.

La funcionalidad para Internet e Intranet se agregan con el Microsoft Internet Information Server, que requiere Windows NT Server.

Sus características generales como sistema operativo son:

- Windows NT Server es un sistema operativo de 32 bits, con lo que el rendimiento general ha mejorado mucho.
- Soporta multitarea con derecho preferente, es decir, en cualquier momento el sistema o el usuario puede interrumpir una tarea determinada. Así se garantiza que una tarea no bloquea al sistema si se ha quedado colgada y el tiempo de procesador se reparte entre todas las tareas que están ejecutando concurrentemente.



- Windows NT Server ofrece un sistema de protección de memoria, cada aplicación va a ser ejecutada en su zona de memoria asignada sin invadir zonas de memoria adyacentes.
- Además ofrece un modelo de memoria plana de 32 bits que permite un acceso a 4GB de memoria virtual.
- Es un sistema operativo multiplataforma, que es capaz de funcionar tanto en un ordenador normal como en un gran sistema.
- Windows NT Server es compatible con distintos sistemas operativos y aplicaciones.  
Diseña las aplicaciones como subsistemas del sistema operativo, cada nueva aplicación es un nuevo subsistema(Windows 3.x, Ms-Dos,etc).
- Ofrece la misma interfaz que Windows 95, por eso es fácil de usar, así como el mismo modelo de programación entre los clientes y servidores de Microsoft.
- Servidor de red distribuidos. Permite las comunicaciones dentro y fuera de la empresa. Microsoft ha combinado los servicios de red asociados a distintos sistemas operativos(como Novell) con los servicios necesarios para construir aplicaciones distribuidas, como aplicaciones Web.
- La administración es muy sencilla, los programas de administración ofrecen una interfaz gráfica al usuario que permite organizar los objetos fácilmente y de una manera jerárquica.

## UNIX

Ya que es un Sistema Operativo muy flexible hecho a la medida de las necesidades de los usuarios avanzados de computadoras y resultando apropiado para aplicaciones multiusuario, es necesario considerar las siguientes características:

- Incluye una gran variedad de programas de utilidad que pueden ser fácilmente adaptados para realizar tareas específicas. Estas utilerías son flexibles, adaptables, portables, y pueden ser usadas junto con filtros y redireccionamientos para hacerlos más poderosos.
- Permite proteger los archivos del usuario contra el acceso por parte de otros usuarios.
- Los dispositivos(como una impresora o una terminal) y los archivos en disco son considerados como archivos por UNIX. Cuando se dá una instrucción al UNIX puede indicársele que envíe el resultado a cualquiera de los diversos dispositivos o archivos. Está desviación recibe el nombre de redireccionamiento de la salida.
- La comunicación entre procesos: UNIX permite el uso de conductos y filtros en la línea de comandos.



Los sistemas operativos mencionados anteriormente que trabajan a su vez con los manejadores de bases de datos analizados en la primera etapa de opciones de solución se muestran en la tabla (2).

		MANEJADORES DE BASES DE DATOS		
		SQL SERVER	INFORMIX ONLINE	ORACLE 9i
SISTEMAS- OPERATIVOS	WINDOWS NT Server	Si	No	Si
	UNIX	No	Si	Si

Tabla 2. Sistemas operativos en los Manejadores de Bases de Datos

Para el sistema se considera factible usar el Sistema Operativo Windows Nt Server y no Unix, y con SQL Server como manejador de Bases de Datos ya que en Servicios Administrativos Aduanales solo se cuenta con ello.

- **Tercero** se analizarán las Herramientas para la creación de Páginas Web

#### **ASP(ACTIVE SERVER PAGES, PÁGINAS ACTIVAS EN EL SERVIDOR)**

Las ASP trabajan, en conjunción con lo que denomina un servidor activo, es decir un servidor con capacidad para procesar dichas páginas.

Una página ASP es texto que no necesita compilarse en ningún momento, ya que son interpretadas por el servidor activo correspondiente.

A continuación se mencionan las principales características de la tecnología ASP :

- Viene implementada por defecto en los servidores web PWS (Windows 95-98-Millennium) e IIS (Windows NT-2000-XP). Además también es posible implementarla en otros servidores, aunque hace falta la instalación de las extensiones adecuadas.
- Se puede mezclar código ASP con código HTML y código de script en cliente, sin tener que compilarlos por separado. Esto facilita la construcción de páginas ASP, siendo posible escribirlas en un simple editor de textos (Notepad, Word, etc.).
- El código ASP se ejecuta en el servidor web, produciendo como salida HTML puro (junto con código de script en el cliente, si así se precisa). Por este motivo, su resultado es interpretado correctamente por cualquier navegador web estándar, aunque se utilice VBScript, compatible tan sólo con Internet Explorer y los navegadores en él basados (aunque es posible ejecutar código VBScript en Netscape Navigator ).
- Permite usar componentes escritos en otros lenguajes de programación, como Visual Basic, Delphi, etc., que pueden ser llamados desde los propios guiones ASP.
- Se ha portado a la plataforma Java, lo que hace que pueda ser utilizado en casi cualquier sistema operativo.

- Permite acceder a las bases de datos de una forma rápida y sencilla, siendo éste uno de sus principales usos.
- Tiene persistencia de variables en memoria entre distintas visualizaciones de página, permitiendo con ello manejar sesiones de usuario, con lo que se resuelve uno de los mayores problemas de la programación web, derivado de la no orientación a conexión del protocolo HTTP.

### **JSP(JAVA SERVER PAGES,SERVIDOR DE PAGINAS JAVA)**

JSP(Página de Servidor Java) se refiere a un tipo especial de páginas HTML, en las cuales se insertan pequeños programas que corren sobre Internet (comúnmente denominados scripts), se procesan en línea para finalmente desplegar un resultado final al usuario en forma de HTML. Por lo general dichos programas hacen consultas a bases de datos y dependiendo del resultado que se despliegue será la información que se muestre a cada usuario de manera individual. Los archivos de este tipo llevan la extensión ".jsp".

En cuanto a sus características se tiene que:

- El Servidor Java de Páginas (Java Server Pages, JSP) ofrece no sólo la independencia de operar en diferentes plataformas y servidores de páginas Web, sino que además combina el poder de la tecnología Java en el servidor con la facilidad de visualizar el contenido de las páginas HTML.
- JSP es una tecnología que generalmente incluye: Componentes estáticos HTML/XML, Elementos JSP especiales, Fragmentos especiales de código escritos en lenguaje Java llamados scriptlets.  
Además, JSP permite apreciar mejor la distinción entre el contenido de la información y su presentación.
- Es un lenguaje interpretado completamente, intercalado directamente dentro de las páginas HTML.
- Un programa en JSP se basa en el modelo orientado a objetos.
- Soporta pocos datos, como los valores numéricos, booleanos y de cadena.
- Las sentencias de JSP pueden reconocer y responder a eventos de usuarios como las pulsaciones de ratón, la entrada de formularios y la navegación por páginas.

### **PHP (PROCESADOR DE HIPERTEXTO, HYPERTEXT PREPROCESSOR)**

Antes de analizar las características se mencionarán los antecedentes del PHP(Preprocessed Hypertext Pages).

PHP (acronimo de "PHP: Hypertext Preprocessor") es un lenguaje "open source" interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

Fué creado por Rasmus Lerdorf a finales de 1994 y aunque no hubo una versión utilizable por otros usuarios hasta principios de 1995. Esta primera versión se llamó, Personal Home PageTools. Al principio, PHP sólo estaba compuesto por algunas macros que facilitaban el trabajo a la hora de crear una página Web. Hacia mediados

de 1995 se creó el analizador sintáctico y se llamó PHP/F1 Versión 2, y sólo reconocía el texto HTML y algunas directivas de mSQL. A partir de este momento, la contribución al código fué pública. El crecimiento de PHP desde entonces ha sido exponencial, y han surgido versiones nuevas como la actual, PHP3 y PHP4.

Lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales, ya que el desarrollo de PHP está concentrado en la programación de scripts en la parte del servidor, así como también para procesar la información de formularios, generar páginas con contenidos dinámicos, etc.

Además, de las características relevantes:

- Puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, incluyendo variantes de Unix como: HP-UX, Solaris.
- Soporta la mayoría de los servidores Web de hoy en día, incluyendo Apache, Netscape, Personal Web Server.
- Se puede procesar la información de formularios, generar páginas con contenidos dinámicos.
- Se tiene la posibilidad de usar programación de procedimientos o programación orientada a objetos.
- Entre las habilidades de PHP se incluyen la creación de imágenes, ficheros PDF y películas Flash.
- Es un soporte para gran cantidad de bases de datos. Escribir una interfaz vía Web para una base de datos es una tarea simple con PHP. Algunas bases de datos que son soportadas por PHP son: Informix, FrontBase, mSQL, MySQL, Oracle.

*En base a las Herramientas analizadas en la tercera etapa, la información se manejará en un ambiente de Bases de Datos Relacional operando en red en un sistema de consulta bajo páginas activas web donde el personal pueda tener acceso a la información de forma ordenada, rápida. Para ello en el presente diseño del sistema se necesita:*

- *Un sistema Windows NT 4.0 con Internet Information Server 3.0 en funcionamiento.*
- *Un sistema de bases de datos (ya que va a confeccionar páginas ASP que accedan a datos). Éste puede ser el propio Access o cualquier otro al que podamos acceder por ODBC(conjunto de funciones que permite acceder a prácticamente cualquier servidor de bases de datos mediante un driver sin tener que cambiar el programa).*

## **2.2 Estructura Organizacional de las Áreas involucradas**

Servicios Administrativos Aduanales está integrada por 9 administraciones ubicadas en los estados fronterizos de la República Mexicana(Baja California Norte, Sonora, Chihuahua, Coahuila, Nuevo León, Tamaulipas, Chiapas, Tabasco, Campeche-Quintana Roo) y un personal que oscila entre 90 y 150 usuarios en cada una de ellas,

laborando en cualquiera de los tres horarios que tienen: 8:00 a 15:00 hrs, 9:00 a 18:00 hrs o 9:00 a 20:00 hrs realizando determinadas actividades de acuerdo a su función y su área a la que pertenece.

La estructura organizacional mostrada en la Figura 2 es para: Servicios Administrativos Aduanales de Baja California Norte ; ya que para las demás administraciones se procede de manera similar.

Servicios Administrativos Aduanales de Baja California Norte está integrada por tres Subadministraciones y un Departamento, los cuales son: Subadministración de Servicios Administrativos, Subadministración de Recursos Humanos, Subadministración de Programación y Presupuesto; y un Departamento de Informática, cada una con las siguientes funciones:

- **Subadministración de Servicios Administrativos:** lugar donde se realizan los trámites referentes al personal de Servicios Administrativos Aduanales .
- **Subadministración de Recursos Humanos:** que se encarga de administrar los bienes materiales utilizados en las diferentes administraciones que conforman a Servicios Administrativos Aduanales .
- **Subadministración de Programación y Presupuesto:** es en donde se controlan los trámites relacionados con gastos e ingresos para Servicios Administrativos Aduanales .
- **Departamento de Informática:** área que colabora con las administraciones anteriores para ayudar al personal a utilizar sistemas informáticos.

#### SERVICIOS ADMINISTRATIVOS ADUANALES



Figura 2. Organigrama de Servicios Administrativos Aduanales

### 2.3 Documento de información de funciones actuales

Es en el área de Subadministración de Recursos Humanos donde se controlan los asuntos relacionados con el personal administrativo, por medio de lo que denominan las incidencias de los Trabajadores. Estas pueden ser de diversos tipos y se tienen catalogados como: Vacaciones, Faltas, Licencias y Artículo 50 del personal que integra Servicios Administrativos Aduanales, generadas en cualquier lapso de tiempo en el cual labore determinada persona. La forma en que interactúa el usuario con el administrador del área correspondiente depende del tipo de incidencia que se reporte y tiene que tratarlo personalmente para después realizar determinados trámites administrativos correspondientes.

Para realizar dichos trámites se procede de la siguiente manera:

Primero las personas encargadas de realizar los trámites de Incidencias solicitan información por escrito (Nombre de la Incidencia, fecha en la cual se elabora la Incidencia, datos personales del trabajador; así como el nombre y cargo de su jefe inmediato de la persona a la cual se le aplicará una determinada Incidencia); con la finalidad de llevar a cabo un control en su departamento .

Después el jefe inmediato entrega al encargado del archivo en el área de Incidencias la solicitud mediante su firma. Se captura el tipo de incidencia en una hoja de Excel y finalmente se reporta tanto a la persona acreedora de la Incidencia como a su jefe inmediato el documento con el reporte que se generó con un número identificador que es utilizado para el archivo.

## 2.4 Estándares de Programación

Para poder llevar a cabo un mejor análisis en el modelado y diseño del sistema se requiere considerar los estándares aplicables en el mismo.

Por ello se dan a continuación características y ventajas en la utilización de los estándares con la finalidad de Introducir términos retomados a lo largo del código para el sistema.

El enfoque estándar se aplica desde la forma en que se nombra a las variables hasta el tipo de comentarios que se ponen en el código.

Visual Basic Script proporciona código fuente de los programas ayudando a comprenderlo y compartirlo ahora a través de los propios programas y a través del mundo.

Microsoft proporciona lineamientos de estándares convencionales para el VBScript; los cuales están documentados en World Wide Web. Sin embargo ninguna ley universal exige que todo programador siga los estándares de Microsoft.

Algunas de las ventajas de los estándares son:

- Los estándares, permiten que se produzcan mejores programas.
- Si los programadores están intercambiando código y usan los mismos estándares es mucho más fácil tener un registro del código, debido a que los buenos estándares hacen que el código sea fácil de leer.
- Cuando se aplican estándares mientras se escriben scripts <sup>1</sup>, encontrará que llega a ser más fácil su escritura.
- El uso de estándares proporcionan que la depuración de los programas sea más fácil; ya que se necesitará revisar variables, rastrear a través de procedimientos <sup>2</sup> y dar un vistazo general a lo que está haciendo el código. En todas estas actividades ciertamente ayudan buenos nombres de variables, buena estructura de script y buenos comentarios.
- Se deben seguir estándares desde el inicio, debido a que rinden beneficios desde el momento en que se crean script.

Para considerar los estándares a continuación se muestran las características más importantes de los mismos:

<sup>1</sup> Secuencia de instrucciones que un programa es capaz de seguir, interpretar y ejecutar.

<sup>2</sup> Secuencias de comandos que se pueden ejecutar en cualquier momento haciendo referencia a su nombre. En Visual Basic Script hay dos tipos de procedimientos: Sub y Function.

**Variables :** Forman la columna vertebral de la mayoría de los programas. En Visual Basic Script hay dos formas de declarar variables: la primera es explícitamente con la ayuda de la palabra reservada Dim, y la segunda forma es inicializar una variable es utilizarla directamente.

Si no se usan las variables correctamente, los resultados pueden ser devastadores.

Para la primera forma:

Una de las formas más fáciles para que sea claro el uso de una variable es darle un nombre significativo.

Por ejemplo **Dim RFC**

Donde la declaración puede decir que la variable con el nombre de RFC puede contener el RFC de una persona y no un número ni ninguna otra cosa. Incluso se podría decir más acerca de la variable cuando se mantiene el código fuente.

Para la segunda forma:

```
<script language="vbscript">
<!--
Dim RFC
RFC="MAM790219"
</script >
```

En donde primero se declara la variable RFC, y después se crea la variable RFC="MAM790219" en otra línea de código para posteriormente realizar la ejecución de la aplicación.

**Prefijos de nombre de variable :** Ya que el nombre descriptivo no le dice todo acerca de una variable, es necesario por convención poner un prefijo de tipo de dato específico, por ejemplo **int** al inicio de variables enteras o **str** al inicio de variables de cadena, solo si las variables se usan para este propósito.

**Prefijos de alcance:** Es un tipo más de prefijo especial de variable. Si se declara una variable inmediatamente al principio del script, sin ponerla dentro de un procedimiento se puede hacer referencia a esa variable desde cualquier lugar del script. Se dice que tiene alcance global o a nivel script. Una buena manera de asignar variables globales es solo usarlas para la información que es importante compartir entre procedimientos. Debido a la convención de denominación de alcance a nivel script se declara en forma como:

```
<script language="vbscript">
<!--
Dim s_RFC
```

Donde:

<script language="vbscript">; es la línea de código que al encontrarla el explorador empezará a tratar el resto de las líneas como código Visual Basic Script; interpretando su contenido en tiempo de ejecución.

Y la línea : Dim s\_RFC es una variable global; ya que al preceder la s\_ al nombre de la variable; se dice que es de alcance global.

**Comentarios:** Los buenos comentarios son una parte importante de los estándares de scripts y de la programación en general. El delimitador de comentario es el carácter de comillas simple o apóstrofo('). Y todo el texto que se

encuentra a continuación de éste símbolo en una línea se trata estrictamente como comentario.

**Constantes:** Son nombres simbólicos para valores que nunca cambian dentro del código. Las constantes son variables a las que solo podemos dar valor una vez. Cuando se lo hayamos dado, quedará ligado a la constante para el resto del programa. Para declarar una constante, se utiliza la palabra reservada Const de la siguiente forma:

**CONST nombre\_constante**

Una vez declarada la constante, le podremos dar un valor solamente una vez en toda la aplicación. Para asignarle ese valor se utilizará la sintaxis siguiente:

**nombre\_constante= valor**

**Objetos:** Son un conjunto de propiedades y métodos que tomamos como una unidad y que están definidos para ayudarnos a realizar ciertas operaciones que se suelen repetir bastantes veces a lo largo de una aplicación cualquiera.

