



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
"ARAGÓN"

**"SISTEMA DE CAPTURA E INCIDENCIAS
DELICTIVAS DE LA PGJ D.F."**

TRABAJO ESCRITO

EN LA MODALIDAD DE SEMINARIOS
Y CURSOS DE ACTUALIZACIÓN Y
CAPACITACION PROFESIONAL
QUE PARA OBTENER EL TÍTULO DE :
INGENIERO EN COMPUTACIÓN
P R E S E N T A :
ENRIQUE VEGA SOLIS

ASESOR: ING. RODOLFO VÁZQUEZ MORALES

MÉXICO, 2005

0350984



Universidad Nacional
Autónoma de México



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.

AGRADECIMIENTOS

A mis padres:

Por apoyarme en todo momento, darme la oportunidad de superación y no perder la confianza en mí.....Los Amo.

A mis Hermanos:

Por aguantarme en mis momentos de desesperación y darme aliento para seguir adelante.

A mis Profesores:

Porque gracias a ellos y a su paciencia, he llegado hasta donde estoy.

A mi esposa:

Por darme el tiempo y el apoyo para sacar a delante mis proyectos, al no perder la fe en mí.....Te Amo.

Al "Caritas":

Por todo el apoyo y la amistad brindada.

A mis compañeros:

Por todos los jalones de oreja que me dieron y el apoyo brindado a lo largo de la carrera, en especial a los "DGMOS"

INDICE

INTRODUCCIÓN.....	3
CAPITULO I DIPLOMADO DE DESARROLLO DE SISTEMAS WEB (WWW).....	5
I.1 MÓDULO 1: FUNDAMENTOS DEL DESARROLLO DE SISTEMAS EN WEB.	9
I.2 MÓDULO 2: ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADOS AL WEB.	10
I.3 MÓDULO 3: INTERFASE DE USUARIO.....	15
I.4 MÓDULO 4: CAPA DE DATOS.....	18
I.5 MÓDULO 5: CAPA DE REGLA DE NEGOCIOS.....	20
I.6 MÓDULO 6 IMPLEMENTACIÓN.....	22
CAPITULO II ENTORNO DEL PROBLEMA.....	25
II.1 ANTECEDENTES DEL PROBLEMA.....	25
II.2 PLANTEAMIENTO DEL PROBLEMA.....	30
II.3 OBJETIVO.....	30
CAPITULO III ANALISIS Y DISEÑO.....	32
III.1 ANÁLISIS.....	32
III.1.1 ANÁLISIS DE LA INFRAESTRUCTURA.....	32
III.1.2 ANÁLISIS DE LAS HERRAMIENTAS DE DESARROLLO.....	33
III.1.3 ANÁLISIS DE REQUERIMIENTOS.....	38
III.2 PROPUESTA DEL SISTEMA.....	40
III.2.1 PROPUESTA.....	40
III.2.2 ALCANCE.....	42
III.2.3 BENEFICIOS.....	42

III.3 DISEÑO	44
III.3.1 CASOS DE USO.....	44
III.3.2 COMPORTAMIENTO DEL SISTEMA.....	55
III.3.3 DIAGRAMÁS DE SECUENCIAS.....	57
III.3.4 BASE DE DATOS	62
III.3.5 INTERFASE DE USUARIO Y REGLAS DE NEGOCIO.....	63
CAPITULO IV DESARROLLO E IMPLEMENTACIÓN	79
IV.1 DESARROLLO.....	79
IV.2 IMPLEMENTACIÓN.....	93
CONCLUSIONES.....	97
BIBLIOGRAFÍA:.....	99
ANEXO I CASOS DE USO	101
ANEXO II CONTRATOS DEL SISTEMA	105
ANEXO III DICCIONARIO DE DATOS	116
ANEXO IV ESQUEMAS DE NAVEGACION.....	125
ANEXO V INTERFAZ DE USUARIO Y REGLAS DE NEGOCIO	130

INTRODUCCIÓN

En la actualidad y con el constante cambio de tecnologías en el ámbito computacional, el ingeniero en computación esta obligado a actualizarse constantemente, para poder desarrollar aplicaciones que exploten al máximo la tecnología de punta, dando solución a la problemática que se presenta día con día dentro del ámbito laboral, de esta manera podrá asegurar un mejor manejo y almacenamiento de información para la correcta toma de decisiones.

El presente trabajo escrito, es la recopilación de información sobre el sistema denominado ROLES, que en la actualidad esta utilizando la Procuraduría General de Justicia del Distrito Federal (PGJ DF), el cual en su época fue una gran solución para resolver la problemática que se presentaba con los reportes solicitados por el área de la Dirección General de Política y Estadística Criminal (DGPEC) de la misma Institución; el cual sólo es un reporte diario de toda la estadística criminal en cada una de las Fiscalías de la Procuraduría, como no se contaba con una infraestructura de red que comunicara a todas las Fiscalías y las áreas centralizadas, el sistema esta desarrollado para funcionar de manera local en cada una de ellas.

Actualmente, la Procuraduría cuenta con enlaces vía microondas que permiten la comunicación entre todas sus dependencias, pero hasta la fecha no se cuenta con ninguna aplicación que explote los recursos de red con los que se cuenta en la actualidad, es por eso que el presente trabajo se enfoca en el análisis del sistema de ROLES, para crear una nueva aplicación que explote estos recursos y a su vez, permita en un futuro modificaciones que lo vayan adecuando a las crecientes necesidades de la institución.

El primer capítulo es un panorama general del diplomado “Desarrollo de Sistemas Web (WWW)”, impartido por la Universidad Nacional Autónoma de México, en la Facultad de Estudios Superiores campus Aragón, del cual se obtuvo el conocimiento necesario para el análisis y desarrollo de una nueva aplicación con arquitectura Web, la cual sustituirá al sistema “ROLES”, brindando mayores beneficios a la institución que el actual sistema.

En el segundo capítulo, se muestran los antecedentes del sistema “ROLES”, para conocer como fue desarrollado; de esta forma, poder hacer un análisis y conocer la problemática que se presenta en la actualidad; de esta manera, hacer un planteamiento del problema y plasmando los objetivos que se quieren lograr con el nuevo proyecto.

El tercer capítulo contiene al análisis de la infraestructura, de las herramientas de desarrollo y de los requerimientos necesarios para el desarrollo de la nueva aplicación; de esta forma, se hace la propuesta del nuevo sistema mostrando los beneficios que se obtendrán con el mismo, presentando los lineamientos que se seguirán y el alcance que tendrá dicha aplicación; ostentando al mismo, con la documentación obtenida de su diseño.

Por último, en el cuarto capítulo, se incluye la forma en que se crea la base de datos y se insertan datos directamente en la misma; también, se presenta el código fuente con el cual se crean las operaciones básicas para el manejo de los datos en la aplicación (altas, bajas, cambios y consultas); se presenta un gráfica de actividades para la preparación de los equipos que utilizaran la aplicación y un calendario programado para instalación del sistema en la Institución.

Capitulo I Diplomado de Desarrollo de Sistemas Web (WWW).

En la actualidad, la mayor parte de las empresas utilizan aplicaciones de bases de datos en web, ya que brindan bajos costos de mantenimiento y actualizaciones de los sistemas de una manera más fácil y sencilla a comparación de las aplicaciones tradicionales.

La arquitectura de las aplicaciones Web funciona de la forma que se observa en la figura 1, un cliente que realiza peticiones al servidor, quien las recibe y procesa para posteriormente enviar las respuestas al cliente.

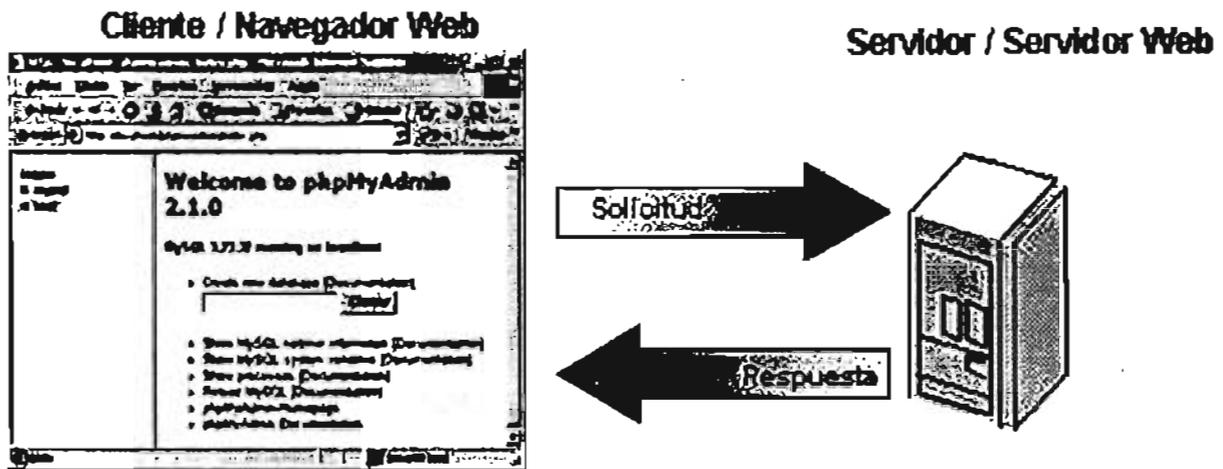


Figura 1 Arquitectura cliente/servidor

Ahora bien, esta arquitectura esta formada de 3 capas: la capa de cliente, la capa de aplicación y la capa de datos, como se puede observar en la Figura 2.