**Declaraciones de Procedimiento:** Al tener una documentación clara y un enfoque consistente en las convenciones de procedimientos, también se hacen importantes:

- Los nombres descriptivos para los procedimientos: Los nombres\_descriptivos son el primer paso para un enfoque bueno y consistente. Debido a que los procedimientos pueden realizar alguna acción, los nombres descriptivos deben comenzar con una palabra de acción (por ejemplo Calcular PDC, donde PDC significa el promedio de calificación); los nombres de procedimiento, deben aparecer con una combinación de mayúsculas y minúsculas para mejorar la legibilidad de las palabras que componen el nombre.
- Los comentarios para procedimientos: Una declaración de procedimiento que tiene al menos una línea de comentario en la cual resume con una frase corta el propósito de ese procedimiento. Además de ser importante el documentar el valor de retorno para condiciones de error, así como las condiciones normales, cuando sea adecuado.

## 2.4.1 Estándares existentes aplicables al proyecto

Los estándares que se aplican al proyecto en Servicios Administrativos Aduanales parten de los estándares convencionales para VBScript y tienen la información que debe aparecer en los comentarios para todo procedimiento; así como para el departamento y el programador responsable de la elaboración de ese procedimiento.

Para ello se han considerado los estándares anteriormente mencionados y una parte de los estándares se ha personalizado, de tal forma que puedan hacer más eficiente y comprensible el código en el momento de introducir el código del sistema.

Además, algunas de las técnicas utilizada para hacer el código más legible fueron:

- El uso la sangría adecuada: Consistiendo en dar el espaciado sobre las instrucciones que están subordinadas en forma lógica a una instrucción de nivel superior.

Por ejemplo en una condición If...then, las instrucciones que se encuentran dentro de la condición se ejecutan únicamente si la condición es cierta. Dando una sangría de dos a cuatro espacios a esas instrucciones indica que la instrucción If siempre se evalúe y que las instrucciones con sangría sólo se ejecutan bajo determinadas condiciones. Y los comentarios que siguen a



continuación de una declaración de procedimiento, por lo general, tienen una sangría de un espacio con respecto al nombre del procedimiento.

- Se configuraron herramientas de desarrollo( como Control Pad Editor de Microsoft) para de forma personalizada resaltar de un color dado las palabras clave de Visual Basic Script. Estas herramientas van un paso más allá de los estándares debido a que no es una acción que se pueda controlar en la manera de escribir de programas. Sin embargo los estándares y las herramientas para la visión y generación de código están orientados a facilitar y hacer más eficiente el mantenimiento del código.
- Comentarios: para el diseño del presente sistema se incluyen comentarios relativos a los scripts específicos disminuyendo así los comentarios excesivos en el interior del script sin sacrificar las ventajas de la documentación.

## 2.5 Hardware

El hardware especifica una característica física que debe poseer el sistema, tales como: la computadora físicamente, sus periféricos y las configuraciones físicas de red requeridas.

A continuación se presenta de forma detallada los requerimientos de cada uno de los equipos con los cuales se cuenta y nos ayudan a utilizar el sistema.

### Hardware para servidores:

Es la parte más importante y crítica de una red. La función del servidor es la de compartir sus recursos a la red, ya sean impresión, de disco o aplicaciones cliente/servidor.

Los requerimientos mínimos de hardware para el servidor está enfocado a la ejecución de servicios básicos de red(impresión, archivos, software institucional) y proyectado para dar servicio al interior de Servicios Administrativos Aduanales. Dichos requerimientos son:

Procesador Pentium III(550 MHz)  
 Memoria RAM 128 MB  
 Disco Duro 20 GB  
 Mouse(bus)  
 Monitor SVGA 14"

El hardware para el servidor con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

Procesador Pentium 4 a 1.7 MHz  
 Disco Duro 40 GBb  
 Memoria RAM 256MB  
 Mouse(bus)  
 Monitor SVGA 14"

### Hardware para clientes:

Los requerimientos mínimos de hardware para el cliente está enfocado a la ejecución de la aplicación desarrollada, en conjunto con todo lo que se requiere para su buen funcionamiento. Dichos requerimientos son:

Procesador mayor a 140MHz  
 Memoria RAM de al menos128 MB  
 Disco Duro 20 GB  
 Mouse(bus)  
 Monitor SVGA 14"  
 Conexión a la red



El hardware para el cliente con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

- Procesador Pentium 4 a 1.7 MHz
- Disco Duro 40 GBb
- Memoria RAM 256MB
- Mouse(bus)
- Monitor SVGA 14"

#### **Hardware para usuarios:**

Los requerimientos mínimos de hardware para el usuario está enfocado a la ejecución de la aplicación desarrollada. Dichos requerimientos son:

- Procesador Pentium II (250MHz)
- Disco Duro 100 Mb libres
- Memoria RAM 64MB
- Mouse(bus)
- Monitor SVGA 14"

El hardware para el usuario con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

- Procesador Pentium 4 a 1.7 MHz
- Disco Duro 40 GBb
- Memoria RAM 256MB
- Mouse(bus)
- Monitor SVGA 14"

## **2.6 Software**

El software son las órdenes, las instrucciones y los comandos estructurados en archivos, directorios y programas.

Con el propósito de unificar la información los diseños de las páginas se desarrollarán únicamente con el software institucional, que a continuación se detalla.

#### **Software para servidores:**

El software para el servidor con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

- Sistema Operativo Windows 2000 Advanced Server
- Manejador de Base de Datos Sql Server
- Acrobat
- Winzip
- Internet Information Server 4.0 (exclusivamente para la publicación)
- SQL- Server

#### **Software para clientes:**

El software para el cliente con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

- Sistema Operativo Windows 2000
- Acrobat
- Winzip
- Internet Information Server 4.0 (exclusivamente para la publicación)
- SQL- Server

#### **Software para usuarios:**

El software para el usuario con el que se cuenta en Servicios Administrativos Aduanales es el siguiente:

- Sistema Operativo Windows 2000
- Office 97 y sus actualizaciones
- Acrobat
- Winzip
- Internet Explorer 4.0 (Navegador)

### **III. DISEÑO DEL SISTEMA**

### III.- DISEÑO DEL SISTEMA

El diseño de un sistema de información produce los elementos que establecen cómo el sistema de bases de datos cumplirá los requerimientos identificados durante el análisis del sistema.

Es en esta parte del sistema en donde se describen los datos que se introducirán, se trabaja con los resultados del análisis haciendo las revisiones pertinentes hasta que el diseño se haya completado y así definir como va a funcionar el sistema.

En el diseño se modela el sistema y se considera la forma para que soporte todos los requisitos.

Una entrada esencial en el diseño es el resultado del análisis, esto es, el modelo conceptual. El modelo conceptual proporciona una comprensión detallada de los requisitos.

Y lo que es más importante, impone una estructura del sistema que debemos esforzarnos por conservar lo más fielmente posible cuando demos forma al sistema.

El Diseño consta de los siguientes elementos:

Diagrama de Casos de Uso: El diagrama de casos de uso representa las formas en las cuales los actores(cada tipo de usuario que interactúa en el sistema) realizan sus funciones en el sistema.

Diagrama de Flujo de Datos: Este diagrama representa el flujo de la información y de las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida; se observa a través de niveles, que son vistos por usuarios de diferentes niveles de abstracción, por lo que los usuarios deben entender el significado de todos los tipos de datos, para ello nos ayudamos de la disponibilidad de un Diccionario de Datos, que es un depósito integrado de los tipos de datos producidos, controlados, intercambiados y mantenidos.

Diccionario de Datos: Ya que los sistemas de información pueden ser complejos y se caracterizan por la heterogeneidad, al almacenar algunos diferentes tipos de datos, cada uno de ellos con su propio modelo pueden ser vistos por usuarios de diferentes niveles de abstracción, por lo que los usuarios deben entender el significado de todos los tipos de datos, para ello nos ayudamos de la disponibilidad de un Diccionario de Datos, que es un depósito integrado de los tipos de datos producidos, controlados, intercambiados y mantenidos.

Diagrama Entidad Relación(ER): El diagrama ER es parte de las herramientas conceptuales que existen para describir datos y las relaciones que existen entre ellos. El diagrama ER está basado en una percepción del mundo, como si se tratara de una colección de objetos básicos(entidades), representados por una colección finita de tablas de dos dimensiones(columnas y renglones) y las relaciones que existen entre estos objetos, las cuales representan acciones o situaciones reales. El diagrama ER se caracteriza por su simplicidad, precisión y flexibilidad.

El diagrama ER se compone principalmente de : entidades, relaciones y atributos.

Las **entidades** hacen referencia a un objeto que es distinguible a través de un rectángulo.

Una **relación** se refiere a la relación de dos o más entidades, puede tener atributos, permite expresar la cantidad de entidades con las que puede asociarse otra entidad. Las relaciones de acuerdo al número de elementos

que se involucran de ambos lados de las entidades que participan se clasifican en tres: **Uno a Uno(1,1)**; si la cardinalidad de una relación es de una entidad en ambas direcciones, **Uno a Muchos(1-M)**; si la cardinalidad es de una a una dirección y de muchas a una y **Muchos a Muchos(M-M)**; si la cardinalidad es de muchas a muchas en ambas direcciones.

Los **atributos** son funciones con rango de valores que permiten identificar a un conjunto de entidades. Para colocar los atributos de manera correcta en una entidad se utiliza el método de normalización que permite verificar si nuestro modelo es correcto o bien si funcionará al ser implementado.

Normalización: Es un proceso que consiste en comprobar que las tablas definidas cumplen unas determinadas condiciones. Se pretende garantizar la no existencia de redundancia y una cierta coherencia en la representación mediante un esquema relacional de las entidades y relaciones del diagrama ER. Mediante la normalización se pueden solucionar diversos errores en el diseño de la base de datos así como mejorarlo. También se facilita el trabajo posterior del administrador de la base de datos y de los desarrolladores de aplicaciones. En este proceso, las relaciones deben cumplir con las Formas Normales, que serán descritas más a detalle en este capítulo.

### 3.1 Modelado del sistema

Para realizar un método <sup>III</sup> sobre los requisitos, condiciones y posibilidades que debe cumplir el sistema es necesario elaborar un modelo<sup>IV</sup>; ya que al modelar:

- Nos ayuda a visualizar como es o que queremos que sea el sistema.
- Nos permite especificar la estructura o el comportamiento de un sistema.
- Nos documentan las decisiones que hemos logrado.

Ya que el sistema de control de Incidencias consta de una Base de Datos y una interfaz gráfica, mismos que hacen de la información un conjunto consistente para poder visualizar, modificar y realizar búsquedas; fue conveniente hacer un modelo de casos de uso. Este modelo describe lo que hace el sistema para cada tipo de usuario; contiene actores, y sus relaciones.

Para el modelado del sistema muchas veces se realizan diagramas. Para el diseño del sistema se utilizaron diagramas de casos de uso y diagramas de flujo de datos; contando como apoyo con las herramientas de UML(Unified Modeling Language, Lenguaje Unificado de Modelado) para dicho fin y considerando para ello tratarlo a continuación.

### 3.2 Diagrama de Casos de Uso

Un caso de uso es un documento narrativo que describe la secuencia de eventos de un actor que utiliza un sistema para completar un proceso.

<sup>III</sup> Proceso disciplinado para generar un conjunto de modelos que describe varios aspectos de un sistema de software en desarrollo, usando alguna notación bien definida.

<sup>IV</sup> Simplificación de la realidad para comprender mejor el sistema que se está creando.

Un actor es una identidad externa del sistema que de alguna manera participa en la historia del caso de uso, por lo regular estimula el sistema con eventos de entrada o recibe algo de él. Los actores están representados por el papel que desempeñan en el caso.

Un diagrama de casos de uso explican gráficamente un conjunto de casos que se utilizan en un sistema, los actores y las relaciones entre estos.

Los casos de uso como se muestran en la Figura 3 se representan en óvalos y los actores son figuras utilizadas. También hay líneas de comunicación entre los casos de uso y los actores; ellas se utilizan con flechas indicando el flujo de información o estímulo.



Figura 3. Notación para un caso de uso y actor

Al analizar con Servicios Administrativos Aduanales los requerimientos del sistema se definió una clasificación de tres niveles de usuario:

1. El Administrador del Sistema, 2. El Administrador de la Base de Datos y 3 El Usuario del Sistema con las funciones que a continuación se mencionan .

El Administrador del Sistema tiene permisos de acceso a todas las pantallas de administración; puede administrar los archivos de cualquiera de las administraciones para crear, borrar o modificar datos. Es responsable de administrar el catálogo de los usuarios.

El Administrador de la Base de Datos puede acceder a la parte de administración de archivos particularmente de las administraciones a las que pertenecen determinados usuarios para crear, borrar o modificar archivos.

El Usuario del Sistema, podrá acceder a aquellos archivos a los cuales se le haya otorgado permiso en la administración de archivos; también podrá solicitar alta, baja y actualización de los mismos.

El actor que encontramos en el sistema es el Administrador del Sistema y sus casos de uso se observan en la Figura 4(pag 36).

#### 1.- Caso de uso Consultar archivos de Incidencias

El Administrador de la Base de Datos consulta la lista de archivos de Incidencias.

El Administrador de la Base de Datos accede a los archivos de Incidencias.

El Administrador de las Base de Datos permite el acceso a los archivos de Incidencia solicitados.

#### 2.- Caso de uso Solicitar alta de archivos de Incidencias

El Administrador de la Base de Datos atiende la solicitud de alta del archivo de Incidencias que desea dar de alta en el sistema por parte del usuario.

El Administrador de la Base de Datos da de alta en el archivo de Incidencias asignando los permisos correspondientes de acceso al archivo de Incidencias.

El sistema permite al Administrador de la Base de Datos acceder al archivo de Incidencias dado de alta.

**3.- Caso de Uso Solicitar baja de archivos de Incidencias**

El Administrador de la Base de Datos atiende la solicitud de baja del archivo de Incidencias por parte del usuario.

El Administrador de la Base de Datos da de baja en el sistema el archivo de Incidencias.

El sistema borra el registro del archivo de Incidencias.

**4.- Caso de Uso Actualizar archivos de Incidencias**

El Administrador de la Base de Datos atiende la solicitud de modificación del archivo de Incidencias por parte del usuario.

El Administrador de la Base de Datos modifica en el sistema el archivo de Incidencias asignando los permisos de acuerdo a la solicitud.

El sistema le permite al Administrador de la Base de Datos el acceso al archivo de Incidencias modificado.

**5.- Caso de Uso Dar de alta usuarios**

El Administrador del Sistema atiende la solicitud de alta de un usuario por parte del Administrador de la Base de Datos.

El Administrador del Sistema da de alta al usuario asignando la categoría de usuario y el área correspondientes, como datos principales.

El Sistema da de alta al nuevo usuario en la Base de Datos.

**6.- Caso de Uso Dar de Baja a Usuarios**

El Administrador del Sistema atiende la solicitud de baja de un usuario por parte del Administrador de la Base de Datos.

El Administrador del Sistema da de baja al usuario en el sistema.

El Sistema da de baja al usuario en la Base de Datos.

**7.- Caso de Uso Modificar Usuarios**

El Administrador del Sistema atiende la solicitud de modificación de un usuario por parte del Administrador de la Base de Datos.

El Administrador del Sistema modifica el usuario asignando la categoría de usuario y las administraciones correspondientes, como datos principales.

El Sistema modifica las propiedades del usuario en la Base de Datos.



**Figura 4. Diagrama de Casos de Uso del actor Administrador del Sistema**

Otro actor que encontramos en el sistema es el Administrador de la Base de Datos sus casos de uso se observan en la Figura 5(Página 37).

El caso de uso del actor Administrador de la Base de Datos se describen paso a paso con algunas frases que resumen las acciones que el sistema necesita hacer cuando interactúa con el Sistema.

#### 1.- Caso de uso Consultar archivos de Incidencias

El Administrador de la Base de Datos consulta la lista de archivos de Incidencias.

El Administrador de la Base de Datos accede a los archivos de Incidencias.

El Administrador de las Base de Datos permite el acceso a los archivos de Incidencia solicitados.

#### 2.- Caso de uso Solicitar alta de archivos de Incidencias

El Administrador de la Base de Datos atiende la solicitud de alta del archivo de Incidencias que desea dar de alta en el sistema por parte del usuario.

El Administrador de la Base de Datos da de alta en el archivo de Incidencias asignando los permisos correspondientes de acceso al archivo de Incidencias.

El sistema permite al Administrador de la Base de Datos acceder al archivo de Incidencias dado de alta.

#### 3.- Caso de Uso Solicitar baja de archivos de Incidencias

El Administrador de la Base de Datos atiende la solicitud de baja del archivo de Incidencias por parte del usuario.

El Administrador de la Base de Datos da de baja en el sistema el archivo de Incidencias.

El sistema borra el registro del archivo de Incidencias.

#### 4.- Caso de Uso Actualizar archivos de Incidencias

El Administrador de la Base de Datos atiende la solicitud de modificación del archivo de Incidencias por parte del usuario.

El Administrador de la Base de Datos modifica en el sistema el archivo de Incidencias asignando los permisos de acuerdo a la solicitud.

El sistema le permite al Administrador de la Base de Datos el acceso al archivo de Incidencias modificado.

#### 5.- Caso de Uso Solicitar alta de usuarios

El Administrador de la Base de Datos solicita al Administrador del Sistema el alta de archivo de Incidencias proporcionando los datos correspondientes.

#### 6.- Solicitar baja de usuarios

El Administrador de la Base de Datos solicita al Administrador del Sistema la baja de archivo de Incidencias proporcionando los datos correspondientes.

#### 7.- Caso de Uso solicitar Actualización de usuarios

El Administrador de la Base de Datos solicita al Administrador del Sistema la actualización de archivo de Incidencias proporcionando los datos correspondientes.