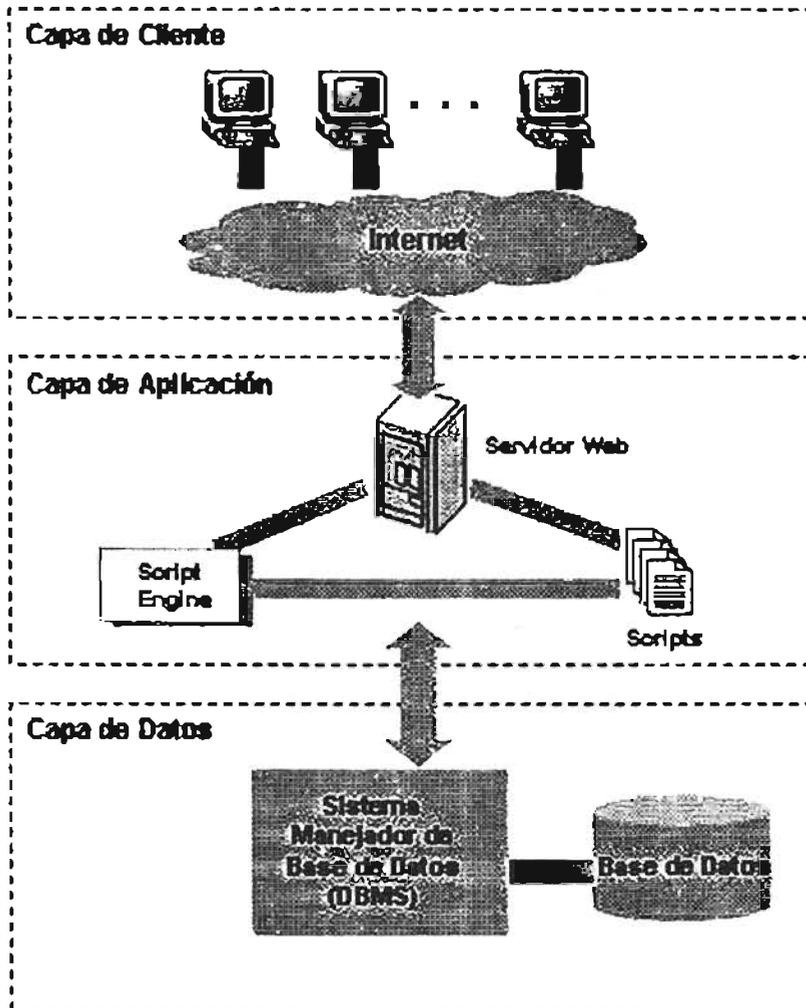


Figura 2 Arquitectura de los sistemas en WEB

Una de las ventajas que ofrece esta arquitectura, es el hecho de que los clientes son ligeros y no dependen del sistema operativo, sólo basta tener un navegador de Internet el cual se comunica al servidor Web mediante el protocolo HTTP (Hypertext Transfer Protocol, Protocolo de Transferencia de Hipertexto), enviando al servidor una petición a través de una página HTML y recibiendo una como respuesta una página igual.

Capa de Cliente

La capa de cliente normalmente es un navegador Web que procesa y muestra recursos en HTML y como ya comentamos se envían las peticiones al servidor mediante el protocolo HTTP y se recibe la respuesta del servidor mediante el mismo protocolo, dos de los más populares navegadores de Internet son Netscape e Internet Explorer.

Capa de Aplicación

La capa de aplicación es la encargada de recibir las peticiones realizadas por los clientes Web, procesarlas para enviarlas al servidor de bases de datos como consultas para leer o escribir información y enviar las respuestas a los clientes Web. Los componentes de ésta capa son: Servidor Web, Lenguaje Script de programación para Web y un Interpretador del Lenguaje Script, como se puede observar en la Figura 2.

Capa de Datos

Esta capa es la base de una Aplicación de Base de Datos en Web, ya que es la encargada de manejar los datos. El control de datos, incluye típicamente el almacenamiento y recuperación de la información, manejo de las actualizaciones, uso de los accesos simultáneos o concurrentes realizados por los procesos llevados a cabo en la capa de aplicación, proveer seguridad e integridad de la

información, así como de disponer de un servicio de respaldo de la información. Esta capa, ésta compuesta de dos elementos: Un Manejador de Base de Datos y un Lenguaje de Gestión con la Base de Datos.

La Universidad Nacional Autónoma de México en su incansable afán de de superación institucional creó un diplomado con las nuevas formas de diseño de sistemas, permitiendo poder realizar proyectos más flexibles para las empresas, brindando así grandes ventajas y una mejor visión para el desarrollo y crecimiento de las mismas; en la Facultad de Estudios Superiores Aragón se me otorgó la oportunidad de formar parte de su nuevo diplomado titulado: "Diplomado de Desarrollo de sistemas en Web (www)"; cuyo objetivo es enseñar los fundamentos y las herramientas necesarias para el planteamiento, análisis y desarrollo de sistemas con arquitectura web.

El Diplomado de Desarrollo de sistemas Web (www), tuvo una duración de 160 hrs., en horario de viernes de 18:00 a 21:00 y sábados de 9:00 a 14:00se desarrollo en base a los siguientes módulos:

- Fundamentos del Desarrollo de sistemas en web (www).
- Análisis y diseño de sistemas orientados al web (www).
- Interfase de Usuario.
- Capa de datos.
- Capa de reglas de Negocio.
- Implementación.

I.1 MÓDULO 1: FUNDAMENTOS DEL DESARROLLO DE SISTEMAS EN WEB.

En este módulo se nos dio una introducción a los sistemas a utilizar para el desarrollo de sistemas con tecnología web; comenzando con las especificaciones que se deben de tomar en cuenta para la instalación del sistema operativo que se usara como plataforma para el funcionamiento del sistema.

El sistema operativo es "Linux" el cual es una variante de "Unix" de libre distribución, el cual aporta grandes características, como son: es multiusuario y multitarea, es gratuito, soporta diversos sistemas de archivos, la mayoría del software gratuito para Unix es compatible con Linux, soporta protocolos de red como TCP/IP, soporta la mayoría de lenguajes y herramientas de programación.

Se presenció una práctica enfocada en la instalación del sistema Red Hat Linux, aportando los conocimientos necesarios para la preparación del servidor para la instalación del sistema operativo, tomando en cuenta las necesidades de la empresa o Institución y los servicios que brindara dicha empresa o Institución.

Se brindó una visión general de las aplicaciones que en la actualidad operan, las cuales, en su mayoría utilizan arquitecturas de una y dos capas, por lo que son muy difíciles de soportar adecuaciones para la problemática que se vaya presentando, además de que cada modificación hecha a las aplicaciones representa grandes costos para la empresa o institución; se explico de manera general las arquitecturas que manejan los sistemas, destacando de esta la arquitectura de tres capas, la cual será utilizada en el diplomado para el desarrollo de aplicaciones.

Se explicaron los lenguajes de programación utilizados para internet, los cuales brindan gran ventaja sobre los lenguajes tradicionales, ya que al tener el producto terminado; sólo se instalará en el servidor, lo cual agilizará el desempeño de las aplicaciones al no depender de los recursos de los clientes (sistema operativo,

capacidad de disco, memoria, etc.); mismo que brinda una gran reducción de costos para la institución o empresa ya que no es necesario tener licencias de programas o invertir grandes cantidades de dinero para la actualización de los clientes para el funcionamiento del sistema.

En cuanto a la lógica de negocios se expuso que es la capa que va a interactuar entre la base de datos y las solicitudes de los usuarios; la cual contiene reglas de validación y lógica de procedimientos necesarios para realizar las operaciones del sistema; con respecto al acceso a los datos, se explicó la estrategia UDA (Universal Data Access), la cual se puede definir como la idea para poder acceder a los datos desde cualquier punto, en cualquier momento y sin problemas.

Para finalizar se presentaron principales manejadores de bases de datos utilizados para los servidores de datos, los cuales tienen la función de gestionar los datos de una o más aplicaciones, los principales son: SQL-Server, Oracle, MySQL, PostgreSQL, Sybase, Informix, entre otros.

1.2 MÓDULO 2: ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADOS AL WEB.

En esta parte del diplomado se obtuvieron conocimientos sobre el análisis y el diseño de sistemas tomando como base la metodología Yourdon, en la cual se nos explica el ciclo de vida de los sistemas de información.

Para poder analizar y diseñar un sistema de información primeramente, se deben tener presente varias interrogantes, que darán la pauta para un buen diseño de sistemas de información, las cuales son; primeramente tomar en cuenta que es lo que se debe de analizar, en segunda instancia que es lo se debe de diseñar, en tercer lugar que se propone moldear y por último que es lo que se va a documentar.

Para el análisis de un sistema de información debe comenzarse por conocer las necesidades de la empresa o institución que desea el sistema, al inicio deberá hacerse una encuesta con el personal de la empresa para comenzar a analizar lo que desean obtener con el nuevo sistema y así poder conocer sus necesidades y requerimientos.

Obtenida la información de la encuesta se visualizará que es lo que vamos a analizar; lo que se debe tomar en cuenta primeramente, es el hardware y el software que se utilizará para el desarrollo de la aplicación, a su vez; el hardware y software necesario para el servidor y los clientes que tendrán funcionando la aplicación.

Se hará un análisis de factibilidad para verificar si es necesario crear una aplicación o no; esto comienza cuando el cliente desea automatizar los procesos de su empresa; lo cual, se puede ver reflejado en ahorro de tiempos en procesos o en ahorro de consumibles en los procesos (papel, mensajerías, etc.); para lo cual, se puede hacer un diagrama de flujo de datos (DFD) general de cómo funcionan los procesos actuales de la empresa e identificar las deficiencias del mismo para poder determinar si es factible la creación de un sistema.