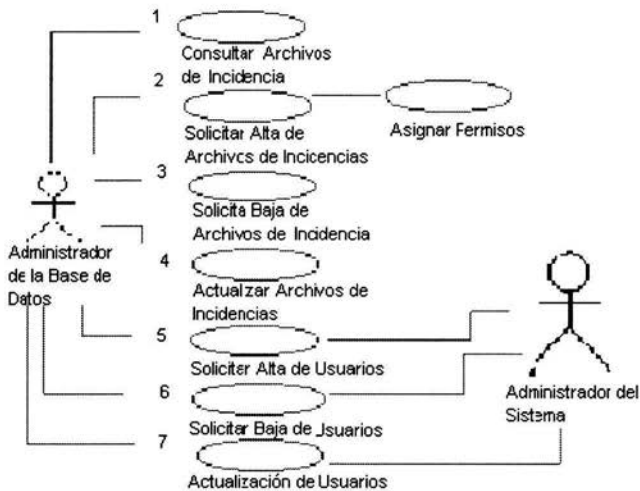


Figura 5. Diagrama de Casos de Uso del actor Administrador de la Base de Datos

Una vez que se identificaron los casos de uso para los Administradores, ahora se describe el caso de uso del actor Usuario con algunas frases que resumen las acciones que el sistema necesita hacer cuando interactúa con el actor Administrador del Sistema.

En el diagrama de la Figura 6 (Página 38) tenemos el caso de uso del actor usuario.

#### 1.- Caso de uso Consultar archivos de Incidencias

El usuario consulta la lista de archivos de Incidencias.

El usuario accede a los archivos de Incidencias.

El sistema permite el acceso a los archivos de Incidencia solicitados.



## 2.- Caso de uso Solicitar alta de archivos de Incidencias

El usuario identifica el archivo de Incidencias que desea dar de alta en el sistema.

El usuario solicita al Administrador de la Bases de Datos el alta del archivo proporcionando los permisos correspondientes de acceso al archivo de Incidencias.

El administrador de las Bases de Datos da de alta en el sistema el archivo de Incidencias solicitado incluyendo los permisos correspondientes.

## 3.- Caso de Uso Solicitar baja de archivos de Incidencias

El usuario identifica el archivo de Incidencias que desea dar de baja en el sistema.

El usuario solicita al Administrador de la Base de Datos la baja del archivo proporcionando los permisos correspondientes de acceso al archivo de Incidencias.

El administrador de la Base de Datos da de baja en el sistema el archivo de Incidencias solicitado incluyendo los permisos correspondientes.

El sistema borra el registro del archivo

## 4.- Caso de Uso Actualizar archivos de Incidencias

El Administrador de la Base de Datos atiende la solicitud de modificación del archivo de Incidencias por parte del usuario.

El Administrador de la Base de Datos modifica en el sistema el archivo de Incidencias asignando los permisos de acuerdo a la solicitud.

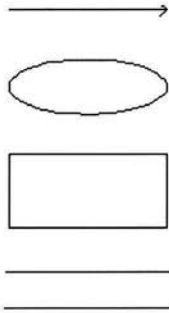


Figura 6. Diagrama de Casos de Uso del actor Usuario

## 3.3 Diagrama de Flujo de Datos

Los diagramas son representaciones gráficas de un conjunto de elementos que la mayoría de las veces se dibujan como grafo conexo de nodos(elementos) y arcos(relaciones). Los diagramas se utilizan para visualizar un sistema desde diferentes perspectivas, ya que son herramientas muy frecuentes utilizadas para el análisis estructurado y el manejo de los datos de una forma eficiente.

La simbología utilizada en los diagramas de flujo de datos se ilustra en la Tabla 4.

**OBJETO****REPRESENTA DEFINICIÓN**

|                              |  |
|------------------------------|--|
| Flujo De Datos               | La punta indica la dirección del flujo de datos  |
| Procesos                     | Denotan un cambio o transformación de los datos  |
| Entidades                    | Personas, documentos o sistemas de cómputo que pueden enviar o recibir datos del sistema |
| Fuentes o depósitos de datos | Representa el almacenamiento de datos  |

Tabla 4 .Objetos de un diagrama de flujo de datos

Ahora se mostrarán los procesos: Agregar Archivos de Incidencia, Eliminar Archivos de Incidencia, Modificar Archivos de Incidencia, Alta de Usuarios, y Acceso de Usuarios, por medio de diagramas de flujo

En la Figura 7. Se muestra el Diagrama de Flujo de Proceso Agregar Archivos de Incidencia. Éste abarca desde la solicitud por parte del usuario al Administrador de Base de Datos de dar de alta un archivo, hasta la consulta del archivo en el sistema por parte del usuario.



Figura 7. Diagrama de Flujo de Proceso Agregar Archivos de Incidencia

Al solicitar Agregar un archivo de Incidencia el nombre de alta de Incidencia se da de alta en la Base de datos.

Una vez que el archivo ha sido agregado a la Base de datos podrá ser consultado.

En la Figura 8. Se muestra el Diagrama de Flujo de Proceso Eliminar Archivos de Incidencia. Éste abarca desde la solicitud por parte del usuario al Administrador de Base de Datos de dar de baja un archivo, hasta la no presentación del archivo en el sistema por parte del usuario.

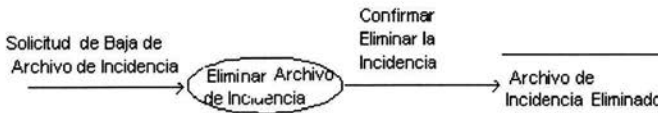


Figura 8. Diagrama de Flujo de Proceso Eliminar Archivos de Incidencia

Como se puede observar en el diagrama, al solicitar Eliminar un archivo de Incidencia en el sistema, se le proporciona el nombre de Incidencia que se quiere Eliminar y el sistema hace la búsqueda del archivo en la Base de Datos, al encontrarlo lo borra de la base de Datos y ya no lo presenta en la lista.

En la Figura 9. Se muestra el Diagrama de Flujo de Modificar Archivos de Incidencia. Éste abarca desde la solicitud por parte del usuario al Administrador de Base de Datos de Modificar un archivo de Incidencia, hasta la modificación del archivo en el sistema por parte del usuario.



Figura 9. Diagrama de Flujo de Proceso Modificar Archivos de Incidencia

Como se muestra en el diagrama, al solicitar Modificar un archivo de Incidencia en el sistema, se le proporciona el nombre de Incidencia que se quiere Modificar y el sistema hace la búsqueda del archivo en la Base de Datos, al encontrarlo lo borra de la base de Datos y ya no lo presenta en la lista.

En la Figura 10. Se muestra el Proceso Alta de Usuarios. Éste abarca desde la solicitud por parte del Administrador del Sistema de dar de alta un usuario, hasta la confirmación de que el usuario está en el sistema.



Figura 10. Diagrama de Flujo de Proceso Alta de Usuarios

Para el diagrama anterior, al solicitar el Alta de un usuario en el sistema, se proporcionan los datos del usuario, tales como: nombre, apellido paterno y materno, fecha de alta, departamento al que pertenece, puesto, extensión, correo electrónico y se le asigna una contraseña; determinando el tipo de usuario al que pertenecerá, recordando que puede ser: Administrador de Base de Datos (sin alguna restricción para consultar archivos de cualquier departamento), Administrador del Sistema (puede consultar archivos de algún departamento), Usuario (solo puede consultar archivos que le hayan sido asignados).

Y una vez proporcionados éstos datos se agrega al usuario a la base de datos y se confirma que haya sido dado de alta.

En la Figura 11 Se muestra el Diagrama de Flujo de Proceso Acceso de Usuarios. Éste abarca desde el ingreso de nombre del usuario y contraseña, hasta la consulta de archivos de Incidencias en el sistema.

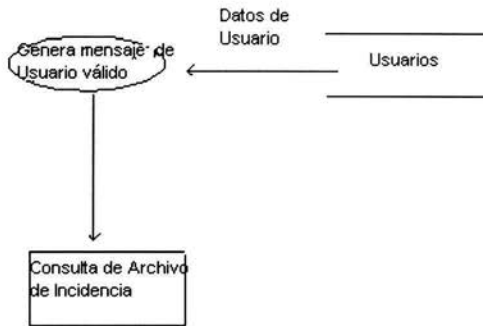


Figura 11. Diagrama de Flujo de Proceso Acceso de Usuarios

Como se puede observar en el diagrama, el usuario al ingresar su nombre de usuario y contraseña, el sistema verifica que sea un usuario válido, es decir que se encuentre dado de alta en la Base de Datos si es un usuario no válido, el sistema envía un mensaje de error haciéndole saber al usuario que no se encuentra en la Base de Datos; si se trata de un usuario válido dependiendo de sus permisos, es decir del tipo de usuario al que pertenece podrá consultar los archivos a los que tengan acceso.

### 3.4 Diccionario de Datos

El diccionario de datos hace referencia a las tablas que van a integrar la base de datos de nuestro sistema y nos indica los campos que tendrán cada una de ellas, así como el tipo de dato que aceptará, la longitud del dato y una breve descripción del campo, como se muestra a continuación:

Tabla : Incidencias.

La tabla de incidencias nos permite construir la tabla principal de datos de incidencia del personal.

| <b>NOMBRE DE COLUMNA</b> | <b>TIPO DE DATOS</b> | <b>LONGITUD DEL DATO</b> | <b>DESCRIPCION</b>  |
|--------------------------|----------------------|--------------------------|---|
| <b>Folio</b>             | <b>Numeric</b>       | <b>9</b>                 | Representa el identificador de registros, y es el número con el cual las incidencias se capturan                    |
| <b>Rfc</b>               | <b>Nvarchar</b>      | <b>12</b>                | Registro Federal de Causantes.  |
| <b>Cve_inci</b>          | <b>Nvarchar</b>      | <b>3</b>                 | Es la clave(usando las primeras tres letras del nombre de la incidencia) con la que se identifican las incidencias. |
| <b>f_inicio</b>          | <b>Datetime</b>      | <b>8</b>                 | Fecha en la cual comienza la incidencia.  |
| <b>f_fin</b>             | <b>Datetime</b>      | <b>8</b>                 | Fecha en la cual finaliza la incidencia.  |
| <b>No_doc</b>            | <b>Nvarchar</b>      | <b>15</b>                | Número con el cual se identifica el documento al generar una incidencia.  |
| <b>Observaciones</b>     | <b>Nvarchar</b>      | <b>100</b>               | Comentarios u aclaraciones que se le indiquen al reporte al generar una incidencia.                                 |
| <b>f_captura</b>         | <b>Datetime</b>      | <b>8</b>                 | Fecha en la cual se realiza la captura de Incidencia.   |
| <b>f_doc</b>             | <b>Datetime</b>      | <b>8</b>                 | Fecha de realiza el documento de incidencia.  |
| <b>No_rh</b>             | <b>Nvarchar</b>      | <b>15</b>                | Número asignado al documento de incidencia en una determinada área.   |

Tabla: niveles\_inciden

La tabla de niveles\_inciden nos permite organizar los usuarios que tendrán permisos para acceder a la base.

| NOMBRE DE COLUMNA | TIPO DE DATOS | LONGITUD DEL DATO | DESCRIPCION  |
|-------------------|---------------|-------------------|--|
| Cve_usuario       | Nvarchar      | 15                | Clave con la que un usuario puede acceder al sistema   |
| Usuario           | Nvarchar      | 85                | Nombre del usuario que puede tener permiso de entrar al sistema                                    |
| n_accion          | Nvarchar      | 5                 | Nivel con el que un usuario determinado puede agregar, modificar y eliminar determinada incidencia |
| clv_acceso        | Nvarchar      | 10                | Contraseña con la cual el usuario puede tener acceso al sistema                                    |

Tabla: Persona

La tabla de Persona nos permite realizar la lista del personal a quien se le hará una incidencia.

| NOMBRE DE COLUMNA | TIPO DE DATOS | LONGITUD DEL DATO | DESCRIPCION                                   |
|-------------------|---------------|-------------------|---|
| RFC               | Nvarchar      | 50                | Registro Federal de Causantes de los usuarios |
| Paterno           | Nvarchar      | 50                | Apellido Paterno de los usuarios              |
| Materno           | Nvarchar      | 50                | Apellido Materno de los Usuarios              |
| Nombre            | Nvarchar      | 50                | Nombre de los usuarios                        |

Tabla: cat\_incidencias

La tabla de cat\_incidencias nos permite tener la relación de incidencias de manera listada.

| <b>NOMBRE DE COLUMNA</b> | <b>TIPO DE DATOS</b> | <b>LONGITUD DEL DATO</b> | <b>DESCRIPCION</b>   |
|--------------------------|----------------------|--------------------------|--|
| cve_inci                 | Nvarchar             | 3                        | Tres primeras letras con las que inicia el nombre de la incidencia |
| Incidencias              | Nvarchar             | 100                      | Nombre de cada una de las incidencias                              |

Tabla de Auditor\_INCI

La tabla de Auditor\_INCI nos permite guardar las acciones realizadas en cada consulta.

| <b>NOMBRE DE COLUMNA</b> | <b>TIPO DE DATOS</b> | <b>LONGITUD DEL DATO</b> | <b>DESCRIPCION</b>   |
|--------------------------|----------------------|--------------------------|--|
| IP                       | Nvarchar             | 20                       | Dirección IP de los equipos de cómputo                       |
| EQUIPO                   | Nvarchar             | 20                       | Número de Computadora  |
| FECHA                    | datetime             | 8                        | Fecha en la que se realiza incidencia                        |
| ACTIVIDAD                | char                 | 4000                     | Instrucción sql generada por actividad realizada del sistema |
| cve_usuario              | Nvarchar             | 15                       | Clave con la que un usuario puede acceder al sistema         |

Tabla Aduanas:

La tabla Aduanas nos muestra de forma listada las características de las Subadministraciones Aduanales.

| NOMBRE DE COLUMNA | TIPO DE DATOS | LONGITUD DEL DATO | DESCRIPCION                                   |
|-------------------|---------------|-------------------|---|
| UNIDAD            | Nvarchar      | 5                 | Número identificador de una Aduana            |
| ADUANA            | Nvarchar      | 255               | Nombre de la aduana                           |
| ADMINISTRADOR     | Nvarchar      | 13                | Rfc de los administradores                    |
| RFC               | Nvarchar      | 50                | Registro Federal de Causantes de los usuarios |

### 3.5 Diagrama Entidad Relación

Para el análisis de Entidades y Relaciones se cuenta con una herramienta gráfica llamada Diagramas Entidad Relación (E-R), que se usa dibujando los diagramas entre las entidades y relaciones vinculadas por medio de líneas de conexión, así como indicando los atributos correspondientes.

El análisis Entidad Relación (E-R) abstrae las tablas en forma de objetos y enlaza dichos objetos mediante puntos de relación del mismo modo en que enlazaríamos objetos si estuvieran en la memoria de nuestro PC.

El diagrama entidad relación se basa en el modelo relacional. En el cual se representa como estructura de almacenamiento de datos en una Tabla o Relación, con un conjunto de datos o valores que poseen los renglones o Registros, y las columnas o Atributos para cada entidad. (Ver Figura 12).

En cuanto a las características principales que posee una tabla se pueden mencionar:

- Cada tabla debe contener un nombre único.
- El orden de los renglones no es significativo.
- No debe haber renglones repetidos.
- Debe haber uno o más atributos que identifiquen a cada renglón (llave primaria)

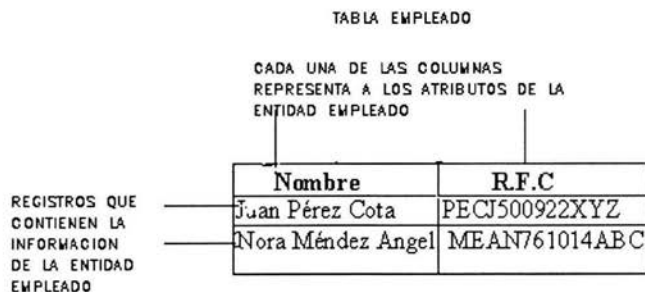


Figura 12. Registros y Atributos que conforman una Tabla



Para realizar el modelo Entidad Relación(E-R) se pueden representar tres diferentes fases: Entidades, Relaciones y Atributos.

➤ Para las Entidades y su modelado se debe considerar:

- Descubrir las entidades
- Definir una llave primaria
- Documentar

En la Figura 13 se muestran las entidades que fueron definidas para la creación del sistema.

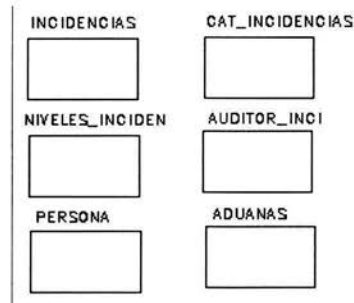


Figura 13. Modelado de Entidades

➤ Para las Relaciones y su modelado se debe considerar:

- Descubrir las relaciones
- Definir el tipo de relación
- Documentar en tablas
- Documentar en el diagrama E-R

Las relaciones de acuerdo al número de elementos que se involucran con las entidades se clasifican en: Uno a Uno(1-1), Uno a Muchos(1-M) y Muchos a Muchos(M-M).

En la Figura 14 se muestra el diagrama Entidad Relación del sistema en cuestión.

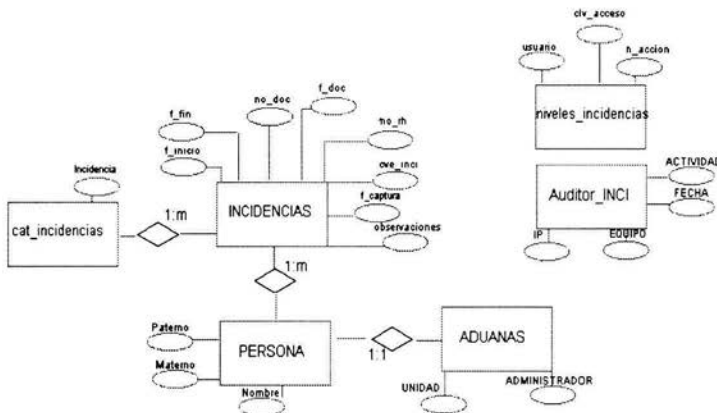


Figura 14. Diagrama Entidad-Relación del Sistema

➤ En la definición de los atributos se debe considerar:

- Descubrir los atributos
- Documentar el atributo en la entidad

Para colocar los atributos de manera correcta en una entidad se utiliza el método de normalización, que permite verificar si nuestro modelo es correcto o bien si funcionará al ser implementado. En la siguiente sección se tratará con detalle.

### 3.6 Normalización

Es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas que además de ser más simples y más estables, son más fáciles de mantener.

Se puede entender como una serie de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar un esquema que minimice los problemas de lógica. En donde cada regla está basada en la que le antecede.

En la Tabla 5 se describe brevemente en que consiste cada una de las reglas, y posteriormente se explican con más detalle.

| Regla                      | Descripción  |
|----------------------------|--|
| Primera Forma Normal (1FN) | Incluye la eliminación de todos los grupos repetidos.  |
| Segunda Forma Normal (2FN) | Asegura que todas las columnas que no son llave sean completamente dependientes de la llave primaria (PK).   |
| Tercera Forma Normal (3FN) | Elimina cualquier dependencia transitiva. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave. |

Tabla 5. Niveles de Normalización

La normalización se adoptó porque el estilo de poner todos los datos en un solo lugar como un archivo o una tabla de la base de datos era ineficiente y conducía a errores de lógica cuando se trataban de manipular los datos; así como también hace las cosas fáciles de entender; permitiendo garantizar la no existencia de redundancia y una cierta coherencia en la representación mediante el esquema de las entidades y relaciones.

Dentro de las ventajas de la normalización de base de datos se puede mencionar el consumo de espacio, ya que una base de datos normalizada ocupa menos espacio en disco que una no normalizada; además se pueden solucionar diversos errores en el diseño de la base de datos, así como mejorarlo por tener menos repetición de datos.

Existen básicamente tres niveles de normalización: Primera Forma Normal (1NF), Segunda Forma Normal (2NF) y Tercera Forma Normal (3NF). Cada una de estas formas tiene sus propias reglas.

### ➤ Primera Forma Normal

La regla de la Primera Forma Normal establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas. Poner la base de datos en la Primera Forma Normal resuelve el problema de los encabezados de columna múltiples. Muy a menudo, los diseñadores de bases de datos crean columnas que representen los mismos datos.

La normalización ayuda a clarificar la base de datos y a organizarla en partes más pequeñas y más fáciles de entender en lugar de tener que entender una tabla gigantesca que tiene muchos y diferentes aspectos; sólo tenemos que entender los objetos pequeños y tangibles, así como las relaciones que guardan con otros objetos también pequeños.

Una relación R se encuentra en primera forma normal (1FN) si cumple lo siguiente:

- Cada columna debe tener un nombre único.
- Dos filas o renglones de una misma tabla no deben ser idénticas.
- Por lo general si la mayoría de las relaciones del sistema cumplen con estas características, podemos decir que la mayoría de las relaciones se encuentran en primera forma normal.

Por ejemplo en la relación `cat_incidencias`, tenemos la llave `Id_incidencia` y el atributo llamado `incidencia`. Con lo que podemos ver que para cada valor de la llave hay un único valor en el atributo, por lo que no hay nombres de columnas repetidos, ni renglones repetidos.

| cat_incidencias |            |
|-----------------|------------|
| Id_incidencia   | Incidencia |
| 1               | Vacaciones |
| 2               | Faltas     |
| 3               | Licencias  |
| 4               | ART 50     |

Como ésta relación maneja un solo valor por cada uno de los campos que conforman al atributo de la entidad, se dice que `cat_incidencias` cumple con la 1FN.

### ➤ Segunda Forma Normal

La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Una vez alcanzado el nivel de la Segunda Forma Normal, se controlan la mayoría de los problemas de lógica. Podemos insertar un registro sin un exceso de datos en la mayoría de las tablas.

Una relación se encuentra en segunda forma normal, cuando cumple con las reglas de la primera forma normal y todos sus atributos que no son claves(llaves) dependen por completo de la clave. De acuerdo con lo anterior, cada tabla que tiene un atributo único como clave está en segunda forma normal. Es decir, los atributos deben depender de toda la clave y no de una parte de ella.

Con la siguiente relación, llamada Incidencias:

| Id folio | cve_inci | RFC  |
|----------|----------|------|
| 1        | Vac      | PML  |
| 2        | Fal      | JHO  |
| 5        | Lic      | MQLN |

Y con la tabla cat\_incidencias se muestra la normalización de la siguiente manera:

| Id_incidencia | Incidencia |
|---------------|------------|
| 1             | Vacaciones |
| 2             | Faltas     |
| 3             | Licencias  |
| 4             | ART 50     |

Las dos tablas presentadas no tienen relación hasta este momento, pero queda ahora como:

| Id folio | cve_inci | RFC  | Id_incidencia | incidencia |
|----------|----------|------|---------------|------------|
| 1        | Vac      | PML  | 1             | Vacaciones |
| 2        | Fal      | JHO  | 2             | Faltas     |
| 5        | Lic      | MQLN | 3             | Licencias  |
| 8        | ART      | ZAZO | 4             | ART 50     |

#### ➤ Tercera Forma Normal

Una tabla está normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas (aquella en la cual existen columnas que no son llave que dependen de otras columnas que tampoco son llave).

Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no deben haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir.

La 3FN consiste en eliminar la dependencia transitiva que queda en una segunda forma normal; en pocas palabras una relación está en tercera forma normal si está en segunda forma normal y no existen dependencias transitivas entre los atributos.

Con dependencias transitivas se hace referencia al existir más de una forma de llegar a referencias a un atributo de una relación.

Considerando lo anterior, se tiene el caso donde existe dependencia transitiva entre las tablas Incidencias y cat\_incidencias:

| Id folio | cve_inci | RFC  | Id_incidencia | Incidencia |
|----------|----------|------|---------------|------------|
| 1        | Vac      | PML  | 1             | Vacaciones |
| 2        | Fal      | JHO  | 2             | Faltas     |
| 5        | Lic      | MQLN | 3             | Licencias  |
| 8        | ART      | ZAZO | 4             | ART 50     |

Se aprecia que incidencia depende del valor de las Incidencias, además de que la incidencia depende del folio, pero incidencia depende básicamente del id\_incidencia, por lo tanto se puede separar la tabla de la siguiente forma:

| Id folio | cve_inci | RFC | Id_incidencia |
|----------|----------|-----|---------------|
| 1        | Vac      | PML | 1             |
| 2        | Fal      | JHO | 2             |

Quedando Finalmente

| Id_incidencia | incidencia |
|---------------|------------|
| 1             | Vacaciones |
| 2             | Faltas     |
| 3             | Licencias  |
| 4             | ART 50     |

## **IV. DESARROLLO DEL SISTEMA**

## IV.- DESARROLLO DEL SISTEMA

En el desarrollo del sistema de Servicios Administrativos Aduanales se construyó el sistema basado en los resultados del diseño; es decir la generación del código fuente, interacción con la base de datos y generación de pantallas.

Para ello se verán los principales puntos con los cuales opera el sistema.

### 4.1 Construcción del Sistema

El sistema de Información de Servicios Administrativos Aduanales consta de una Base de Datos en SQL Server 2000 a la cual se le introducen datos a través de varias páginas HTML y ASP. Por esta razón se construye el Sistema con las herramientas con que cuenta la Institución y que a continuación se detallan.

#### 4.1.1 Creación de la Base de datos

En la base de datos es en donde se concentra toda la información acerca de los usuarios que pueden hacer consultas en el Sistema de Información de Servicios Administrativos Aduanales.

Para crear la Base de Datos en SQL Server 2000, se realiza el proceso mostrado en las Figuras 15 y 16(página 52), en donde se le da nombre a la base de datos, definiendo propiedades, así como el tamaño.

Durante el diseño de una base de datos, utilizando Access, podemos establecer relaciones entre las diferentes tablas de la base de datos para que al modificar o borrar registros en unas tablas, se produzca una eliminación o actualización en cascada en las tablas relacionadas.

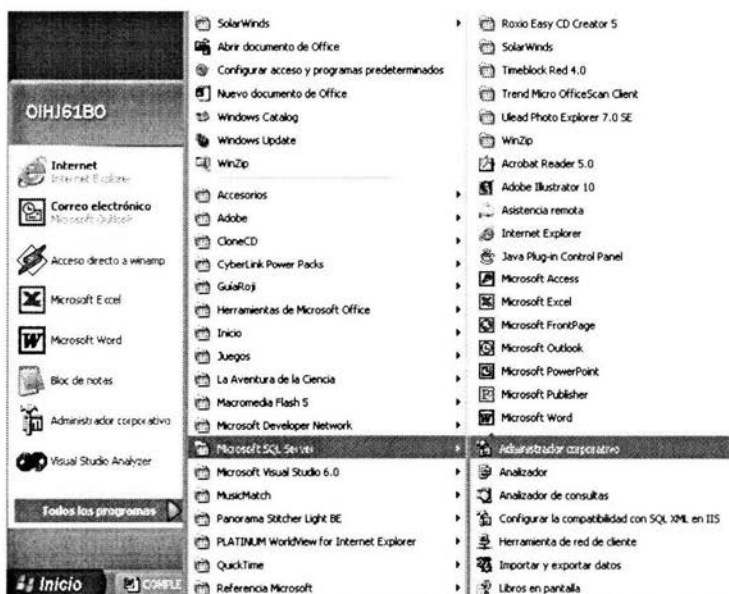


Figura 15. Diseño de la Base de Datos en SQL Server.

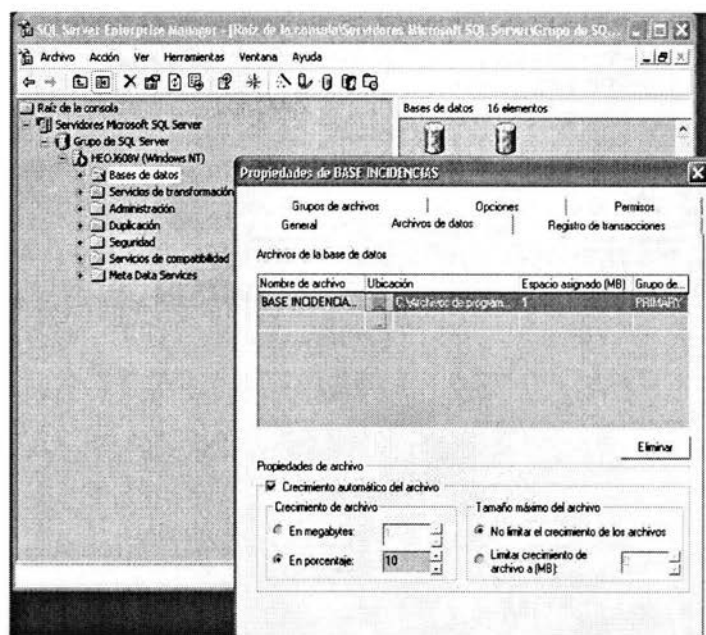


Figura 16. Creación de la Base de Datos Incidencias

Una vez creada la base de datos se procede a crear las tablas que contendrá dicha base.

El número de tablas son 6: tabla1 cat\_incidencias, tabla2 Incidencias, tabla3 niveles\_inciden, tabla4 PERSONA, tabla5 Auditor\_INCI, tabla6 ADUANAS. (Ver la Figura 17 donde se muestran dichas tablas).

En dicha figura se observa a la derecha una serie de tablas generadas automáticamente a las 6 anteriores, puesto que al introducir las tablas en SQL Server por el propio manejador de bases de datos se agregan.



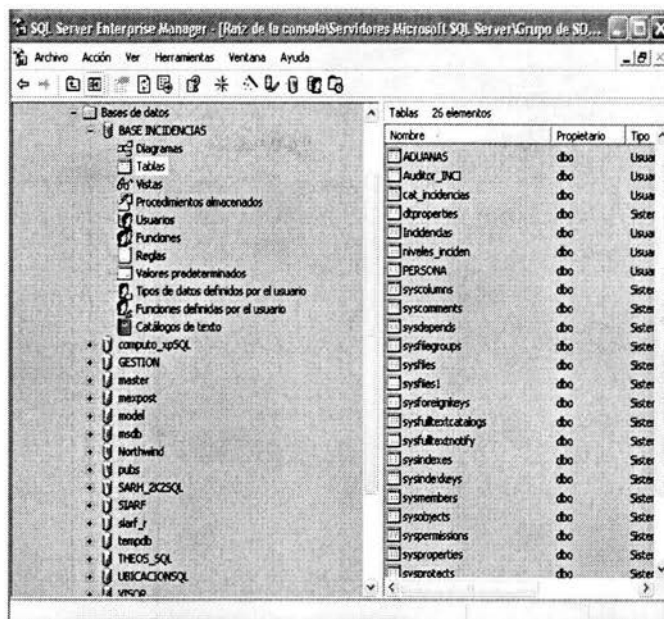


Figura 17. Tablas de la base de datos Incidencias.

## 4.1.2 Creación de las tablas en la Base de Datos

Para la creación de cada tabla se utilizó el ayudante de creación de tablas propio de SQL Server 2000, en el cual se definen cada uno de los campos que contendrá la tabla, así como cual será la llave primaria, el número de caracteres y el tipo de variable que aceptará cada uno de los campos y si aceptará valores nulos o no.

Un ejemplo de la creación de una tabla se puede ver en la Figura 18 donde se crea la tabla Incidencias. Para ello se definen cada uno de los campos de esta tabla; así como sus propiedades: Nombre, Tipo de Dato, Tamaño, Valores Nulos, y Predeterminado.

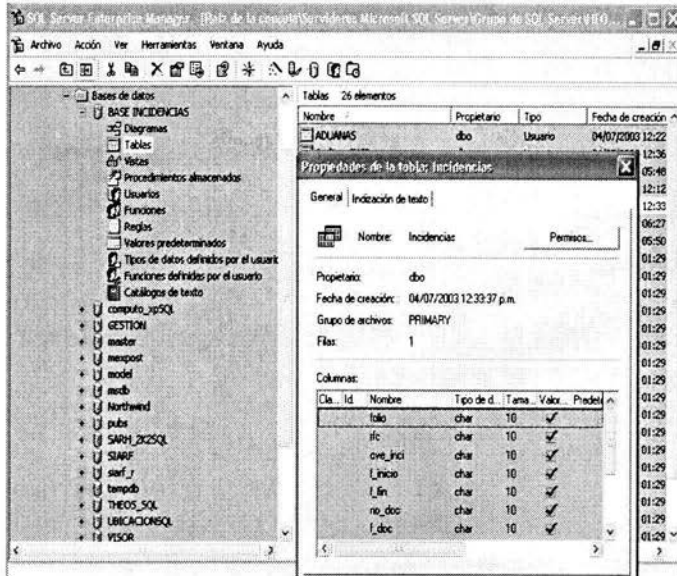


Figura 18. Creación de la tabla Incidencias.

La tabla cat\_incidencias comprende el listado de las incidencias que podrán ser consultados y se componen de cve\_inci, incidencias.

La tabla Incidencias es una de las más importantes, ya que en ella se encuentra toda la información de las Incidencias del personal que podrán ser consultados y se componen de : folio, rfc, cve\_inci, f\_inicio, f\_fin, no doc, f\_doc, no\_rh, observaciones, f\_captura.

La tabla niveles\_inciden en ella se encuentra la información de las Incidencias; los campos que componen a esta tabla son: cve\_usuario, usuario, n\_accion, clv\_acceso.

La tabla PERSONA relaciona los documentos con el tipo de usuario, sus campos son: RFC, Paterno, Materno, Nombre.

La tabla Auditor\_INCI se compone de los siguientes campos: cve\_usuario, IP, EQUIPO, FECHA, ACTIVIDAD.

La tabla ADUANAS. nos da información acerca del acceso a las Incidencias, sus campos son: UNIDAD, ADUANA, ADMINISTRADOR, RFC.

Cabe mencionar que para diseñar el modelo relacional se crearon las tablas, y el diagrama entidad relación con sus propiedades y atributos por medio de la herramienta encontrada en el software PowerDesigner DataArchitect.

En las figuras 19 , 20 y 21(páginas 55 y 56 ) se muestran las pantallas generadas para dicho fin.