Después de haber obtenido el análisis de factibilidad se harán las estimaciones de costos y tiempos para el desarrollo del sistema, tomando en cuenta que las estimaciones son siempre inexactas y sin olvidar que hay diversos factores, que se deben tomar en cuenta para dichas estimaciones; entre los factores a tomar en consideración primeramente es, que quien haga las estimaciones debe ser un gran negociador, en segundo lugar no se debe considerar que todas las personas relacionadas con el proyecto tienen la misma capacidad de trabajo y se tomará en consideración un promedio entre todos ellos; en tercer lugar no se puede estimar el trabajo de uno mismo ya que son varias las personas que participarán en el proyecto; en cuarto lugar, no se pueden entregar estimaciones detalladas prematuramente, es mejor entregar una serie de estimaciones a lo largo del

proyecto y por último se debe de considerar el factor humano de trabajo ya que toda persona llega a tener imprevistos a lo largo de los proyectos.

Se hará también un análisis de costo/beneficio en el caso de que sea necesario o si el usuario así lo requiera; primeramente, se deben de analizar los costos, entre ellos destacan el costo que será necesario para comenzar a desarrollar el sistema (software y hardware), el costo de construir el sistema (salarios por persona, gastos extras, costos de capacitación, costo de reclutamiento de personal), el costo de desarrollo del sistema (costo por hora-hombre de trabajo o por la cantidad de líneas de código del sistema), el costo por la implementación del sistema (gastos para capacitación de usuarios, gastos de instalación (software y hardware del servidor y las terminales)) y gastos operacionales del sistema (mantenimiento del sistema).

Entre los beneficios se analizarán los beneficios tácticos y los beneficios estratégicos; los primeros, se refieren a la reducción de tiempos en los procesos actuales, reducción en uso de consumibles (papel, plumas, etc....); también, es posible incluir la reducción de personal para realizar dichos procesos; los segundos se refieren a identificar que nuevos mercados se pueden abrir con el nuevo sistema o atraer nuevos clientes, entre otros.

Una vez terminado el análisis del sistema, se pasa a lo que se debe diseñar, comenzando por el diseño de la base de datos, ya que esta será el motor de todo nuestro sistema; se debe cuidar siempre la integridad referencial de la información evitando la redundancia de la misma, lo cual, se logrará con el correcto y eficaz uso de "llaves"; estas, pueden ser de tres formas distintas, las llaves primarias son aquellas que sirven para identificar de manera única e irrepetible a un registro dentro de la base de datos; las foránea sirven para relacionar dos o más tablas entre si con un campo en común y las llaves condicionales son aquellas en las que se debe de respetar el contenido de los campos en particular por medio de condiciones específicas.

Se desprende que dentro de la relación que guardan las tablas en una base de datos, se pueden encontrar tres tipos de relaciones las cuales son: Uno a uno (solo un campo de la tabla A se relaciona con un solo campo de la tabla B), uno a muchos (un campo de la tabla A se puede relacionar con varios campos de la tabla B) y muchos a muchos (varios campos de la tabla A se relacionan con varios campos de la tabla B), esto se puede apreciar mejor en el modelo entidad-relación de las bases de datos.

Relacionado con el diseño de la base de datos, se encuentra el desarrollo de un diccionario de datos, el cual, explicará de manera clara el diseño físico de la base de datos; es decir, describirá el nombre de cada tabla, qué campos contiene, qué tipo de datos se almacena en cada campo y qué condiciones se deben de cumplir para el almacenamiento de los datos.

Dentro de lo que se debe modelar se brindó una explicación sobre la metodología UML y sobre las dos herramientas tradicionales para el modelado, las cuales son, el diagrama de flujo de datos (DFD) y el diagrama entidad relación de los datos.

Un diagrama de flujo de datos, es la representación gráfica de los procesos identificados dentro del sistema; muestra los procesos, el flujo y el producto terminado; los procesos se representan mediante el nombre del proceso dentro de un óvalo, el flujo se representa mediante flechas en las que se indica el movimiento de bloques o paquetes de información de una parte del sistema a otra y el producto terminado se representa por medio de rectángulos.

En la metodología UML el diagrama de flujo de datos, es sustituido por un diagrama de casos de uso, el cual, representa la forma en que trabaja un sistema o un negocio; los componentes del diagrama de casos de uso, son:

- Actores: es el rol que juegan las personas que utilizaran el sistema.
- Caso de uso: es una función específica identificada dentro del sistema.
- Relación: es la forma en que interactúan los actores con los casos de uso.

Una vez construido el diagrama de casos de uso se debe hacer una identificación de todas las funciones necesarias para el funcionamiento del sistema y se vaciarán en una tabla; debiéndose documentar la descripción de cada caso de uso y los contratos del sistema.

La descripción de los casos de uso muestra de manera detallada el propósito que tiene el sistema, describiéndonos como interactúa el usuario con el sistema y las respuestas del sistema hacia el usuario; los contratos ayudan a definir el comportamiento que tiene el sistema, describiendo el efecto de todas las operaciones sobre el mismo. Ya con estos datos se puede elaborar un diagrama de secuencias, el cual, representa gráficamente las llamadas entre los módulos del sistema hechas por el usuario; éste diagrama, ayuda a identificar posibles errores que pueden presentarse dentro del sistema que impidan el libre flujo de datos.

El diagrama entidad-relación, muestra la relación que existe entre la información dentro de la base de datos, consta de dos componentes principales, el primero son los tipos de objetos (tablas) los cuales se representan por medio de rectángulos y son la colección o un conjunto de objetos; y el segundo son las relaciones, que son representadas por líneas terminadas por rombos en los diagramas y representan las conexiones o asociaciones de los objetos entre los tipos de objetos.

Dentro de lo que se debe documentar se encuentra que este punto es el resultado de las actividades anteriores, por lo que se deben de haber generado o se debe generar la siguiente documentación:

- Análisis de Software y Hardware.
- La especificación de los procesos.
- Las estimaciones (Opcional).
- El diagrama Entidad relación.
- El diccionario de datos.
- El manual de Usuario.
- El manual técnico con el listado y las referencias de los programas.

I.3 MÓDULO 3: INTERFASE DE USUARIO.

Esta parte del diseño de sistemas es importante, ya que será el ambiente en el que trabaje el usuario final y de ella depende la aceptación o no del sistema por parte del usuario, es por esto que se debe poner cuidado en el diseño de la misma.

Dentro de este punto en el diplomado se explicaron diversos puntos a tomar en cuenta para el diseño de la interfaz de usuario, entre los más destacables tenemos los siguientes:

- Debe ser un diseño amigable; es decir la gama de colores y diseños de pantallas deben ser lo más agradables hacia el usuario, de preferencia con diseños sencillos que no hagan sentir al usuario dificultad para ingresar la información y a su vez que sienta que no es una herramienta que desplazará su trabajo sino que simplificará sus procesos en cuestiones de tiempo.
- Congruencia con los procesos; se refiere a que todos los procesos deben tener una secuencia congruente, y no hacer redundante la captura de información para que los usuarios no sientan que es un doble esfuerzo trabajar con el sistema.

- Distribución lógica; el sistema debe estar distribuido de manera lógica para facilitar el trabajo de los diversos usuarios que interactúan con el sistema y de esta manera, asignar perfiles que permitan distinguir entre la gama de usuarios que lo utilizarán.
- Navegación lógica; las pantallas del sistema deberán estar distribuidas congruentemente, llevando una secuencia entre ellas para que los usuarios cometan el menor número de errores y no sientan que el sistema es una carga para el desempeño de sus funciones.
- Implementación de ayuda; es importante implementar ventanas de ayuda dentro del sistema para que los usuarios puedan consultarlas en cualquier momento dentro del sistema, ya que ellas los orientarán en los momentos que tengan dudas sobre la captura de la información.
- Estandarización con otros sistemas; se aconseja que si la empresa o institución manejan otros sistemas, se trate de estandarizar lo más posible los nuevos sistemas con los anteriores, para que si en un futuro la empresa o institución decide unirlos en un sólo sistema sea más sencillo de integrarlos o en su caso de compartir la información entre ellos.

Para el desarrollo de la interfaz de usuario se dió una introducción a al lenguaje HTML, el cual, es estándar para todos los navegadores existentes y tomando en cuenta que nuestro sistema estará desarrollado para un ambiente web, es una opción aceptable para el desarrollo; además de que se pueden utilizar hojas de estilos (CSS) las cuales nos facilitarán el desarrollo de la interfaz de usuario.

En esta parte del diplomado se obtuvieron los conocimientos para poder crear páginas web utilizando únicamente el lenguaje HTML, creando así, desde un encabezado simple hasta un formulario completo; entre los conocimientos adquiridos tenemos los siguientes:

- Creación de encabezados.
- Creación de listas.
- Creación de tablas.
- Creación de menús.
- Utilización de imágenes
- Hipervínculos
- Creación de Formulario.

Dentro de cada uno de los puntos anteriores se explicó la forma de dar formatos, cambiar tipos de letras, colores, tamaños, etc.; es decir, a dar el formato la vista deseada en cada uno de los puntos anteriores.

Una vez visto la forma de crear hasta un formulario totalmente realizado con lenguaje HTML, se brindó una introducción a lo que son las hojas de estilo (CSS), las cuales, son etiquetas que nos ayudarán a definir los formatos que tendrán los diversos componentes de nuestra interfaz de usuario de un manera más sencilla.