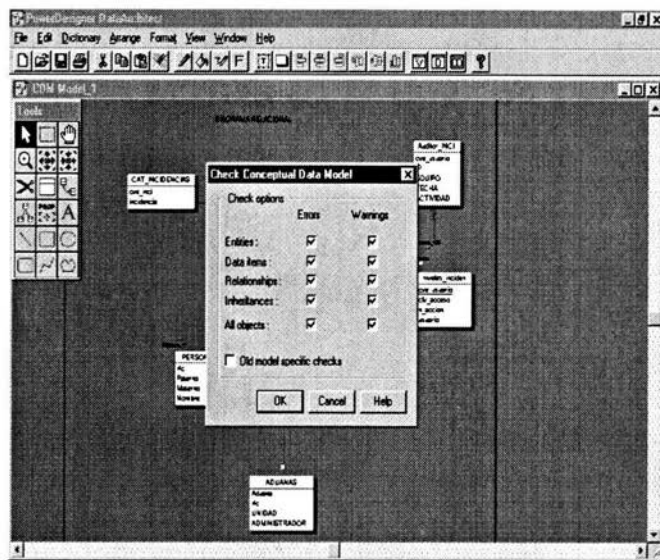


Figura 19. Creación del Modelo Relacional

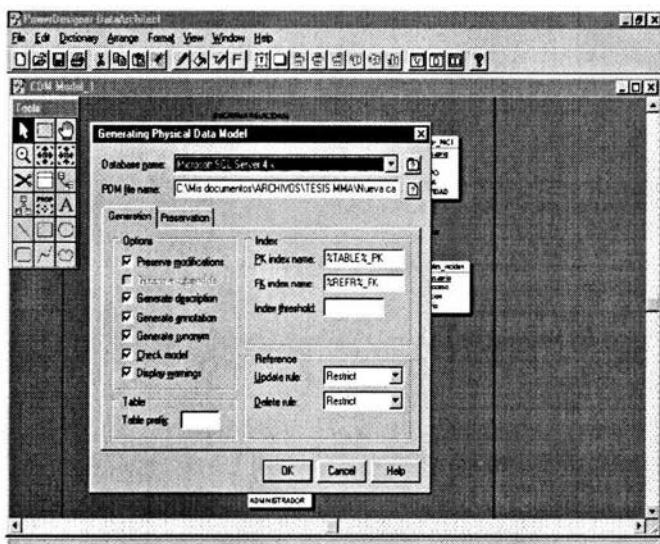


Figura 20. Conexión con SQL Server

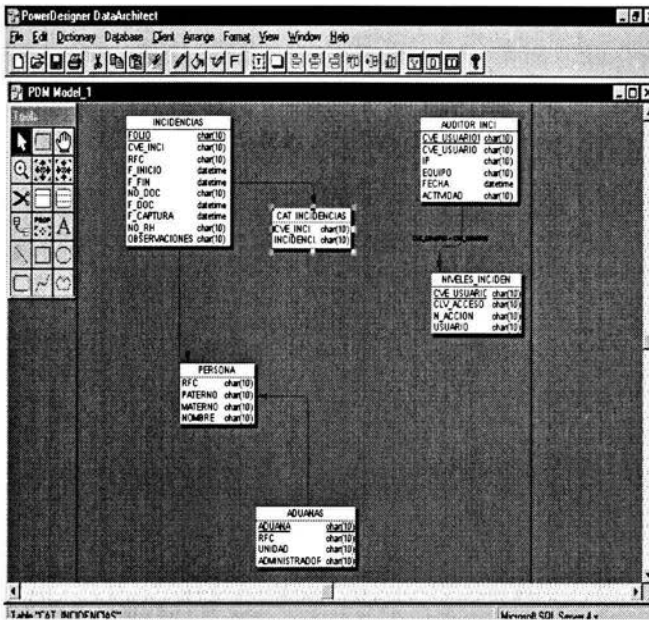


Figura 21. Diagrama Relacional para el Sistema.

## 4.2 Construcción de Pantallas

La creación de páginas en HTML(HyperText Markup Language, Lenguaje de Marcación de Hipertexto) comienza con la definición del esquema a emplear desde la página inicial.

El diseño de la página está encaminado a dar un panorama general de lo que un usuario puede manejar.

En Servicios Administrativos Aduanales se cuenta con un formato estándar para el diseño de su logotipo, mismos que son utilizados en todos sus diseños de sistemas y páginas Web. Para dicho estándar la ubicación se presenta en la parte superior centrada de la pantalla principal sobre un fondo azul.

En base a dicho estándar se construyeron las pantallas en HTML(Figura 22), dividiendo la ventana en dos partes; la primera para el logotipo, y la segunda para la Identificación del Usuario.

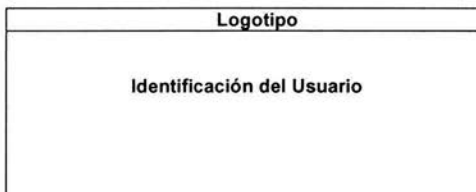


Figura 22. Definición de los Frames en la Pantalla

En el área donde se define como Logotipo, contiene el encabezado que se utiliza en la primera página del sitio.

En la parte definida como Identificación del Usuario se tiene la opción de cada usuario para poder acceder al sistema. En la Figura 23 se muestra el resultado final del desarrollo de una de las pantallas.

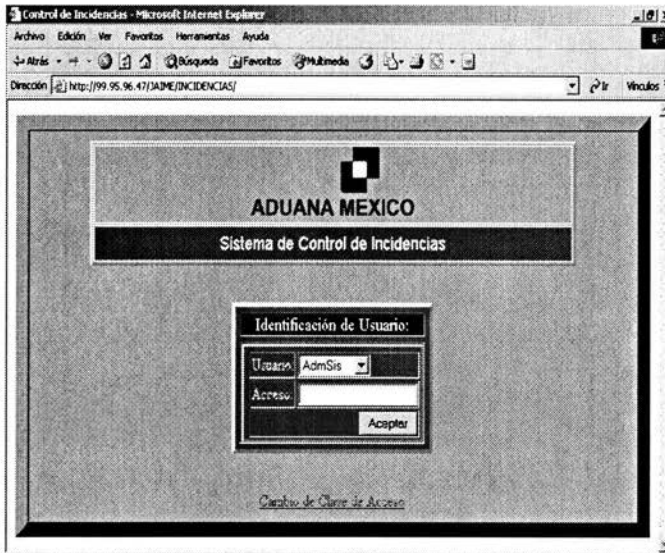


Figura 23. Presentación del formato de la pantalla principal

Para la siguiente parte del desarrollo consiste en la selección de la consulta que se desea realizar a la base de datos. Para ello durante la presentación se presentan dos áreas horizontales de la página, tomando un frame definido como Personal y otro como Captura. En la Figura 24 se muestra la división final de la pantalla.

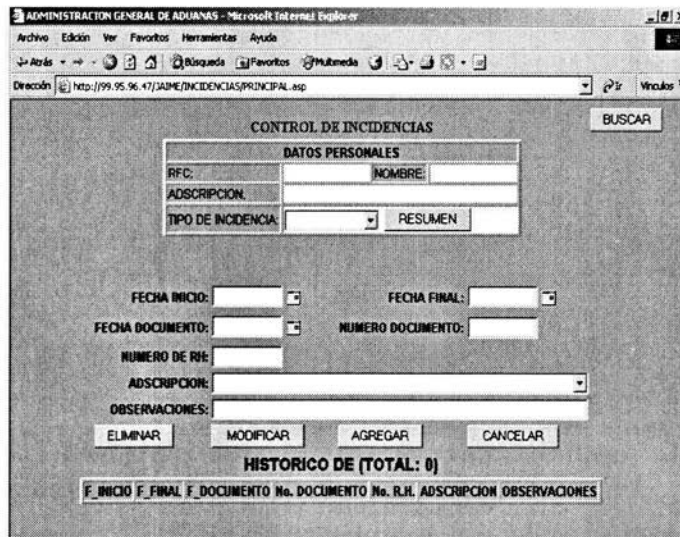


Figura 24. Presentación del formato de la pantalla

## **V. IMPLANTACIÓN Y PRUEBAS DEL SISTEMA**

# V.- IMPLANTACIÓN Y PRUEBAS DEL SISTEMA

La implantación es el proceso que nos permite asegurarnos que el sistema funcione de acuerdo a los requerimientos del análisis y así permitir a los usuarios que lo puedan operar.

## 5.1 Pruebas del sistema

Las pruebas son parte importante en el desarrollo de sistemas. Se realizan con el propósito de encontrar fallas y se establecen para mejorar la calidad del sistema. En las pruebas se requiere que se descarten las ideas acerca de lo desarrollado en el sistema, ya que, al descubrir los errores, se logra superar cualquier conflicto en el sistema.

Los objetivos de las pruebas son:

- Asegurar la obtención y formalización de los requerimientos del usuario y verificar que son adquiridos de una manera completa y correcta .
- Mostrar errores no descubiertos en los requerimientos establecidos.
- Obtener un conjunto de pruebas que tengan la mayor probabilidad de descubrir los defectos del sistema.
- Ejecutar el programa con la intención de descubrir un error.

### 5.1.1 Categoría de Pruebas

Las pruebas se centran en los procesos lógicos internos del programa, asegurando que todas las sentencias se han comprobado, realizándose para poder detectar errores cuando se produzcan resultados reales de acuerdo con los resultados requeridos.

Las categorías de pruebas pueden ser: caja blanca, caja negra, unitarias, de integración, regresión, de validación, mismas que se enuncian a continuación.

#### **Caja blanca**

Se centra en la estructura de control del programa, basado en un minucioso examen de detalles procedimentales donde se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

#### **Caja negra**

Examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del programa.

Las pruebas son diseñadas para validar los requisitos funcionales sin fijarse en el funcionamiento interno del programa.

#### **Unitarias**

Son las pruebas que centran el proceso de verificación sobre un programa o módulo con la finalidad de encontrar problemas en la lógica y problemas técnicos en el código.

**De integración**

La prueba de integración es una técnica sistemática para construir la estructura del programa que esté de acuerdo con lo que dicta el diseño.

**Regresión**

Ayudan a asegurar que los cambios (debidos a las pruebas o por otros motivos) no introducen un comportamiento no deseado de errores adicionales.

Con este tipo de pruebas se permiten detectar fallas que se hayan introducido durante las modificaciones a un sistema, permitiendo verificar que dichas modificaciones no impacten y que sigan cumpliendo con los requerimientos planteados.

**De validación**

Cuando el sistema funciona de acuerdo con las expectativas razonables del usuario, se ha encontrado y corregido errores de interfaz y puede comenzar serie final de pruebas.

## 5.1.2 Consideraciones para la ejecución de Pruebas

Es importante tomar en cuenta algunas consideraciones al realizar las pruebas, para así obtener mejores resultados en la detección de errores cometidos al desarrollar el sistema.

**Riesgos**

Los riesgos son aquellos que pueden afectar negativamente la ejecución de las pruebas.

**Suposiciones**

Son las premisas que pueden afectar positiva o negativamente la ejecución de las pruebas complicando o facilitando las actividades de las pruebas.

**Condiciones**

Lleva a cabo la definición de datos, funciones, y comportamiento al implementar el sistema, así como la información sobre el rendimiento y limitaciones que delimitan el sistema.

**Restricciones**

Están relacionadas con el desarrollo del proyecto en sí: la tecnología de pruebas, el estado de los ambientes de pruebas.

## 5.1.3 Pruebas aplicadas

Considerando los elementos que se requieren para que se lleven a cabo las pruebas, un servidor es utilizado como el ambiente de pruebas; mismo que contiene la información de Servicios Administrativos Aduanales.

Por lo antes mencionado se determinaron puntos de control entre cada página de HTML, probando todas las opciones del sistema con datos válidos y no válidos, si la página tenía más de una opción se regresaba para seguir probando todas las opciones incluidas en ella, realizando así la aprobación o no de la página.

Para probar los accesos de las ligas a las páginas en las casillas donde se llenaba la información requerida se probó que los accesos a la base de datos fueran correctos y que no hubieran problemas para desplegar la información.



Así, las pruebas que se aplicaron al Sistema de Servicios Administrativos Aduanales fueron:

- Caja negra
- Unitarias
- De integración
- De validación,

### Caja negra

Estas pruebas consistieron en entrar a la página e ir observando la lista de las diferentes administraciones para consultar aquellas a las que se tenía acceso tal como se muestra en la figura 25.

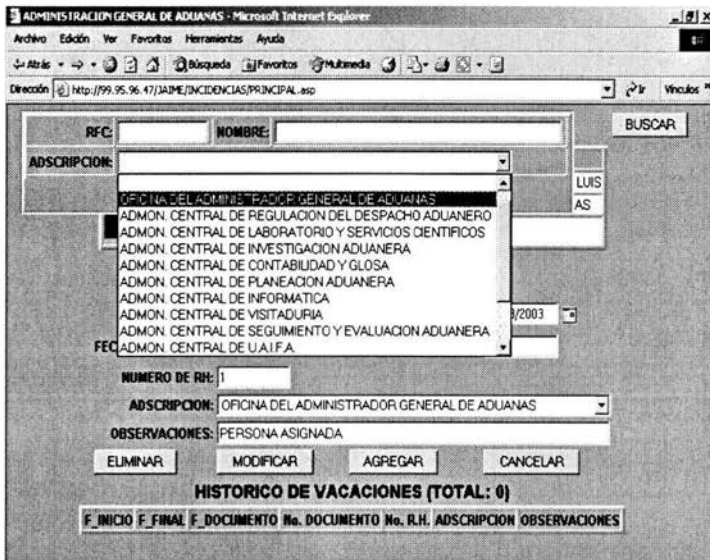


Figura 25. Prueba de Caja Negra

### Unitarias

Se llevaron a cabo pruebas en cada una de las páginas; consistiendo en validar los datos tanto de entrada como de salida, esto es, que se validó al introducir datos correctos en algunas celdas y al oprimir un botón al hacer la consulta el despliegado de los datos que fueran correctos; también se validaron al introducir datos erróneos desplegará un mensaje de error.

Para ello se validó el login y password al entrar al entrar a la pantalla principal; y en caso de insertar datos correctos el sistema pasa a la siguiente pantalla. De no validarlos el sistema solicita que nuevamente teclee información; y en caso de no insertar ningún dato también se muestra en la pantalla un mensaje solicitando insertar de nuevo los mismos. Con lo cual se presentan las figuras 26 y 27 de la página 61.

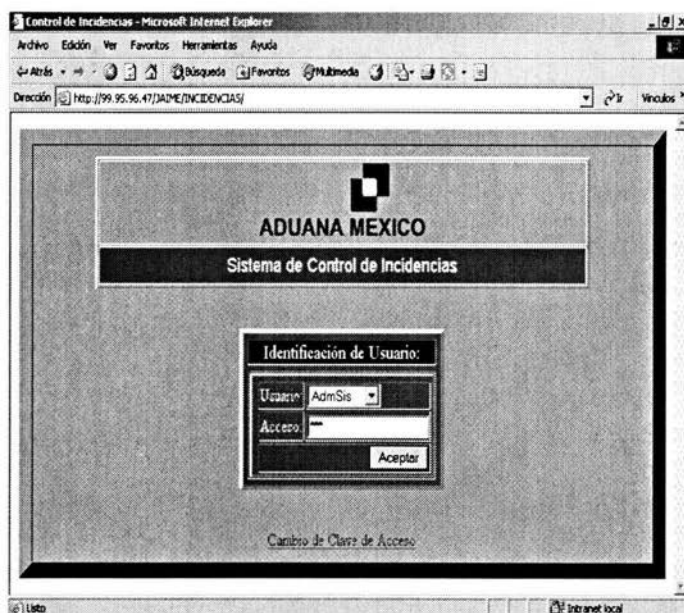


Figura 26. Prueba para validar el tipo de usuario en la pantalla principal

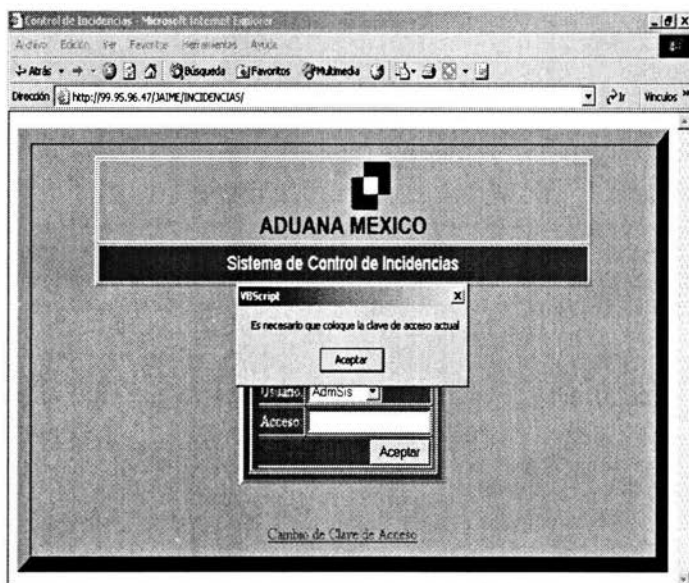


Figura 27. Mensaje al no insertar una contraseña

## De integración

Para realizar este tipo de pruebas se verificó que las ligas estuvieran bien y que no hubiera ningún problema al navegar en el sistema cuando se introdujeron datos ni al desplegarlos.

Para tal finalidad se presenta la figura 28 con el listado de datos para ciertas consultas de RFC.

| TIPO RFC | NOMBRE                              | PUESTO  | ADUANA    |
|----------|-------------------------------------|---|-----------|
| H        | CRUZ GOMEZ ALEJANDRO                | PROFESIONAL EJECUTIVO DE SERVS ESPECIALIZADOS | OJINAGA   |
| H        | RAMIREZ PORTILLA ALEJANDRO DE JESUS | PROFESIONAL EJECUTIVO DE SERVS ESPECIALIZADOS | CANCUN    |
| H        | FUENTES JUAREZ DAVID ALEJANDRO      | PROFESIONAL EJECUTIVO DE SERVS ESPECIALIZADOS | REYNOS.   |
| H        | HERNANDEZ GOMEZ AT                  | PROFESIONAL EJECUTIVO DE SERVS ESPECIALIZADOS | COTZAQUER |

Figura 28. Prueba de Integración

## De validación

Para realizar este tipo de pruebas, el usuario navegó por el sistema, y realizó consultas, tanto de ligas como de información. Mostrando para ello la figura 29.

| RFC           | NOMBRE                       | PUESTO               | ADUANA                                    |
|---------------|------------------------------|----------------------|---|
| AAEM668051001 | ESPINOSA MIGUEL              | COORDINACION (CO)    | ADMN CELA DE VISITADU                     |
| AEM162082501  | ARTEAGA MARQUEZ JOSE LUIS    | MANDO EJECUTIVO (EE) | OFICINA DEI ADMINISTRA GENERAL DE ADUANAS |
| AEM1673051001 | ANGELES MOLINA MARGARITA     | MANDO EJECUTIVO (EE) | TIJUANA, B (                              |
| AEMP51052301  | ACEVES MARTINEZ BLANCA ROSA  | TECNICO SUPERIOR     | ADMN DE SERVICIOS ADMINISTRA              |
| VAEM69111601  | VACA ESTRADA MARTHA PATRICIA | TECNICO SUPERIOR     | TAMPICO, T.                               |

Figura 29. Consulta de un usuario por su RFC

## 5.2 Mantenimiento

Una parte fundamental para el buen funcionamiento y posible crecimiento de un sistema se basa en el mantenimiento que a éste se le aplique, a continuación se da la descripción de algunos tipos de mantenimiento.