Los CSS brindan una ventaja en la creación de páginas web por medio del HTML, ya que si se utilizan de manera adecuada es más sencillo modificar toda la vista de las pantallas con hacer unos simples cambios en lugar de tener que modificar la mayor parte de la codificación de las páginas web.

Otra herramienta que se proporciono fue, el uso del programa Macromedia Dreamweaver; el cual, es un programa visual que nos permite de manera más sencilla crear páginas web y hojas de estilo para el desarrollo de la interfaz de usuario; además de poder interactuar con el código fuente y modificarlo libremente para obtener los resultados deseados en las páginas web; aunándole que permite hacer conexiones directamente con gran parte de las bases de datos para la creación de los formulario de una manera sencilla.

I.4 MÓDULO 4: CAPA DE DATOS

Este módulo lo considero de los más importantes para el diseño de sistemas, ya que de un buen diseño de la base de datos depende un mejor manejo e integridad de la información de un sistema.

Una base de datos, es un conjunto de objetos interrelacionados; los objetos de un mismo tipo se les define como entidades y las relaciones encontradas entre las entidades se define como asociación.

Es importante decir que, como se esta trabajando sobre una arquitectura de tres capas, la base de datos se debe de hacer independiente a los programas de aplicación; teniendo como objetivo, las siguientes funciones:

- Permitir la definición de todos los datos. Deberá aceptar la definición de todos los objetos que participaran dentro de la base de datos.
- Admitir la manipulación de los datos. Deberá permitir todas las operaciones necesarias en los datos (inserción, borrado, consultas, etc.).
- Establecer controles de seguridad para los datos. Garantizar que sólo los usuarios autorizados puedan realizar operaciones correctas sobre la base de datos.

El ciclo de vida de una base de datos se puede descomponer en tres etapas:

1. Concepción: Consiste en reproducir el mundo real con ayuda de un modelo grafico para obtener un modelo que se deberá analizar y detectar posibles errores o fallas antes de crear la base de datos.
2. Creación de la Base de Datos (Vacía): Consiste en traducir el esquema anterior en una serie de instrucciones y órdenes entendibles por el manejador de base de datos para la creación de las entidades, atributos y sus relaciones.

3. Explotación: Es la parte en que los datos serán manipulados por lo lenguajes de programación para realizar las operaciones necesarias sobre ellos (inserción, borrado, consulta, etc.).

Se debe tener un gran cuidado en la concepción de la base de datos, porque es la base para el desarrollo de toda la base de datos, se debe de cuidar que no exista redundancia entre los datos, incoherencias, perdidas de datos y dependencias funcionales; tomando en cuenta estos aspectos se puede asegurar que se hará un diseño correcto de la base de datos.

Otro punto tratado dentro del módulo fueron las sentencias SQL; el cual, es un lenguaje normalizado y utilizado por diversos manejadores de bases de datos para realizar determinadas operaciones sobre los datos y la estructura de los mismos.

El lenguaje SQL, está compuesto por comandos, cláusulas, operadores y funciones de agregado, los cuales al combinarse crean las instrucciones necesarias para la creación, actualización y manipulación de la base de datos.

Se aprendió a utilizar las instrucciones de este lenguaje para crear una base de datos, y manipular los mismos contenidos en la basen mencionada, entre ellas, se encuentran sentencias para:

- Crear tablas.
- Eliminar tablas.
- Modificar tablas.
- Definir llaves
- Consultar registros
- Insertar datos.
- Actualizar datos.
- Borrar Datos.
- Hacer consultas diversas.

- Ordenar Datos.
- Manejo de campo fecha.
- Manejo de campo Hora.
- Agrupación de datos.
- Conteo de Datos.

Se hizo una práctica de los conocimientos adquiridos sobre el manejo de la base de datos MySQL, creando una base de datos, agregando información a la base y manipulándolos para hacer altas, bajas, cambios y consultas de los mismos.

I.5 MÓDULO 5: CAPA DE REGLA DE NEGOCIOS.

Este módulo, hace referencia a las reglas que debe de seguir la aplicación para garantizar el buen funcionamiento del sistema; estas reglas, son llamada reglas de negocio y se pueden clasificar en:

- Reglas del modelo de datos: se encargan de controlar la información básica almacenada para cada atributo de una entidad.
- Reglas de Relación: se encargan de controlar las relaciones entre los datos; es decir, los datos no pueden ser eliminados si están relacionados con otros datos.
- Reglas de Derivación: son un conjunto de reglas que especifican y controlan la obtención de información que se puede calcular a partir de la ya existente.
- Reglas de restricción: son las encargadas de restringir cualquier operación sobre la información si no es autorizada.
- Reglas de Flujo: son las que determinan o limitan el flujo de la información a través de un sistema de información.

Dentro del sistema, se deben de distribuir las reglas de negocio entre las capas del sistema, la decisión de donde serán puestas se simplifica si se toma en cuenta que tipo de regla es y de este modo colocarla en la capa más adecuada; las reglas del modelo de datos como son las que definen el tipo de datos que se aceptaran en cada campo se deben de colocar en el servidor; como las reglas de relación cuidan la relación entre los datos, y como la mayoría de los manejadores de bases de datos brindan una herramienta para cuidar dichas relaciones, se deberán colocar en el servidor; como las reglas de derivación, proporcionan información sobre una ya almacenada, es más factible que se coloquen en el servidor ya que es él quien contiene dicha información, y las reglas pueden hacerse desde un lenguaje de programación o con sentencias SQL; las reglas de restricción, deben de colocarse en la capa intermedia, ya que son las encargadas de aceptar o denegar la información antes de almacenarse en la base de datos y por último, las reglas de flujo se encontrarán en la capa intermedia ya que manejan el flujo de los datos.

Para el desarrollo de las reglas de negocio, se vio a PHP como lenguaje para el desarrollo de las aplicaciones, ya que hace dúo con MySQL; se comprendió la sintaxis del lenguaje en cuestión; se adquirieron conocimientos para el manejo de:

- Variables
- Arreglos
- Cadenas
- Funciones
- Controles de Flujo
- Repeticiones
- Ciclos
- Operadores
- Pasar Variables por URL
- Utilización de cookies
- Trabajar con Bases de Datos

- Lectura de datos
- Borrado de datos
- Agregado de Datos

Nos aportaron librerías útiles para PHP como el ADODB, el cual es una herramienta que ayuda en el manejo de las altas bajas y cambios dentro de catálogos sencillos de alguna aplicación.

También proporcionaron la librería SecurePages, la cual permite la aplicación de perfiles de seguridad a nuestro sistema, desde la clave de acceso al mismo para asignarles un perfil de trabajo y acceso sólo a las páginas autorizadas para su perfil.

1.6 MÓDULO 6 IMPLEMENTACIÓN.

En este módulo se tuvo una interacción entre los asesores y los alumnos del diplomado, en el cual, se integraron todas las partes del sistema para lograr el producto terminado.

Una herramienta auxiliar para el ensamblaje de las partes del sistema es Dreamweaver, ya que al ir desarrollando las pantallas de la interfaz de usuario, se pueden anexar las instrucciones de código PHP para las reglas de negocio y a su vez, crear la conectividad con la base de datos.

Una vez ensamblado todo el sistema se deben hacer un cronograma de actividades para la implementación del sistema en la empresa o institución; esta se debe de adecuar a las necesidades de la empresa, ya que se estará interrumpiendo su trabajo diario para implementar dicho sistema.

Para la creación del cronograma de actividades para la implementación, se deben tomar en cuenta los siguientes aspectos:

- Preparación del servidor: se estimarán tiempos junto con la empresa o institución para no afectar a su trabajo diario.
- Preparación de los clientes: De igual forma que el anterior, se deben de estimar los tiempos junto con la empresa o institución.
- Instalación de la aplicación en el servidor: Determinar el tiempo necesario para la instalación de la nueva aplicación y en su caso, el tiempo necesario para la exportación de los datos de los sistemas anteriores.
- Pruebas de conectividad: Una vez preparado el servidor con la aplicación se debe de determinar un período de pruebas de conectividad con el servidor.
- Capacitación del personal: se debe de programar un ciclo de capacitación para los usuarios del sistema, esta actividad se puede realizar de forma paralela con las anteriores.

Una vez instalado el sistema y capacitado el personal, se deberá programar un período de prueba para la determinación del buen funcionamiento del sistema; agotado este tiempo se programarán periodos de mantenimiento para hacer respaldos de la información y asegurar de esta manera el optimo desempeño de la aplicación.

Para finalizar, a continuación se presenta un esquema de actividades y productos terminados de todos los módulos del diplomado.

Módulo	Actividad	Producto terminado
1	Especificación de servidores	Preparación del servidor para la instalación de las aplicaciones
1	Linux	instalación de linux con sus módulos para el lenguaje PHP y el manejador de bases de datos MySQL
1	Introducción al desarrollo de sistemas web	Conceptos básicos para el desarrollo de sistemas web.
2	Análisis del sistema	Análisis del software y hardware, análisis de factibilidad, análisis de costo/beneficio.
2	Diseño del sistema	Diseño preeliminar de la base de datos
2	Modelado del sistema	Diagrama de flujo de datos
2	Metodología UML	Diagrama de casos de uso, Descripción de los casos de uso, contratos, funciones del sistema y diagramas de secuencias.
3	HTML	Conocimientos básicos para la creación de páginas web con encabezados, listas, menús, tablas y formularios.
3	Dreamweaver	Uso del dreamweaver para la interfaz de usuario.
4	Modelado de la base de datos	Modelo de la base de datos y diccionario de datos
4	SQL	Creación de sentencias para la creación de la base de datos y para el manejo de la información.
4	MySQL	Creación de la base de datos y sentencias para el manejo de los datos.
5	Tipos de reglas de negocio	Identificación de las reglas de negocio.
5	Ubicación de las reglas de negocio.	Ubicar las diversas reglas de negocio en la capa más adecuada para su función.
5	PHP	Sentencias y librerías del lenguaje de programación para la creación de las reglas de negocios.
6	Integración	Integración de las diversas partes del sistema para obtener el producto final
6	Implementación	Cronograma de actividades para la implementación del sistema
6	Mantenimiento	Cronograma de mantenimiento y respaldos al sistema

CAPITULO II ENTORNO DEL PROBLEMA

II.1 ANTECEDENTES DEL PROBLEMA

En 1997 la Dirección General de Política y Estadística Criminal (DGPEC), solicitó a cada una de las fiscalías un reporte diario de incidencias delictivas el cual, debía de contener el número de la averiguación previa, el delito, la fecha, hora y el lugar de los hechos, una reseña de los mismos, los generales del o de los denunciantes, los datos personales del o de los detenidos y la determinación de la averiguación previa; dejando a cada Fiscalía, hacerlo a su criterio, la única condición era que estuviera integrada la información descrita anteriormente.

De aquí, nace el sistema de ROLES, fué creado por Manuel Acosta Cuelca, quien se desempeñaba como personal de informática en la fiscalía de Miguel Hidalgo, quien fué entrevistado y expresó: "El sistema no fué diseñado para ser institucional, era una aplicación para automatizar y agilizar el reporte que se nos solicitaba diariamente por la DGPEC en nuestra Fiscalía; únicamente, buscaba satisfacer nuestras necesidades particulares, al ser reubicado en la Fiscalía de Iztacalco, instalé mi aplicación y le hice modificaciones para que funcionara en esta nueva fiscalía; al ver los directivos de la DGPEC que el reporte entregado por Miguel Hidalgo y por Iztacalco eran idénticos, indagaron sobre el mismo, hasta contactarme; de esta forma, el sistema fué adecuado y promovido para ser usado por todas las fiscalías del Distrito Federal, logrando así una homogeneidad de información".

El sistema de roles fué diseñado en el año de 1997 y sus características, son:

- Diseñado en el lenguaje CLIPPER 5.0,
- Es una aplicación servidora de archivos.
- Contiene tablas con catálogos.
- Sólo funciona de manera local y separada para cada agencia del Ministerio Público.
- Únicamente permite utilizar la aplicación a un usuario a la vez.
- La información por Fiscalía se encuentra separada por cada una de sus agencias.
- Ambiente en línea.
- Ingreso de información de manera secuencial.

Se diseñó de esta forma porque la PGJDF no contaba con una infraestructura de red con comunicaciones remotas (MAN), sólo tenía redes de área local (LAN) para cada Fiscalía, dichas redes contaban con una topología de anillo con cable coaxial, el sistema operativo de los servidores era NOVELL y terminales con Windows 95.

Los servidores, eran equipos Pentium II a 350 MHz, con 256 Megabytes de memoria RAM y un disco duro de 40 Gigabytes y las terminales eran equipos con procesadores 486 a 66 Mhz, con 32 megabytes de memoria RAM y discos duros de 2 Gygabytes.

El sistema de ROLES es utilizado para llevar un control estadístico de todas y cada una de las averiguaciones previas iniciadas por los ministerios públicos y a su vez ser la fuente de información para la creación un reportes homogéneo de las incidencias delictivas; tomando al sistema como una caja negra, su uso se ejemplifica a través de la figura 3.

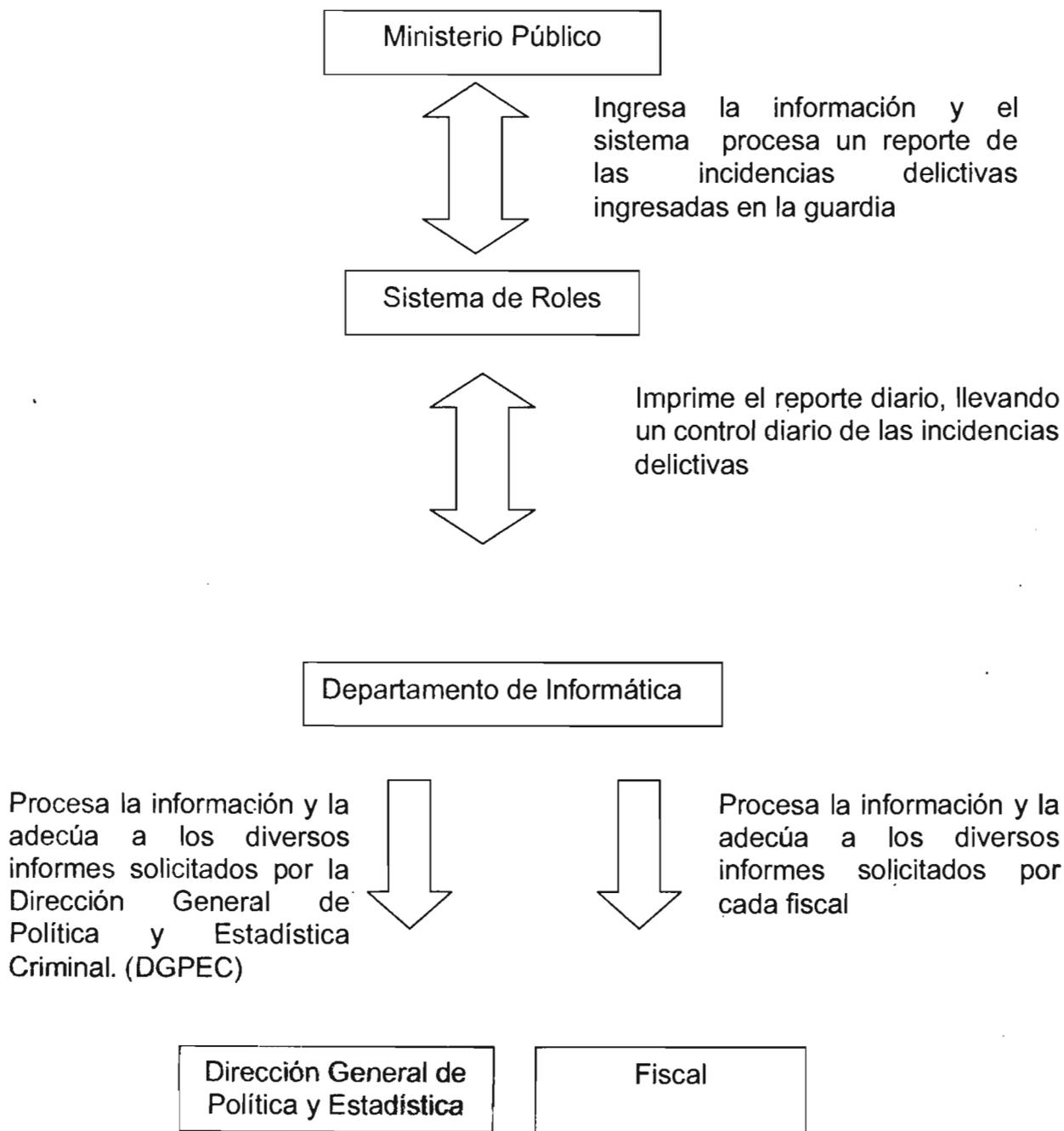


Figura 3 Esquema de la Finalidad General del sistema Roles.

El sistema de roles en su momento fué una herramienta de gran utilidad, ya que no era necesario hacer los reportes de las incidencias delictivas a mano, además de estandarizar la información; pero como era una aplicación creada para ser trabajada de manera local, adolece de seguridad e integridad en los datos; por otra parte, es una aplicación que no tiene un análisis de diseño, puesto que fué creada sólo para resolver las necesidades en una sola agencia del Ministerio Público.

Sus deficiencias, son:

- Su diseño; para ingresar o modificar algún dato dentro de la averiguación, se deben de recorrer todos los campos de uno en uno, no tiene la funcionalidad de poder posicionarse sobre el campo deseado; esto representa tiempo perdido para el usuario.
- El tamaño de la base de datos; si la tabla donde se encuentran los registros excede más de los veinticinco megabytes de tamaño, el sistema comienza a funcionar indebidamente, revolviendo la información que se ingresa con la que se tiene guardada dentro de la base de datos.
- Manejo de fechador, el sistema lo maneja por medio de una tabla que lo va incrementando automáticamente; en variadas ocasiones este sistema se salta los cambios de turno, avanzando la fecha según se encuentre dentro de la tabla que la maneja, lo cual no es fiable, ya que se salta las fechas adelantándolas varios días.
- Validación de números de averiguación previa; si dos usuarios entran a dar de alta una averiguación y le asignan en mismo número, no despliega ningún error de duplicidad de folio, sino hasta el momento en que uno de los usuarios guarda su información, acepta la petición de uno de los usuarios y al segundo se le bloquea el sistema, perdiendo su información.
- Sólo puede ser utilizado por un usuario a la vez; si dos usuarios ingresan al sistema, asignan folios diferentes y guardan su información al mismo

tiempo, el sistema no lo soporta y se bloquea, perdiendo así la información de ambos usuarios.