Para llevar a cabo el mantenimiento se centra en el cambio que vá asociado a la corrección de errores, a las adaptaciones requeridas y a las mejoras producidas por los requisitos de los usuarios.

Existen diversos tipos de mantenimientos, entre los que se pueden mencionar: el correctivo, adaptativo, perfectivo y preventivo.

**Correctivo.** Es el proceso que incluye el diagnóstico y corrección de uno o más errores en el sistema. El mantenimiento correctivo modifica el sistema para corregir los defectos.

**Adaptativo.** El mantenimiento adaptativo se debe a cambios en el programa y a la adaptación de nuevos módulos. Este tipo de mantenimiento produce modificaciones en el programa para acomodarlo a los cambios de su entorno.

**Perfectivo.** Comprende los cambios solicitados al programador del sistema. A medida que se usa el sistema se reciben por parte de los usuarios las recomendaciones acerca de posibilidades para realizar modificaciones; llevándolo más allá de las funciones originales.

**Preventivo.** Este tipo de mantenimiento previene errores; ya que se dá cuando se realizan cambios en el sistema con el fin de mejorar algún proceso. Se puede considerar que el mantenimiento de la información se maneja para garantizar que los resultados dados por el sistema sean correctos, y así permitir que sirva el sistema para las necesidades de los usuarios finales.

En esencia el mantenimiento preventivo hace cambios en programas de computadora a fin de que se puedan corregir, adaptar y mejorarse fácilmente.

En Servicios Administrativos Aduanales el mantenimiento correctivo se presentó cuando se tuvo un problema o error en el sistema, para lo cual se pidió a los usuarios del sistema que informaran ante cualquier anomalía .

Dadas las características del programa se tendrían que adecuarse nuevas modificaciones y/o cambios, para lo cual se deben considerar también el mantenimiento adaptativo.

Y una vez que el sistema esté trabajando se realizarán mantenimientos preventivos, con la finalidad de garantizar los resultados óptimos del sistema y previendo posibles errores.

## 5.3 Capacitación

Para tener una buena operación del sistema es necesario brindar capacitación al Administrador del Sistema y a todo usuario que tenga relación con el mismo.

Para el Administrador del sistema se brinda capacitación en:  
Desarrollo e Implementación del sistema.  
Mantenimiento del programa.

Para los usuarios se contempla la capacitación:  
Manual para el usuario.

# **RESULTADOS Y CONCLUSIONES**

## RESULTADOS Y CONCLUSIONES

Los resultados que se obtuvieron al desarrollar el sistema de Servicios Administrativos Aduanales son:

En la fase de análisis se detectaron las necesidades que los usuarios tenían; así como la forma en la cual querían que funcionara el sistema.

Se observó la forma en la que se llevaba a cabo el proceso de consulta; con lo cual se permitió detectar algunos problemas que el usuario reportaba.

En la fase de diseño se elaboró la organización general del sistema, mostrando las entradas, los procesos, las entidades y las salidas de acuerdo a la información que se recopiló en el análisis.

Se diseñaron y describieron las rutinas que utiliza el sistema.

Se definió el modelo de datos, en el cual se establecieron las entidades de datos, las claves de dichas entidades, y con ello posteriormente obtener el diagrama entidad-relación del sistema.

En la fase de desarrollo del sistema se generaron las pantallas y las tablas que pertenecen a la base de datos.

En la fase de pruebas los usuarios introdujeron datos al sistema, con la intención de encontrar problemas funcionales en la lógica, y en el código del programa desarrollado, para así finalmente tener una presentación deseada.

Por último, se generó el manual para el usuario mismo que ayudó a dar seguimiento general del sistema. Además de la documentación del código, para dar mantenimiento al código en un momento determinado.

De manera general el sistema de Servicios Administrativos Aduanales nos permite:

- 1.- Ayudar a los usuarios de Aduanas a consultar archivos de Incidencias de una manera segura.
- 2.- Accesar al sistema en cualquier momento, pero limitado a trabajar sobre la plataforma Windows que con ASP se ejecuta, ya que depende de el servidor para que se pueda trabajar en el sistema.
- 3.- Utilizar el sistema en todas las áreas de la empresa con sus respectivas modificaciones, puesto que solo se ha considerado una Administración dentro de Servicios Administrativos Aduanales.

Finalmente la información se manejará en un ambiente de Bases de Datos para mejorar la eficiencia en el manejo de transacciones operando en red bajo un esquema cliente- servidor y en un sistema de consulta bajo página web donde el personal pueda tener acceso a la información en una forma más eficiente con una administración centralizada de la información y llevando a cabo la captura de acuerdo a las necesidades del área correspondiente.

# **BIBLIOGRAFÍA**

## BIBLIOGRAFÍA

Jesús Bobadilla, Alejandro Alcocer, ACTIVE SERVER PAGES 3. CREACIÓN DE APLICACIONES WEB A TRAVÉS DE EJEMPLOS, Alfaomega ra-ma 2000.

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE SISTEMAS DE BASES DE DATOS ABRIL,1994.

I.T.Hawryszkiewicz, ANÁLISIS Y DISEÑO DE BASES DE DATOS, Megabyte.Noriega Editores. Editorial Limusa 1994.

Henry F. Korth & Abraham Silberschatz, ANÁLISIS Y DISEÑO DE SISTEMAS, Segunda Edición, Mc Graw Hill.

Georges Gardarin, BASES DE DATOS, Paraninfo Segunda Edición 1990, Madrid.

Robert J.Muller, DATABASE DESIGN for SMARTIES, Using UML for Data Modeling ACADEMIC PRESS 1999.

Gary W.Hansen, James V.Hansen, DATABASE MANAGEMENT AND DESIGN Second Edition, Prentice-Hall 1996.

David M. Kroenke, DATABASE PROCESSING FUNDAMENTALS, DESIGN, IMPLEMENTATION, Fourth Edition. USA 1988.

Pearson Bryan Pfaffenberger, DICCIONARIO DE TÉRMINOS DE COMPUTACIÓN, Prentice Hall1999.

Bryan Pfaffenberger, DICCIONARIO PARA USUARIOS DE COMPUTADORAS E INTERNET, Prentice Hall Hispanoamericana.1996. 6ª Edición.

Adoración de Miguel, Mario Piattini, Esperanza Marcos, DISEÑO DE BASES DE DATOS RELACIONALES, Alfaomega grupo Editor 2000.

Henry F.Korth, Abraham Silberschatz, FUNDAMENTOS DE BASES DE DATOS. Mc Gaw Hill 1993 España,2ª Edición.

Roger S. Pressman, INGENIERÍA DEL SOFTWARE, Mc Graw Hill Cuarta Edicion.1998.

Mark L. Gillenson, INTRODUCCIÓN A LAS BASES DE DATOS, Mc Graw Hill 1987.

Antonio López-Fuensalida, METODOLOGÍAS DE DESARROLLO, Producción Automática de Software con Herramientas CASE, Macrobit Corporation ra-ma 1991.

Brian Wilson, SISTEMAS, CONCEPTOS, METODOLOGÍA Y APLICACIONES, Editorial Limusa 1993.

Alice Y.H.Tsai, SISTEMAS DE BASE DE DATOS: ADMINISTRACIÓN Y USO, Prentice-Hall Hispanoamericana,S.A, 1988.

Alfons González Pérez, SQL SERVER.PROGRAMACIÓN Y ADMINISTRACIÓN, Alfaomega ra-ma1999.



## Pags web

[www.oracle.com](http://www.oracle.com)(Características de Manejador Oracle 9i)

[www.microsoft.com](http://www.microsoft.com)(Sistema Operativo Windows)

[www.informix.com](http://www.informix.com)(Características de Manejador de Base de Datos Informix)

Artículo de [MSDN](#) sobre las nuevas características de [SQL Server 2000](#)

# **ANEXOS**

## **APÉNDICE A. MANUAL DEL USUARIO**

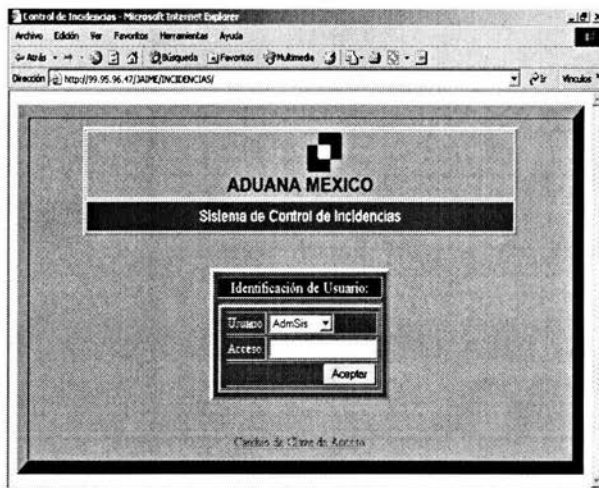
## **APÉNDICE B. DOCUMENTACIÓN DEL CÓDIGO**

## ANEXOS

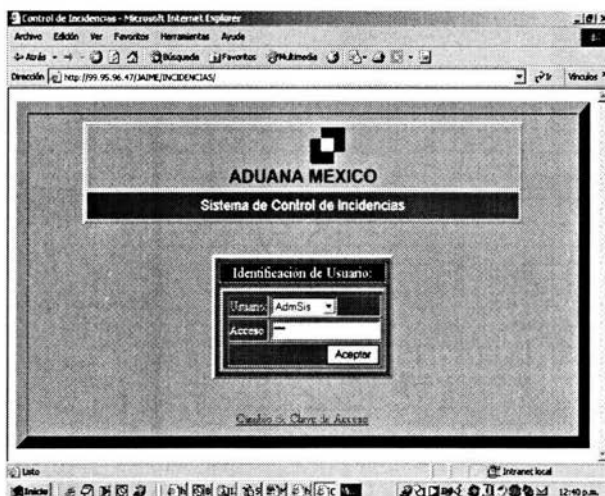
# APÉNDICE A.MANUAL DEL USUARIO

A continuación se presentarán las pantallas que utilizará el manual del usuario, así como la explicación de las mismas.

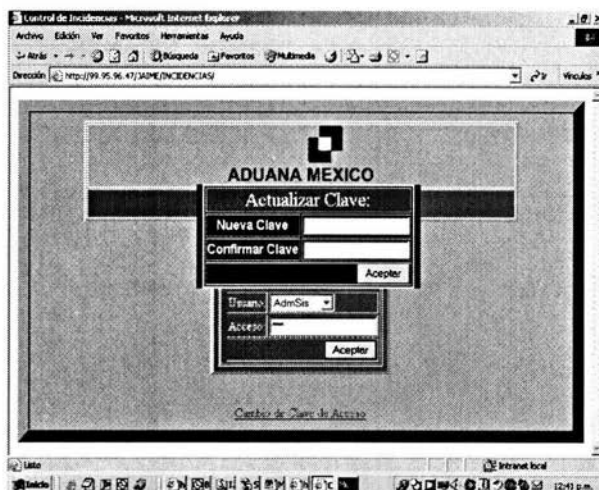
Para iniciar la sesión se tiene la pantalla principal con la clave del usuario a ser introducida.



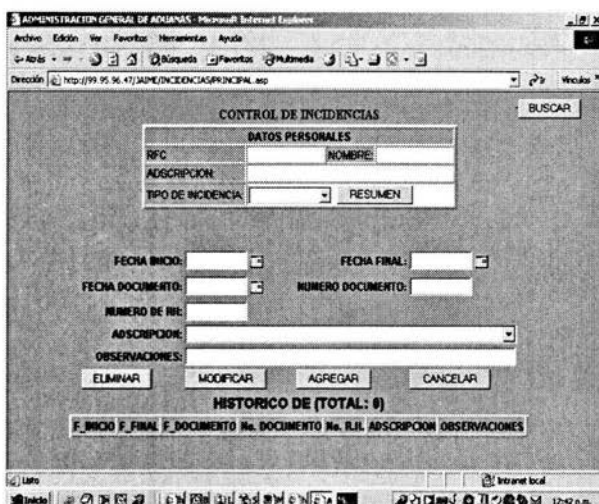
Una vez que se introduce el password en donde aparece la etiqueta con el título de acceso, se procede a dar click en el botón aceptar.



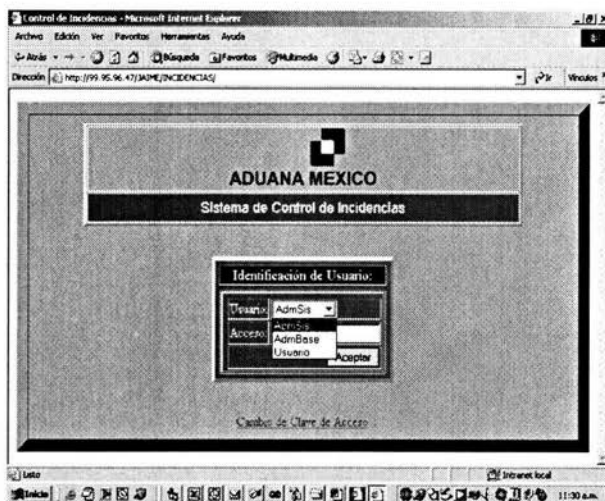
En la Liga que aparece en la parte inferior de la pantalla se tiene la opción de cambiar la clave de acceso, misma que aparece con una pantalla donde se requiere introducir una nueva clave y confirmarla, para finalmente aceptarla con el botón respectivo.



Una vez que se ha introducido la clave de acceso se presenta la pantalla de Control de Incidencias, dividido en tres partes; en la primera los datos personales a los cuales se le aplicará una determinada incidencia; la segunda parte en donde se requieren datos para registrar la Incidencia y en la última parte en donde se muestran los resultados generados después de introducir los datos requeridos.



En la pantalla de inicio al sistema también se puede cambiar el tipo de usuario, que dependerá de la clase de movimiento que desee elaborar el usuario de acuerdo a los permisos que tenga, por ejemplo agregar, insertar, borrar o modificar datos.



Si la clave de acceso no fue la correcta se muestra otra pantalla con la nota aclaratoria que la clave de acceso ha sido errónea.



Para buscar por primera vez alguna persona se puede dar click en el botón buscar y posteriormente se despliega un pequeño recuadro con los tres tipos de datos con los que se puede realizar la búsqueda; esto es por RFC, Nombre o Adscripción.

Al desplegar por adscripción el listado se presentan las administraciones a las cuales se puede buscar el personal.

Una vez que se han dado la adscripción, el RFC o el nombre o alguna combinación de ellas, se despliega un listado que coincide con los datos introducidos y con solo esas características.

En las opciones de fechas se puede escoger por medio de la imagen de calendario que tiene adjunto.



Finalmente se pueden continuar introduciendo los datos para la incidencia respectiva. Es necesario seleccionar el tipo de incidencia de lo contrario no se puede realizar el reporte de Incidencia, desplegándole el error.



Ya que se han llenado datos, se procede a agregar(haciendo click al botón con el mismo nombre y con el cual se pueden rectificar los datos, ya que enseguida de ello se presenta un cuadro con la advertencia si desea en realidad agregar los datos); para que finalmente se desplieguen en el histórico.

The screenshot shows a web browser window with the URL `http://99.96.47/ASME/INCIDENCIAS/PRINCIPAL.asp`. The page title is 'ALMINISTRACION GENERAL DE ADMANAS'. The main content area is titled 'CONTROL DE INCIDENCIAS' and includes a 'BUSCAR' button. Below this is a 'DATOS PERSONALES' section with a small profile picture and the following fields:

- RFC: AJMB51052001
- NOMBRE: ACEVES MARTINEZ BLANCA ROSA
- ADSCRIPCION: ADMON DE SERVICIOS ADMINISTRATIVOS
- TIPO DE INCIDENCIA: ARTICULO 50 (dropdown menu)
- RESUMEN: (text input field)

Below the personal data is a 'vbscript:VALIDACION' dialog box with the question '¿Agregar Registro?' and 'SI' and 'No' buttons. The main form has the following fields:

- FECHA INICIO: 05/08/2003
- FECHA DOCUMENTO: 04/08/2003
- NUMERO DE RRH: (text input field)
- ADSCRIPCION: ADMON DE SERVICIOS ADMINISTRATIVOS (dropdown menu)
- OBSERVACIONES: (text input field)
- FINAL: 23/08/2003
- ENTO: (text input field)

At the bottom of the form are buttons for 'ELIMINAR', 'MODIFICAR', 'AGREGAR', and 'CANCELAR'. Below the buttons is a section titled 'HISTORICO DE ARTICULO 50 (TOTAL: 0)' with a table header: 'F\_INICIO F\_FINAL F\_DOCUMENTO No. DOCUMENTO No. RRH ADSCRIPCION OBSERVACIONES'. The browser's status bar shows 'Internet local' and the time '12:31 p.m.'.

Después de agregar, modificar o eliminar un registro aparece una pantalla donde se hace mención la operación hecha y que en la parte de la base de datos se anexarán los datos.