- Otra deficiencia ocurre cuando se está iniciando una averiguación y el sistema se encuentra en el guardado del número de la averiguación previa y la energía eléctrica falla, provocando un apagón en la terminal; el sistema bloquea el número de la averiguación, ya que al volver a ingresarlo envía un mensaje indicando que la misma ya existe, al querer modificar dicha averiguación envía el mensaje de que ésta no existe y al tratar de hacer el cambio de turno envía el mensaje de que la averiguación no cuenta con un trámite.
- El manejo de la información; es la necesidad de utilizar la información almacenada en la base de datos para la creación de nuevos reportes y esto es casi imposible, puesto que el sistema solamente guarda un histórico de la incidencia diaria; la cual, no puede ser modificada o usada para la creación de nuevos reportes; teniendo en cuenta que si el sistema falla adelantando fechas, al momento de regresarlo a la fecha actual, sustituye el informe que tenga la misma fecha perdiendo de esta manera el histórico de la incidencia.

En conclusión, el sistema actual tiene grandes deficiencias dentro de su diseño como es: tamaño de la base de datos, secuencialidad en el ingreso o modificación de la información y en el manejo de la misma para la creación de reportes; en cuanto a la integridad de la información se encuentra la duplicidad y la baja seguridad en la misma; por lo que se necesita rediseñar todo el sistema para resolver toda su problemática, además de robustecerlo en el manejo de la información adicionando nuevos filtros para la creación de diversos reportes que permitan el mejor manejo de la información.

II.2 PLANTEAMIENTO DEL PROBLEMA

En la actualidad el sistema de ROLES ha dejado de ser funcional por la creciente necesidad de hacer informes diversos y por las deficiencias vistas; de este análisis es posible afirmar que el problema es que no existe un sistema que proporcione una información real y verídica sobre la incidencia delictiva; además, la información, no puede ser transmitida a una hoja de cálculo para el manejo y vistas de la misma; agregando también, que no existen reportes o consultas que manejen rangos de fechas, delitos específicos, colonias, coordinaciones territoriales y todas las anteriores en conjunto.

Se pueden destacar los siguientes puntos a resolver del problema en cuestión:

- Duplicidad de información.
- Sólo puede trabajar un usuario a la vez.
- Los reportes son insuficientes en la actualidad.
- Limitación en el tamaño de la base de datos.
- Criterios de búsquedas insuficientes.
- Ambiente en línea.
- Baja seguridad del sistema.

II.3 OBJETIVO

El objetivo general es, crear una nueva aplicación para resolver la problemática presentada en el sistema actual, ya que el diseño del vigente sistema no cuenta con la seguridad necesaria y el manejo de la información es inadecuado para la toma de decisiones; se busca el mejoramiento del mismo, con un nuevo diseño mejorando la seguridad y el manejo de información, además de pasar de un recurso compartido a un sistema totalmente para red (cliente-servidor).

Los objetivos específicos, son:

- Crear una aplicación nueva totalmente para red, (ambiente cliente-servidor).
- Mejorar el manejo de información para la generación de reportes diversos.
- Robustecer la seguridad de la información.
- Agregar un apartado especial para el robo de vehículos.
- Consultar en línea la información.
- Exportar la información a hojas de cálculo.
- Ahorro de papel.
- Explotación de la intranet institucional.

CAPITULO III ANALISIS Y DISEÑO

III.1 ANÁLISIS

III.1.1 ANÁLISIS DE LA INFRAESTRUCTURA.

En la actualidad, la Fiscalía de Miguel Hidalgo cuenta con una red de área metropolitana de topología de estrella, con un cableado estructurado con UTP de nivel 5 y terminadores con base telefónica; con la capacidad de velocidad de transferencia de datos de 10 a 100 megabytes por segundo, en sede se cuenta con un switch Cisco con capacidad de 100 nodos y en las agencias foráneas se cuenta con concentradores de 24 nodos; cada una de las agencias tanto en la sede como en foráneas, se cuenta con un enlace dedicado por parte de Telmex y un enlace via microondas por parte de la PGJ DF, el cual sólo se utiliza en caso de fallas con el enlace de Telmex.

En cuestión de equipo de cómputo, se cuentan con 150 terminales para usuarios y 14 servidores, 30 terminales son equipos Pentium 1 a 100 Mhz con 32 Mb. De memoria RAM y discos duros de 2 Gb; 50 terminales Celeron a 500 Mhz con 32 Mb. De Ram y discos duros de 4 Gb y las 70 restantes son equipos Pentium III a 500 Mhz con 64 Mb de ram y discos duros de 10 Gb; 2 servidores Xeon con 1024 Mb de Ram y un disco duro de 100 Gb; 5 servidores Pentium III con 256 Mb de ram y un disco duro de 60 Gb y 7 servidores Pentium III con 256 Mb de Ram y un disco duro de 80Gb; todos los equipos cuentan con una tarjeta de red de 10/100 con base T.

Los servidores, cuentan con dos sistemas operativos, el primero con Linux Red Hat versión 7, conteniendo un manejador de base de datos Informix, y el segundo con Windows NT Server Versión 4 con una suite de Microsoft Office 2000 en versión "Small Busines"; las terminales Pentium I, cuentan con un sistema

operativo Windows 95 y una suite Microsoft Office 97; las demás terminales, cuentan con sistema operativo Windows 98 SE y una suite Microsoft Office 2000 "Small Business".

Los servidores con Linux, tienen la finalidad de ser el alojamiento de la base de datos para los nuevos sistemas de la PGJ DF; en ellos se encuentra la información de los sistemas SSAP y SCAMPA; mientras que los servidores con Windows NT, sólo tienen la finalidad de ser los administradores de accesos a los usuarios, servidor de impresión y servidor de archivos, en estos servidores se encuentran los sistemas de Roles, APCOM y APCMESAS, los cuales, en la actualidad están dejando de ser funcionales.

III.1.2 ANÁLISIS DE LAS HERRAMIENTAS DE DESARROLLO

Se hizo un análisis de las diversas herramientas de desarrollo para el sistema, de las cuales, se optó por utilizar, las siguientes:

LINUX

Linux, posee todas las características de los sistemas comerciales usados en los entornos de trabajo más exigentes y se encuentra actualmente en unas cotas de eficacia que le hacen dominar el mercado de los grandes servidores y, estar cada vez más presente en los equipos de los usuarios domésticos. Este nivel de prestaciones ha sido posible gracias a la disponibilidad de su código fuente, que permite a cualquier programador del mundo un acceso total a su corrección y mejora, además de ser un software libre.

Algunas de sus principales características, son:

- Multiproceso. Permite la ejecución de varias aplicaciones simultáneamente.
- Multiusuario. Distintos usuarios pueden acceder a los recursos del sistema simultáneamente aunque se trate de una instalación en una sola máquina.
- Multiplataforma. Funciona con la mayoría de plataformas del mercado: Intel 386/486/Pentium, Motorola 680, Sun Sparc,...
- Shells programables que lo convierten en el sistema más flexible que existe.
- Soporte para cualquier cantidad y tipo de dispositivos directamente en el núcleo.
- Soporte para la mayoría de sistemas de ficheros.
- También se podría citar su arquitectura modular que evita los sistemas monolíticos y permite prácticamente que el usuario se fabrique un núcleo *a la carta*. Estas características lo convierten, probablemente, en uno de los sistemas más avanzados que existen.

Para el desarrollo de este proyecto, se ha seleccionado a este sistema operativo por la gran confiabilidad que proporciona, aunando que este sistema operativo ya se encuentra en función en diversas áreas del Gobierno del Distrito Federal incluyendo a la PGJ DF.

APACHE

Se optó usar a Apache como servidor web porque es uno de los mejores servidores de Webs utilizados en la red internet desde hace mucho tiempo, únicamente le hace competencia un servidor de Microsoft, el IIS. Por lo que éste servidor es uno de los mayores triunfos del software libre; además, se debe considerar que los servidores de la PGJ DF en la actualidad, está utilizando a Linux como sistema operativo en la mayoría de sus dependencias.

Entre sus bondades se puede encontrar que es un servidor de web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos; destacando del mismo las siguientes características:

- Implementa los últimos protocolos, aunque se base en el HTTP / 1.1
- Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo y con la API de programación de módulos.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de errores y parches para solución de los mismos.

La versión actual del Apache es la 1.2.4 (1.3 ya está en beta). En la nueva versión, se incluyen características como el soporte para Windows NT y Windows 95, así como la inclusión de cuatro dígitos en las fechas para evitar los problemas del año 2000.

MYSQL

Se eligió a MySQL como administrador de bases de datos puesto que es un software libre, la documentación es muy buena e incluso en español y muchas otras aplicaciones son compatibles con esta base de datos, entre las que se encuentran sistemas de foros o portales, lo cuál nos servirá en un futuro para el desarrollo de nuevas aplicaciones.

Entre sus características podemos destacar las siguientes:

- Es muy utilizada en servidores, como complemento a PHP, otro lenguaje de código abierto. Y, por supuesto, muy conocida en el mundo Linux, entre programadores "liberales".
- Se basa en el lenguaje SQL (Structured Query Language).
- Se clasifica dentro de las bases de datos relacionales.

- Es compatible con muchos lenguajes, aunque el dúo perfecto lo forma con PHP.
- Ideal para bases de datos en internet.
- Rápida, fiable, segura.

PHP

PHP (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje de "código abierto" interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor.

Las principales características por las que se eligió a PHP como herramienta de desarrollo, son:

- El código se ejecuta en el servidor a diferencia de otros
- Su uso es simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales.
- Puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies.
- Puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más.
- Soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros.
- También, tiene la posibilidad de usar programación procedimental o programación orientada a objetos.

- No se encuentra limitado a resultados en HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También, puede presentar otros resultados, como XHTML y archivos XML. PHP puede autogenerar estos archivos y almacenarlos en el sistema de archivos en vez de presentarlos en la pantalla.

Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir un interfaz vía web para una base de datos es una tarea simple con PHP. Las siguientes bases de datos, están soportadas actualmente:

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

Soporta ODBC (el Estándar Abierto de Conexión con Bases de Datos), así que puede conectarse a cualquier base de datos que soporte tal estándar; cuenta con soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros.

DREAMWEAVER

Dreamweaver es un editor HTML profesional para diseñar, codificar y desarrollar sitios, páginas y aplicaciones Web. Tanto si se desea controlar manualmente el código HTML o si se prefiere trabajar en un entorno de edición visual, proporcionando útiles herramientas que mejorarán la creación Web.

Las funciones de edición visual de Dreamweaver permiten crear páginas de forma rápida, sin escribir una sola línea de código. No obstante, se puede interactuar utilizando la forma visual de creación anexando código manualmente para la adecuación del sitio web a las necesidades de cada proyecto en cuestión.

También, incluye numerosas herramientas y funciones relacionadas con la codificación para crear aplicaciones Web dinámicas basadas en bases de datos empleando lenguajes de servidor como ASP, ASP.NET, ColdFusion Markup Language (CFML), JSP y PHP; facilitando de esta manera la creación de Páginas dinámicas con conexión a bases de datos.

III.1.3 ANÁLISIS DE REQUERIMIENTOS

III.1.3.1 COSTO DE DESARROLLO

En lo que respecta al costo del desarrollo del Sistema de Incidencias Delictivas, lo que se debe tomar en cuenta, son los recursos necesarios para llevar a buen término la construcción del sistema, que van desde los recursos tecnológicos (hardware y software) hasta los recursos humanos (personal y horas hombre de trabajo), para éste proyecto el costo de hardware y software no se incluye ya que el equipo donde se realizara el proyecto es parte de la PGJ DF y en el caso de software se utilizará Software Libre.

En cuanto a los recursos humanos y horas-hombre empleados para la realización del proyecto, se debe mencionar que se desarrollará por una sola persona y que sólo se emplearan 4 horas hombre por semana dentro del horario laborable por un lapso de 30 semanas, calculando que la hora-hombre aproximadamente se paga a 100 pesos, se tendría un cálculo de 12,000 pesos en lo que respecta a los recursos humanos, pero hay que sumar el precio de la capacitación que en este caso será de 18,000 pesos, por lo que resultaría un total de 34,000 pesos.

III.1.3.2 COSTO DE OPERACIÓN

En lo que se refiere a costos de operación, hay que considerar los recursos necesarios para que el sistema opere, desde los materiales hasta los humanos, para ello se debe tener en cuenta los equipos de cómputo y software necesarios para que el sistema opere; así mismo, se debe considerar el costo de la capacitación y el del personal necesario para dar soporte al sistema.

Primeramente, se tiene que para poder utilizar el sistema se requiere de un equipo de cómputo como servidor y de clientes para que ingresen a los recursos de la aplicación, además de que exista una red; lo cual, en este caso no es necesario calcular, ya que la PGJ DF en la Fiscalía de Miguel Hidalgo cuenta con este tipo de infraestructura; para poder usar la aplicación en los clientes es necesario un navegador Web (Netscape o Internet Explorer) en cada uno de ellos (contenido en el sistema operativo Windows 98), esto no representa gasto alguno; y en los servidores no es necesario hacer alguna modificación, ya que los recursos con los que se cuenta son suficientes (léase capítulo dos).

Para determinar el costo del software, se toma en cuenta básicamente el sistema operativo, el servidor Web, el manejador de bases de datos y un intérprete de lenguaje para desarrollo en Web, en ese sentido, la propuesta de este proyecto se basa en la utilización de Software Libre y por ello se propone a Linux como

sistema operativo, Apache como servidor Web, MySQL como manejador de bases de datos y el módulo PHP para Apache como intérprete del lenguaje .

En cuanto al personal necesario para dar soporte al sistema no será necesario erogar recursos monetarios para su capacitación ya que el Gobierno del Distrito Federal tiene planeado para el año 2006 impartir una serie cursos de software libre para los servidores públicos, que incluyen los productos antes mencionados, pero sí es necesario calcular el costo del sueldo de la persona encargada del sistema y que como mínimo deberá tener un nivel de asistente administrativo "G" y cuyo sueldo oscila alrededor de los 10 mil pesos por mes, aunque esto no debe tomarse como el costo de operación ya que estas personas se encuentran dentro del departamento de informática de cada una de las Fiscalías y áreas centrales; los cuales, además de sus tareas asignadas podrán dar soporte al sistema.

III.2 PROPUESTA DEL SISTEMA

III.2.1 PROPUESTA

Se propone la creación de una nueva aplicación con un ambiente cliente-servidor, basado en una arquitectura de tres capas y de forma modular, lo cual, ofrecerá la facilidad de anexar nuevos módulos o agregar nuevas funciones al sistema sin grandes costos; contendrá una gran diversidad de consultas para la creación de reportes, agregando la opción de poder exportar dicha información a una hoja de cálculo para el manejo de la información.

Se tomará como sistema operativo a Linux puesto que la Institución ya cuenta con este sistema, anexándole el apache como servidor Web; por cuestiones de licenciamiento no se puede utilizar su manejador de base de datos actual (informix), por lo que se propone usar MySql como el motor de base de datos por

ser software libre; como lenguaje de desarrollo de aplicaciones a PHP por la gran diversidad de opciones que ofrece para Páginas web, además de formar la mejor dupla con MySql y como auxiliar para dar vistas a la interfaz de usuario a dreamweaver.

Las ventajas que se obtendrían con el nuevo sistema, son:

- Una nueva aplicación con plataforma Web.
- Ambiente gráfico y amigable.
- Sistema modular.
- No se requiere de Licencias.
- Cambios a futuro de la aplicación de manera sencilla.
- Diseño (Cliente-Servidor)
- Ahorro de papel.
- Diversidad de consultas.
- Reportes Diversos.
- Consultas en tiempo real con módulos para directores Generales, Fiscales y responsables de agencia.

Las desventajas de la propuesta:

- Sistema no institucionalizado.
- Sólo se usará de inicio en la Fiscalía de Miguel Hidalgo

III.2.2 ALCANCE

En primera instancia, el sistema sólo será local para la Fiscalía de Miguel Hidalgo y se apegará a los siguientes lineamientos:

- La Base de Datos se diseñará en SQL con MySQL
- La aplicación se desarrollará en PHP.
- Se utilizará Apache como servidor web.
- El sistema operativo para el servidor será Linux.
- Se hará conforme a normas de usabilidad.
- Se utilizará un diseño de arquitectura de tres capas.
- Se usaran diagramas de diseño UML

El sistema incluirá las siguientes características:

- Mecanismos de seguridad para el acceso al mismo.
- Sólo abarcará la Fiscalía de Miguel Hidalgo de la PGJDF.
- Soportará la consulta en tiempo real de la información.
- Creará diversos reportes.
- Exportación de los reportes a archivos.

III.2.3 BENEFICIOS

Los beneficios que se esperan del sistema, no están relacionados con la reducción del personal en las áreas de atención, sino con la obtención de un reporte verídico y versátil para la implementación de operativos para la reducción de la incidencia delictiva en el Distrito Federal, es decir, contar con la información necesaria y verídica de cada una de las coordinaciones territoriales para una correcta toma de decisiones en la implementación de operativos de seguridad y de esta manera, reducir los índices delictivos dentro del Distrito Federal, dicha información podrá

ser consultada en cualquier momento y desde cualquier punto dentro de la red de la PGJ DF, los beneficios del sistema se enlistan a continuación:

- Aprovechamiento y explotación de los recursos de red con los que cuenta la PGJ DF.
- Ahorro de recursos en el diseño y desarrollo de sistemas propios para la gestión de averiguaciones previas.
- Ahorro en licencias para el desarrollo de nuevos sistemas basados en productos comerciales al utilizar herramientas y Software Libre
- Permitirá sustituir aplicaciones previamente desarrolladas que utilicen software comercial, con la finalidad de conseguir ahorros, por concepto de licencias con la sustitución de dichas aplicaciones.
- Contar con una infraestructura de software propia del Gobierno, que pueda modificar en el momento que así lo necesite.
- Se evitará depender de software comercial, ya que estos pueden en un momento determinado desaparecer y dejar al Gobierno sin soporte.
- Contar con información confiable de las incidencias delictivas y los trámites de las mismas.
- Se evitará depender de sistemas operativos para ejecutar las aplicaciones.
- El sistema ayudará en el aprovechamiento al máximo los recursos de hardware, ya que los clientes de la aplicación no requieren ser equipos de cómputo con grandes recursos de hardware y software, con un navegador Web, será suficiente.
- El mantenimiento y actualización del sistema será sencillo y a bajo costo puesto que se desarrollará con Software Libre.
- Contar con un control de los usuarios que ingresan al sistema y realizan acciones dentro del mismo.
- Una interfaz sencilla y amigable que facilitará la capacitación del personal que operará el sistema, toda vez que los sistemas en web utilizan pocos controles: cuadros de texto, botones de radio, cajas de confirmación, listas desplegables y botones.

- Disponibilidad de la información
- Evitar pérdida y duplicación de la información.
- Ahorro de papel ya que podrá ser consultada la información en cualquier momento por directores de área y fiscales.
- Ahorro de tiempo en la generación de reportes relativos a las incidencias delictivas.

III.3 DISEÑO

Los esquemas de diseño del sistema están basados en la metodología UML, porque proporciona una visión más completa de la forma en que trabajará el nuevo sistema antes de comenzar a desarrollarlo.