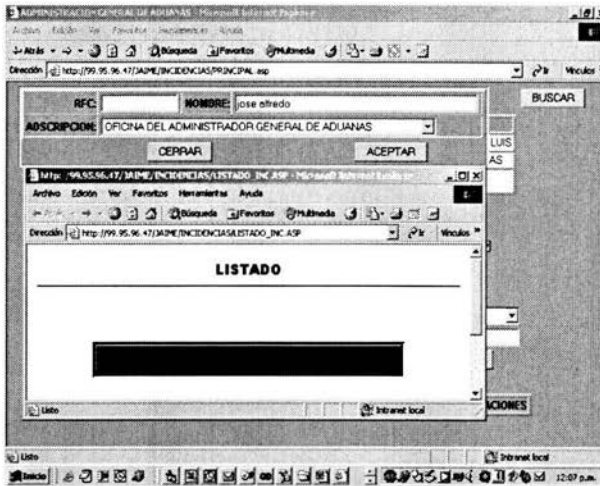
The screenshot shows the same web browser window as the previous one. The 'CONTROL DE INCIDENCIAS' section is still visible, but the 'AGREGAR' button is now disabled. A large black box with white text is centered on the page, containing the following message:

**SISTEMA DE CONTROL DE INCIDENCIAS  
ASA  
SE AGREGA DATOS EN LA BASE DE DATOS**

The browser's status bar shows 'Internet local' and the time '12:31 p.m.'.



Se puede dar el caso que por parte de la búsqueda no se localice ninguna coincidencia, por lo que se dá otra pantalla de información.



# APÉNDICE B. DOCUMENTACIÓN DEL CÓDIGO

## CAPTURA\_INC

PAGINA DONDE LOS PROCESOS A REALIZAR SON DE ACUERDO A LAS PETICIONES DEL USUARIO.

```
<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<!--#include virtual="/INCIDENCIAS/dbcon.asp"-->
<HTML>
<%
```

```
    DIM RS_DATOS
    DIM LS_RFC
    DIM LS_NOMBRE
    DIM LS_ADSCRIPCION
    DIM LS_SQL
    DIM LS_BANDERA
    DIM LS_INCI
    DIM LS_FOLIO
    dim ls_titulo
    dim RS_titulo
    DIM LS_FI
    DIM LS_FF
    DIM LS_NO_DOC
    DIM LS_F_DOC
    DIM LS_NO_RH
    DIM LS_ADSCRIPCION
    DIM LS_OBSERVACION
```

```
LS_RFC = Request.QueryString("RFC")
LS_INCI = Request.QueryString("CVE_INCI")
LS_FOLIO = Request.QueryString("FOLIO")
LS_ADSCRIPCION = Request.QueryString("CVE_ADSC")
    IF LS_FOLIO <> "" THEN
        LS_SQL = "SELECT * FROM incidencias
        where FOLIO = " & LS_FOLIO & ""
        set RS_DATOS = Server.CreateObject("ADODB.Recordset")
        RS_DATOS.Open LS_SQL, conn,3,1
            IF RS_DATOS.RECORDCOUNT <> 0 THEN
                LS_RFC = RS_DATOS("RFC")
                LS_INCI = RS_DATOS("CVE_INCI")
                LS_FI = RS_DATOS("f_inicio")
                LS_FF = RS_DATOS("f_FIN")
                LS_NO_DOC = RS_DATOS("NO_DOC")
                LS_F_DOC = RS_DATOS("f_DOC")
                LS_NO_RH = RS_DATOS("NO_RH")
                LS_ADSCRIPCION = RS_DATOS("CVE_ADSCRIPCION")
                LS_OBSERVACION = RS_DATOS("OBSERVACION")
            END IF
        END IF
    I
    F LS_RFC = "" THEN
        LS_RFC = Request.FORM("RFC")
    END IF
    LS_RFC = TRIM(LS_RFC)
%>
<SCRIPT LANGUAGE=VBScript>
    SUB FS_AGREGA()
        DIM LS_RFC
```

```
    DIM LS_INCI
    LS_INCI = "<%=LS_INCI%>"
    LS_RFC = "<%=LS_RFC%>"
        IF TRIM (LS_RFC) = "" THEN
            MSGBOX "Necesita seleccionar Personal"
```

```
    Exit Sub
    END IF
```

```
SUB FS_MODIFICA()
    DIM LS_FILIO
    LS_FOLIO = "<%=LS_FOLIO%>"
        IF TRIM (LS_FOLIO) = "" THEN
            MSGBOX "Necesita seleccionar un Registro para poder Modificarlo"
```

```
    Exit Sub
    END IF
    IF TRIM (Document.All.TXT_FI.value) <> "" THEN
        IF NOT
            ISDATE(Document.All.TXT_FI.value) OR
            LEN(Document.All.TXT_FI.value)<>10 then
            MSGBOX "Error en el Formato de Fecha DD/MM/AAAA (Ejemplo: 01/02/2003 = 1° de Feb. de 2003) EN LA FECHA INICIO"
```

```
    Exit Sub
        end if
        Else
            MSGBOX "Error en el Formato de Fecha DD/MM/AAAA (Ejemplo: 01/02/2003 = 1° de Feb. de 2003) EN LA FECHA INICIO"
```

```
    Exit Sub
    END IF
    IF TRIM (Document.All.TXT_FF.value) <> "" THEN
        IF NOT
            ISDATE(Document.All.TXT_FF.value) OR
            LEN(Document.All.TXT_FF.value)<>10 then
            MSGBOX "Error en el Formato de Fecha DD/MM/AAAA (Ejemplo: 01/02/2003 = 1° de Feb. de 2003) EN LA FECHA FIN"
```

```
    Exit Sub
        end if
        Else
            Document.All.TXT_FF.value = Document.All.TXT_FI.value
        END IF
```

```
SUB FS_ELIMINA()
    DIM LS_FILIO
    LS_FOLIO = "<%=LS_FOLIO%>"
        IF TRIM (LS_FOLIO) = "" THEN
            MSGBOX "Necesita seleccionar un Registro para poder Eliminarlo"
```

```
    Exit Sub
    END IF
    IF MSGBOX ("¿Se Eliminar el Registro Actual?", VBQUESTION+VBYESNO,"VALIDACION") = VBNO THEN
        END IF
        Document.FORMA.submit
```

```
</SCRIPT>
<TABLE ALIGN="center" BORDER=2>
<TR>
    <TD ALIGN="center"><FONT FACE="Arial" SIZE=2><B>F_INICIO</TD>
    <TD ALIGN="center"><FONT FACE="Arial"
```

```

SIZE=2><B>F_FINAL</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>F_DOCUMENTO</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>No. DOCUMENTO</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>No. R.H.</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>ADSCRIPCION</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>OBSERVACIONES</TD>
</TR>
<%do while not RS_DATOS.EOF%>
<TR>
<%FOR LI_I = 0 TO 6
IF LI_I = 0 THEN%>
<TD><FONT FACE="Arial" SIZE=2><A
HREF="CAPTURA_INC.ASP?RFC=<%=LS
RFC%>&FOLIO=<%=RS_DATOS("FOLIO"
)%>&CVE_INCI=<%=LS_INCI%>"><%=RS_
DATOS(LI_I)%></A></TD>
<%ELSE%>
<TD><FONT FACE="Arial"
SIZE=2><%=RS_DATOS(LI_I)%></TD>
<%END IF
next%>
</TR>
<%RS_DATOS.MOVENEXT
LOOP%>
</TABLE>
</FORM>
</BODY>
</HTML>

```

**PRINCIPAL**  
**PAGINA DONDE HACE REFERENCIA A LOS**  
**FRAMES QUE ACCESAN A LA PÁGINA PRINCIPAL.**

```

<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<html>
<head>
<title>ADMINISTRACION GENERAL DE
ADUANAS</title>
</head>
<frameset rows="35%,*" frameborder="no"
framespacing=0>
<frame name="CABECERA"
src="PERSONALES.asp" noresize scrolling="auto"
frameborder="no">
<frame name="CUERPO"
src="CAPTURA_INC.ASP">
</frameset>
</body>
</frameset>
</html>

```

**GUARDA\_INCI**  
**PAGINA DONDE SE PRESENTAN LOS MENSAJES**  
**DE DATOS GUARDADOS HACIA LA BASE DE**  
**DATOS.**

```

<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<!--#include virtual="/incidencias/dbcon.asp"-->
<HTML>
<HEAD>
<%

```

```

DIM LS_FOLIO
dim LS_SQL
DIM LS_DATOS
DIM LS_VALOR
DIM LS_CAMPO
DIM LS_SQLA
DIM LS_ACCION

```

```

LS_ACCION = Request.QUERYSTRING("ACCION")
SUB SQL()
SUB SQL()
'RUTINA PARA QUITAR LOS
APOSTROFOS DE LA VARIABLE LS_SQL
Y PODER INSERTARLA A LA BASE
DIM LI_I
DIM LS_M
DIM LS_LARGO

```

```

LS_LARGO = LEN (LS_SQL)
FOR LI_I = 1 TO LS_LARGO
LS_M = MID(LS_SQL,LI_I,1)
IF LS_M = "" THEN
LS_SQLA = LS_SQLA & " "
ELSE
LS_SQLA = LS_SQLA & LS_M
END IF
NEXT
END SUB

```

**CCLAVE**  
**PAGINA DONDE SE ENCUENTRA EL CAMBIO DE**  
**CLAVE**

```

<%dim LS_CVE_USUARIO
DIM LS_IP
DIM LS_EQUIPO
DIM LS_CLAVE
DIM LS_SQL
DIM LS_CVEACCESON
LS_CVE_USUARIO =
REQUEST.FORM("CMB_USUARIO")
LS_CVEACCESO =
REQUEST.FORM("TXT_CVEACCESO")
LS_CVEACCESON = REQUEST.FORM("ncvec")
LS_SQL = "SELECT * FROM niveles_incidencia where
usuario = " & LS_CVE_USUARIO & ""
set RSCustomerList = Server.CreateObject
("ADODB.Recordset")
RSCustomerList.Open LS_SQL,conn,3,1

```

**ACCESO**  
**PAGINA PRINCIPAL DE ACCESO AL SISTEMA**

```

<!--#include virtual="/INCIDENCIAS/dbcon.asp"-->
<html>
<head>
<SCRIPT LANGUAGE=VBScript>
SUB FS_CACCESO()
if document.all.TXT_CVEACCESO.value =
"" then
MSGBOX "Es necesario que coloque la
clave de acceso actual"
else
CLAVE.STYLE.VISIBILITY="visible"
end if
end sub
SUB FS_ENVIA()

```

```

"WINDOW.NAVIGATE "EFICIENCIA.ASP"
if document.all.ncvec.value <> "" then
  if document.all.ncvec.value <>
    document.all.ncve.value then
      MSGBOX "Error en nueva clave de acceso"
    else
      document.FormA.ACTION = "CCLAVE.ASP"
      document.FormA.submit
    end if
  else
    MSGBOX "Error en nueva clave de acceso"
  end if
end sub
</SCRIPT>

DIM LS_ERROR
LS_ERROR = Request.querystring
("CODIGO_ERROR")
SQLQuery = "SELECT * FROM NIVELES_INCIDEN"
set RSCustomerList = Server.CreateObject
("ADODB.Recordset")
RSCustomerList.Open SQLQuery.conn,3,1
%>
<FORM NAME = FORMA ACTION="val_acceso.ASP"
METHOD=POST>
<DIV ID="CLAVE"
STYLE="position:absolute;top:120px;left:250px;width:3
00px;height:40px;background-color:#000066;z-
index:0;visibility:HIDDEN;">
<TABLE BORDER=2 BORDERCOLOR=#FFFFFF
BORDERCOLORDARK=#FFFFFF>
<TR ALIGN="center" BGCOLOR=#0000FF><TD
COLSPAN=2><FONT FACE="Times New Roman"
SIZE=5 COLOR=#FFFFFF>Actualizar
Clave:</FONT></TD>
<TR><TD ALIGN="center"><FONT FACE="Arial"
SIZE=3 COLOR=#FFFFFF><b>Nueva
Clave</TD><TD ALIGN="center"><INPUT
type="password" name=ncve></TD></TR>
<TR><TD ALIGN="center"><FONT FACE="Arial"
SIZE=3 COLOR=#FFFFFF><b>Confirmar
Clave</TD><TD ALIGN="center"><INPUT
type="password" name=ncvec></TD></TR>
<TR><TD colspan=2 align=right><INPUT
type="button" value="Aceptar" name=button1
ONCLICK = "FS_ENVIA()"></TD></TR>
</TABLE>
</DIV>
<TABLE BORDER=15 WIDTH="100%"
HEIGHT="100%" BORDERCOLORDARK=#000066
CELLSPACING=0 CELLPADDING=0
BORDERCOLORLIGHT=#0033CC>
<br>
<TD WIDTH="100%" HEIGHT="100%"
ALIGN="center">
<TABLE BORDER=2 WIDTH="80%" align=center>
<TR>
<TD WIDTH=100% HEIGHT=36
BORDERCOLORDARK=#FFFFFF align=center>
<IMG SRC="imagenes/ADUANA.gif" BORDER=0>
</TD>
</TR>
</TABLE>
<TD WIDTH=100% HEIGHT=36
BGCOLOR=#0000FF
BORDERCOLORDARK=#FFFFFF align=center>
<FONT FACE="Arial" SIZE=4
COLOR=#FFFFFF>Sistema de Control de
Incidencias</FONT>
</TD>
</TR>
</TABLE>
<br>
<br>
<FORM ACTION="val_acceso.asp" METHOD=POST>
<TABLE BORDER=4 BGCOLOR=#0000CC
CELLSPACING=6>
<TR ALIGN="center" BGCOLOR=#000066>
<TD><FONT FACE="Times New Roman" SIZE=4
COLOR=#CCFFCC>Identificación de
Usuario:</FONT></TD>
<TR><TD><TABLE BORDER=2>
<TR>
<TD></TD>
<TD></TD>
<TD></TD>
</TR>
</TR>
<TR><TD><TABLE BORDER="2"
<TR><TD></TD></TR>
<TD><FONT FACE="Times New Roman"
COLOR=#FFFFFF>Usuario:</FONT></TD>
<TD>
<SELECT NAME="CMB_USUARIO">
<% Do While Not RScustomerList.EOF %>
<OPTION VALUE="<%=
RSCustomerList("USUARIO")%>"><%=
RSCustomerList("USUARIO")%></option>
<%RScustomerList.MoveNext
Loop
RSCustomerList.Close
set RSCustomerList= nothing%>
</SELECT>
</TD>
</TR>
<TR>
<TR>
<TD></TD>
<TD><FONT FACE="Times New Roman"
COLOR=#FFFFFF>Acceso:</FONT></TD>
<TD><INPUT TYPE="password"
NAME="TXT_CVEACCESO" SIZE=20
MAXLENGTH=40></TD>
</TR>
<TR>
<TD></TD>
<TD colspan=2 align="right">
<INPUT TYPE="submit" VALUE="Aceptar">
</TD>
</TR>
</TABLE>
</FORM>
<%
If LS_ERROR = "1" Then
%><DIV ALIGN="center">
<FONT FACE="Arial" COLOR="red">
<B>Lo siento, su clave de acceso es
erronea.<br>Intentelo de nuevo.</B>
</FONT>
</DIV>
<%End If%>
<br>
<u><font onclick = "FS_CACCESO()"
color=Blue>Cambio de Clave de
Acceso</font></u>
</TD>
</TR>
</TABLE>
<INPUT TYPE="hidden" NAME="TXT_RFC" VALUE =
"<%= LS_RFC%>">
</FORM>
</body>
</html>

```

```

<!--#include virtual="/SARH/dbcon.asp"-->
<html>
<%dim LS_CVE_USUARIO
DIM LS_CLAVE
LS_CVE_USUARIO =
REQUEST.FORM("CMB_USUARIO")
LS_CVEACCESO =
REQUEST.FORM("TXT_CVEACCESO")
SQLQuery = "SELECT * FROM niveles_inciden
where usuario = " & LS_CVE_USUARIO & ""
set RSCustomerList = Server.CreateObject
("ADODB.Recordset")
RSCustomerList.Open SQLQuery,conn,3,1
IF RSCustomerList.Recordcount <> 0 then
if LS_CVEACCESO =
RSCustomerList("CLV_ACCESO") THEN
Session.Value("USUARIO_INCI") =
LS_CVE_USUARIO
Session.Value("NIVEL_INCIDENT") =
RSCustomerList("N_INCIDENCIAS")
Session.Value("NIVEL_ACCION") =
RSCustomerList("N_ACCION")
%>
<meta HTTP-EQUIV="Refresh"
CONTENT="1;URL=PRINCIPAL.asp" target="_top">
<%
ELSE
%>
<meta HTTP-EQUIV="Refresh"
CONTENT="1;URL=acceso.ASP?CODIGO_ERROR=
1" target="_top">
<%
END IF
end if
%>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
<title>Valida Usuario</title>
<meta name="Description" content="">
<meta name="Keywords" content="">
<meta name="Author" content="Unregistered user">
<meta name="Generator" content="AceHTML 4 Pro">
</head>
<body ><center>
</body>
</html>

```

**RESUMEN\_INCI**

PAGINA DONDE SE MUESTRA EL RESUMEN DE  
INCIDENCIAS

```

<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<!--#include virtual="/INCIDENCIAS/dbcon.asp"-->
<%
DIM LS_FILTRO
dim LI_COLUMNAS
dim ls_sql
DIM LS_TOTAL(5)
DIM LS_RFC

LS_RFC = Request.QUERYSTRING("RFC")
'Rutina de armado de filtro cobn parametros
'Recibe el nombre del campo
'el valor que se recibe del control y'el tipo de filtro a
usar N=normal (=) y cualquier otra cosa para LIKE

```

```

SUB FS_FILTRO(AS_CAMPO, AS_CONTROL,
AS_TIPO)
IF AS_CONTROL <> "" THEN
IF LS_FILTRO <> "" THEN
IF AS_TIPO = "N" THEN
LS_FILTRO = LS_FILTRO & " AND " &
AS_CAMPO & " LIKE '%" & AS_CONTROL
& "%'"
END IF
ELSE
IF AS_TIPO = "N" THEN
LS_FILTRO = " " & AS_CAMPO &
" = " & AS_CONTROL & ""
ELSE
LS_FILTRO = " " & AS_CAMPO &
" LIKE '%" & AS_CONTROL &
"%"
END IF
END IF
END IF
END SUB

CALL
FS_FILTRO("RFC",Request.QUERYSTRING("RFC"),
")
IF LS_FILTRO <> "" THEN
LS_FILTRO = " WHERE " & LS_FILTRO
END IF

LS_SQL = "SELECT INCIDENCIA, COUNT(*) AS
TOTAL, SUM(DATEDIFF(DAY, f_inicio, f_fin + 1)) AS
DIAS, CVE_INCI FROM T_INCIDENCIAS " &
LS_FILTRO & " GROUP BY incidencia, CVE_INCI"
set RS_DATOS = Server.CreateObject
("ADODB.Recordset")
RS_DATOS.Open LS_SQL, conn,3,1
IF RS_DATOS.RecordCount<>0 THEN
LI_COLUMNAS =
RS_DATOS.FIELDS.COUNT
END IF

%>
<HTML>
<HEAD>
</HEAD>
<BODY BACKGROUND="IMAGENES/formato2.jpg"
BGGROUPPROPERTIES="fixed">
<%IF RS_DATOS.RecordCount<>0 THEN%>
<FORM METHOD="post" NAME="FORMA">
<TABLE WIDTH="100%" ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial Black"
SIZE=4>RESUMEN DE INCIDENCIAS<HR
COLOR="#000080"></TD>
</TR>
</TABLE>
<BR>
<TABLE BORDER=2 ALIGN="center" WIDTH="50%">
<TR>
<%
FOR LI_I = 0 TO LI_COLUMNAS-2
%>
<TD ALIGN="center"
BGCOLOR=#CCCCCC><FONT FACE="Arial"
SIZE=4><B><%=RS_DATOS.FIELDS(LI_I).NAME
%></TD>
<%
NEXT%>
</TR>
<%DO WHILE NOT RS_DATOS.EOF%>
<TR>
<%
FOR LI_I = 0 TO LI_COLUMNAS-2%>
<TD >

```

```

<%if RS_DATOS.FIELDS(LI_I).NAME =
"INCIDENCIA" THEN%>

<FONT FACE="Arial" SIZE=3><B><A
HREF="CAPTURA_INC.ASP?RFC=<%=LS_RFC%>&
CVE_INCI=<%=RS_DATOS("CVE_INCI")%>"
TARGET="CUERPO"><%=RS_DATOS(LI_I)%></A>
<%ELSE%>
<CENTER><FONT FACE="Arial"
SIZE=3><B><%=RS_DATOS(LI_I)%>
<%=LS_TOTAL(LI_I)=LS_TOTAL(LI_I)+RS_DATOS(LI
_I)%>
<%END IF%>
</TD>
</TR>
<%
NEXT
RS_DATOS.MoveNext %>
</TR>
<%LOOP%>
<TR BGCOLOR=#CCCCCC>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=3><B>TOTAL</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=3><B><A
HREF="total_INC.ASP?RFC=<%=LS_RFC%>"
TARGET="CUERPO"><%=LS_TOTAL(1)%></a></T
D>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=3><B><%=LS_TOTAL(2)%></TD>
</TABLE>
<BR>
</FORM>
<%ELSE%>
<TABLE WIDTH="100%" ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial
Black" SIZE=4>LISTADO<HR
COLOR=#000080></TD>
</TR>
</TABLE>
<TABLE WIDTH="100%" HEIGHT="100%">
<TR>
<TD>
<TABLE BGCOLOR=#000066 BORDER=2
BORDERCOLOR=#000066
BORDERCOLORDARK=#FFFFFF
ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial
Black" SIZE=5
COLOR=#CC0000>REGISTROS NO
LOCALIZADOS</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
<SCRIPT LANGUAGE="VBSCRIPT">

</SCRIPT>
<%=LS_sQL%>
<%END IF%>
</BODY>
</HTML>

PERSONALES
PAGINA DE BÚSQUEDA AL PERSONAL

<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<!--#include virtual="/INCIDENCIAS/dbcon.asp"-->
</HTML>

<%
DIM RS_DATOS
DIM LS_RFC
DIM LS_NOMBRE
DIM LS_ADSCRIPCION
DIM LS_SQL
DIM LS_BANDERA

LS_ADSCRIPCION =
Request.QueryString("CVE_ADSC")
LS_RFC = Request.QueryString("RFC")
IF LS_RFC = "" THEN
LS_RFC = Request.FORM("RFC")
END IF

LS_RFC = TRIM(LS_RFC)
%>
<SCRIPT LANGUAGE=VBScript>
SUB FS_MBUSCAR()
BUSCAR.STYLE.VISIBILITY="visible"
end sub
SUB FS_HBUSCAR()
BUSCAR.STYLE.VISIBILITY="HIDDEN"
END SUB
SUB FS_BUSCAR()
Document.FORMA.Action =
"LISTADO_INC.ASP"
Document.FORMA.submit
END SUB
SUB FS_INCIDENCIA()
'Muestra área de captura e histórico
DIM LS_INCI
dim LS_U
LS_INCI =
DOCUMENT.ALL.CMB_INIDENCIA.VALUE
LS_U =
"CAPTURA_INC.ASP?RFC=<%=LS_RFC%>
"&CVE_INCI=" & LS_INCI &
"&CVE_ADSC=<%=LS_ADSCRIPCION%>"
'msgbox ls_u
window.open LS_U, "CUERPO"
END SUB

SUB FS_RESUMEN()
'Muestra área de captura e histórico
DIM LS_INCI
dim C
dim LS_U
LS_INCI ="<%=LS_RFC%>"
IF LS_INCI = "" THEN
MSGBOX "NO HA
SELECCIONADO AL
PERSONAL"
Exit Sub
END IF
</SCRIPT>
<HEAD>
<TITLE></TITLE>
<META NAME="Description" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Generator" CONTENT="AceHTML 4
Pro">
<META HTTP-EQUIV="Content-Type"
CONTENT="text/html; charset=iso-8859-1">
</HEAD>
<BODY BACKGROUND="IMAGENES/person2.jpg">
<FORM ACTION="LISTADO.ASP" METHOD="post"
NAME="FORMA" TARGET="blank">
<DIV ID="BUSCAR" STYLE="position: absolute; top:
5px; left: 15px; width: 550px; height: 100px; a-indez: 1;
visibility: hidden; background: #000066; border:
groove; color: #000000" ALIGN="center">

```

```

<TABLE BORDER=2 BORDERCOLOR=#FFFFFF
BORDERCOLORDARK=#CCCCCC>
<TR>
  <TD ALIGN="right"><FONT FACE="Arial" SIZE=2
  COLOR=#FFFFFF><B>RFC:</TD>
  <TD WIDTH=50><INPUT TYPE="text"
  NAME="TXT_RFC" SIZE=13
  MAXLENGTH=13>&nbsp;</TD>
  <TD ALIGN="right"><FONT FACE="Arial" SIZE=2
  COLOR=#FFFFFF><B>NOMBRE:</TD>
  <TD WIDTH=50><INPUT TYPE="text"
  NAME="TXT_NOMBRE" SIZE=50
  MAXLENGTH=85>&nbsp;</TD>
</TR>
<TR>
<TD ALIGN="right"><FONT FACE="Arial" SIZE=2
  COLOR=#FFFFFF><B>ADSCRIPCION:</TD>
<!-- Conecta la BD y realiza la Consulta -->
<%
  LS_SQL = "SELECT * FROM ADUANAS WHERE
  UNIDAD LIKE '80%' order by UNIDAD"
  set RS_AUXILIAR = Server.CreateObject
  ("ADODB.Recordset")
  RS_AUXILIAR.Open LS_SQL,conn,3,1
  %>
<!-- Define los encabezados de la tabla de resultados -
->
<!-- Despliega los resultados de la consulta -->
<TD COLSPAN=3><SELECT
  NAME="TXT_ADSCRIPCION">
<OPTION VALUE=""></OPTION>
<OPTION VALUE=" " SELECTED></OPTION>
<%DO WHILE NOT RS_AUXILIAR.EOF%>
<OPTION VALUE="<%= RS_AUXILIAR(1)%>"><%=
  RS_AUXILIAR(1)%></OPTION>
<%RS_AUXILIAR.MoveNext
  loop%>
</SELECT>
<% RS_AUXILIAR.Close
  set RS_AUXILIAR= nothing
  %>
</SELECT>
</TD>
</TR>
</TABLE>
<TABLE WIDTH=100%>
<TR>
  <TD ALIGN="right"><INPUT TYPE="button"
  VALUE="CERRAR" ID="button1"
  NAME="BOT_CERRAR"
  ONCLICK="FS_HBUSCAR()" ></TD>
  <TD ALIGN="right"><INPUT type="button"
  value="ACEPTAR" id=button1
  name=BOT_CERRAR
  ONCLICK="FS_BUSCAR()"></TD>
</TR>
</TABLE>
</DIV>
<TABLE ALIGN="center">
  <TR>
    <TD><FONT FACE="Tahoma" SIZE=3
    COLOR=#000000><B>CONTROL DE
    INCIDENCIAS</FONT></TD>
</TR>
</TABLE>
<%IF LS_RFC <> "" THEN
  LS_SQL = "SELECT * FROM T_ActPlaza WHERE
  RFC = " & LS_RFC & ""
  set RS_DATOS = Server.CreateObject
  ("ADODB.Recordset")
  RS_DATOS.Open LS_SQL, conn,3,1
  %>
<TABLE BORDER=2 ALIGN="center">
  <TR BGCOLOR=#CCCCCC ALIGN="center">
    <TD ROWSPAN=4 WIDTH=10 ALIGN="center"
    VALIGN="middle"><IMG SRC="/FOTOS/<%=
    LEFT(LS_RFC,10)%>.jpg" BORDER=0 WIDTH=60
    HEIGHT=80></TD>
    <TD COLSPAN=4 ALIGN="center"><FONT
    FACE="Arial" SIZE=2><B>DATOS
    PERSONALES</TD>
  </TR>
  <TR BGCOLOR=#CCCCCC>
    <TD><FONT FACE="Arial" SIZE=2>RFC:</TD>
    <TD BGCOLOR=#FFFFFF><FONT FACE="Arial"
    SIZE=2><%= RS_DATOS("RFC")%></TD>
    <TD><FONT FACE="Arial" SIZE=2>NOMBRE:</TD>
    <TD BGCOLOR=#FFFFFF><FONT FACE="Arial"
    SIZE=2><%=RS_DATOS("NOMBRE")%></TD>
  </TR>
  <TR BGCOLOR=#CCCCCC>
    <TD><FONT FACE="Arial"
    SIZE=2>ADSCRIPCION:</TD>
    <TD COLSPAN=3 BGCOLOR=#FFFFFF><FONT
    FACE="Arial" SIZE=2><%=
    RS_DATOS("ADUANAFUN")%></TD>
  </TR>
  <TR BGCOLOR=#CCCCCC>
    <TD colspan=3><B>RS_DATOS.CLOSE
    ELSE%>
  <TABLE BORDER=2 ALIGN="center">
  <TR BGCOLOR=#CCCCCC ALIGN="center">
    <TD COLSPAN=4 ALIGN="center"><FONT
    FACE="Arial" SIZE=2><B>DATOS
    PERSONALES</TD>
  </TR>
  <TR BGCOLOR=#CCCCCC>
    <TD><FONT FACE="Arial" SIZE=2>RFC:</TD>
    <TD BGCOLOR=#FFFFFF WIDTH=100><FONT
    FACE="Arial" SIZE=1></TD>
    <TD><FONT FACE="Arial"
    SIZE=2>NOMBRE:</TD>
    <TD BGCOLOR=#FFFFFF WIDTH=100><FONT
    FACE="Arial" SIZE=1></TD>
  </TR>
  <TR BGCOLOR=#CCCCCC>
    <TD><FONT FACE="Arial"
    SIZE=2>ADSCRIPCION:</TD>
    <TD COLSPAN=3 BGCOLOR=#FFFFFF><FONT
    FACE="Arial" SIZE=1></TD>
  </TR>
  <%END IF%>
  <TR BGCOLOR=#CCCCCC>
    <TD><FONT FACE="Arial" SIZE=2>TIPO DE
    INCIDENCIA:</TD>
  <%
  LS_SQL = "SELECT * FROM
  CAT_Incidencias ORDER BY POSICION"
  set RS_DATOS = Server.CreateObject
  ("ADODB.Recordset")
  RS_DATOS.Open LS_SQL, conn,3,1%>
  <TD COLSPAN=4 BGCOLOR=#FFFFFF><FONT
  FACE="Arial" SIZE=1>
  <SELECT NAME="CMB_INIDENCIA"
  ONCHANGE="FS_INCIDENCIA()">
  <OPTION VALUE=""></option>
  <%DIM LI_LARGO
  DIM LI_I
  LI_LARGO = LEN(Session.Value("NIVEL_INCIDEN"))
  DO WHILE NOT RS_DATOS.EOF
  LI_I=LI_I+1
  IF LI_I <= LI_LARGO THEN
  IF
  MID(Session.Value("NIVEL_INCIDEN"),LI_I,
  1) = 1 THEN%>

```

```

<OPTION
VALUE="<%=RS_DATOS("cve_inci")%>"><
%=RS_DATOS("incidencia")%></option>
<%END IF
END IF%>
<%RS_DATOS.movenext
LOOP%>
</SELECT>
&nbsp;<input type="button" value="RESUMEN"
ONCLICK="FS_RESUMEN()"></TD>
</TR>
</TABLE>
<DIV ID="BOTONES"
STYLE="position:absolute;top:5px;left:680px;width:95p
x;height:145px;z-index:0;visibility:VISIBLE">
<TABLE ALIGN="center" WIDTH="100%">
<TR>
<TD><input type="button" value="BUSCAR"
ONCLICK="fs_mbuscar()"></TD>
</TR>
</TABLE>
</DIV>
</FORM>
<SCRIPT>
'Limpia área de captura e histórico
dim LS_UU
LS_UU =
"CAPTURA_INC.ASP?RFC="<%=LS_RFC%
>&CVE_INCI="
window.open LS_UU, "CUERPO"
</SCRIPT>
</BODY>
</HTML>

```

#### TOTAL\_INCI LISTADO DE INCIDENCIAS AGREGADAS

```

<!--#include virtual="/INCIDENCIAS/seguridad.asp"-->
<!--#include virtual="/INCIDENCIAS/dbcon.asp"-->
<%
DIM LS_FILTRO
dim LI_COLUMNAS
dim ls_sql
DIM LS_TOTAL(5)
DIM LS_RFC
LS_RFC = Request.QUERYSTRING("RFC")
LS_FILTRO = " WHERE RFC = " & LS_RFC
& ""
LS_SQL = "SELECT INCIDENCIA, f_inicio,
f_fin, no_doc, f_doc, no_rh, observacion,
FOLIO FROM T_INCIDENCIAS " &
LS_FILTRO & " order by incidencia, f_inicio"
set RS_DATOS = Server.CreateObject
("ADODB.Recordset")
RS_DATOS.Open LS_SQL, conn,3,1
IF RS_DATOS.RecordCount<>0 THEN
LI_COLUMNAS =
RS_DATOS.FIELDS.COUNT
END IF
%>
</HTML>
</HEAD>
</HEAD>
<BODY BACKGROUND="IMAGENES/formato2.jpg"
BGPROPERTIES="fixed">
<%IF RS_DATOS.RecordCount<>0 THEN%>
<FORM METHOD="post" NAME="FORMA">
<TABLE WIDTH="100%" ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial Black"
SIZE=4>Relación de Incidencias

```

```

(<%=RS_DATOS.RecordCount%>)<HR
COLOR="#000080"></TD>
</TR>
</TABLE>
<BR>
<TABLE ALIGN="center" BORDER=2>
<TR>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>INCIDENCIAS</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>F_INICIO</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>F_FINAL</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>F_DOCUMENTO</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>No. DOCUMENTO</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>No. R.H.</TD>
<TD ALIGN="center"><FONT FACE="Arial"
SIZE=2><B>OBSERVACIONES</TD>
</TR>
<%do while not RS_DATOS.EOF%>
<TR>
<%FOR LI_I = 0 TO 6
IF LI_I = 0 THEN%>
<TD><FONT FACE="Arial" SIZE=2><B><A
HREF="CAPTURA_INC.ASP?RFC="<%=L
S_RFC%>&FOLIO="<%=RS_DATOS("FOLI
O")%>&CVE_INCI="<%=LS_INCI%>"><%=
RS_DATOS(LI_I)%></A></TD>
<%ELSE%>
<TD><FONT FACE="Arial"
SIZE=2><%=RS_DATOS(LI_I)%></TD>
<%END IF
next%>
</TR>
<%RS_DATOS.MOVENEXT
LOOP%>
</TABLE>
<BR>
</FORM>
<%ELSE%>
<TABLE WIDTH="100%" ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial Black"
SIZE=4>LISTADO<HR
COLOR="#000080"></TD>
</TR>
</TABLE>
<TABLE WIDTH="100%" HEIGHT="100%">
<TR>
<TD>
<TABLE BGCOLOR=#000066 BORDER=2
BORDERCOLOR=#000066
BORDERCOLORDARK=#FFFFFFF ALIGN="center">
<TR>
<TD ALIGN="center"><FONT FACE="Arial Black"
SIZE=5 COLOR=#CC0000>REGISTROS NO
LOCALIZADOS</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</SCRIPT LANGUAGE="VBSCRIPT">
</SCRIPT>
<%END IF%>
</BODY>
</HTML>

```