III.3.1 CASOS DE USO

ACTORES DEL SISTEMA

Un actor, es el rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

Los actores identificados para el sistema, son:

- **ADMINISTRADOR:** Puede agregar, modificar, borrar y consultar datos de los catálogos, perfiles y averiguaciones previas, además de poder cambiar de turno, generar reportes e imprimir.
- **MINISTERIO PÚBLICO:** Puede agregar, modificar y borrar datos de las averiguaciones previas, además de poder generar reportes, cambiar turno e imprimir.

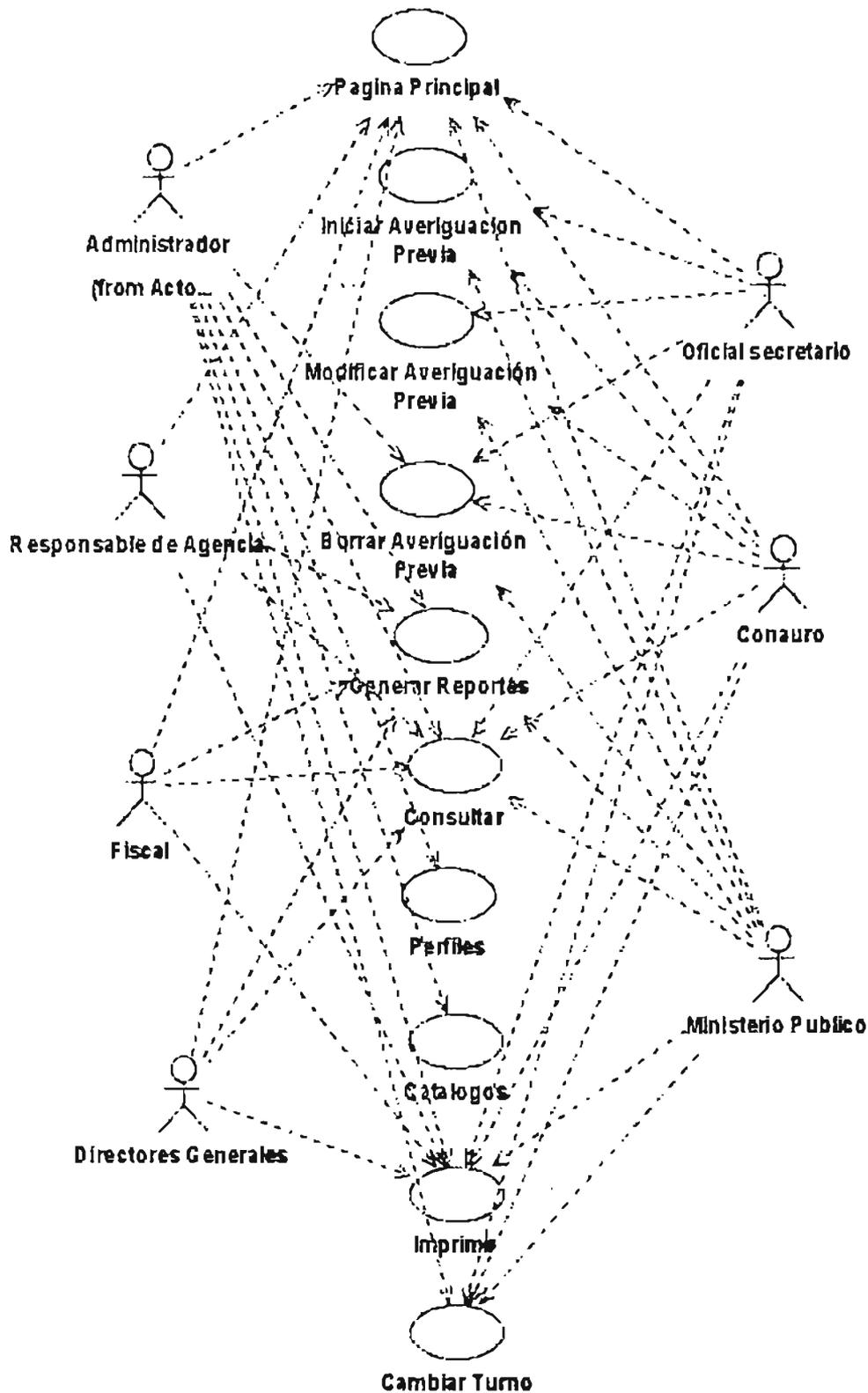
- **CONAURO:** Puede agregar, modificar y borrar datos de las averiguaciones previas, cambiar turno, consultar e imprimir.
- **OFICIAL SECRETARIO:** Puede agregar, modificar y borrar datos de las averiguaciones previas, consultar, cambiar turno e imprimir.
- **RESPONSABLE DE AGENCIA:** Puede generar reportes, consultar el rol de guardia e imprimir.
- **FISCAL:** Puede generar reportes, consultar el rol de guardia e imprimir.
- **DIRECTORES GENERALES:** puede generar reportes, rol de guardia e imprimir.

CASOS DE USO

Un diagrama de caso de uso es la representación de la interacción que tiene un actor con una tarea específica dentro del sistema.

Se presentarán a continuación, únicamente los casos de uso principales del sistema, si se desea consultar los demás casos de uso puede hacerlo en el anexo I del presente trabajo.

CASOS DE USO GENERAL DEL SISTEMA



III.3.1.3 FUNCIONES DEL SISTEMA

No.	FUNCIÓN
R1	Validar Usuario
R2	Asignar permisos de usuario
R3	Salir de Sesión
Módulo de Iniciar Averiguación Previa	
R4	Agregar datos de A.P.
R5	Agregar datos de Denunciante
R6	Agregar datos de Detenido
R7	Agregar datos de vehículo
Módulo de Modificar Averiguación Previa	
R8	Modificar datos de A.P.
R9	Modificar datos de Denunciante
R10	Modificar datos de Detenido
R11	Modificar datos de vehículo
Módulo de Borrar Averiguaciones Previas	
R12	Borrar datos de A.P.
R13	Borrar datos de Denunciante
R14	Borrar datos de Detenido
R15	Borrar datos de vehículo
Módulo de Generar Reportes	
R16	Generar Reporte
Módulo de Consultar	
R17	Consultar A.P.
R18	Consultar denunciante
R19	Consultar Responsable
R20	Consultar Rol de Guardia
Módulo de Perfiles (Usuarios)	
R21	Agregar perfil
R22	Modificar perfil
R23	Borrar perfil
R24	Consultar perfil
Módulo de Catálogos	
R25	Agregar en catálogos
R26	Modificar en catálogos
R27	Borrar en catálogos
R28	Consultar en catálogos
Módulo de imprimir	
R29	Imprimir Reporte generado
R30	Imprimir rol de guardia
R31	Imprimir consulta
Módulo de Cambiar turno	
R32	Validar contraseña
R33	Cambiar turno

DESCRIPCIÓN DE LOS CASOS DE USO.

Caso de uso: Página Principal
Nombre: Página Principal
Actores: Todos
Propósito: Validar usuarios, asignar permisos y entrar a la aplicación
Tipo: Primario
Descripción: Cada vez que entra cualquier usuario accede a esta página, la cual valida el usuario y muestra los contenidos dependiendo de su perfil.
Referencias cruzadas: R1,R2

Cursos de eventos: **Página Principal.**

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a la Página principal del sistema	2	Muestra la Página y solicita el usuario y password de acceso a la aplicación
3	El usuario proporciona los datos de usuario y password	4	El sistema valida al usuario consultando su perfil dentro del sistema, asigna los permisos y muestra la información dependiendo del perfil
		5	Enlaza al usuario a la página principal de trabajo.
		6	Si la contraseña es incorrecta muestra un mensaje y la vuelve a solicitar

Caso de uso: Iniciar Averiguación Previa
Nombre: Iniciar Averiguación Previa
Actores: Ministerio público, Oficial secretario y Conauro
Propósito: Agregar un registro de averiguación previa al sistema
Tipo: Primario
Descripción: Cada vez que el usuario accede a este caso, es para agregar el registro de una nueva averiguación previa al sistema
Referencias cruzadas: R3,R4,R5,R6,R7

Cursos de eventos: **Iniciar Averiguación Previa**

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Muestra la Página de la opción deseada con el formulario de llenado para agregar la averiguación previa al sistema
3	El Usuario llena los datos requeridos por el sistema para anexarlos a la base de datos y selecciona guardar	4	Valida los datos y pregunta al usuario si son correctos

No.	Acción del actor	No.	Respuesta del sistema
5	El usuario confirma que son correctos los datos ingresados	6	El sistema guarda la información en la base de datos mandando un aviso al usuario y regresa a la Página principal de trabajo
7	El usuario selecciona que no son correctos los datos	8	Regresa al formulario para modificar los datos incorrectos

Caso de uso: Modificar Averiguación Previa
Nombre: Modificar Averiguación Previa
Actores: Ministerio público, Oficial secretario y Conauro
Propósito: Modificar datos de un registro de averiguación previa almacenado en el sistema
Tipo: Secundario
Descripción: Cada vez que el usuario accede a este caso, es para modificar algún dato dentro de la Averiguación previa guardada en el sistema.
Referencias cruzadas: R3,R8,R9,R10,R11

Cursos de eventos: **Modificar Averiguación Previa**

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Muestra la Página de la opción deseada y solicita el número de la averiguación previa
3	Ingresa el número de la averiguación previa	4	Localiza el registro y lo muestra en pantalla
5	El usuario selecciona y modifica los datos deseados	6	Valida los datos y pregunta al usuario si son correctos
7	El usuario confirma que son correctos los datos	8	El sistema modifica y guarda la información en la base de datos mandando un aviso al usuario y regresa a la Página principal de trabajo
9	El usuario selecciona que los datos no son correctos	10	Regresa al formulario para modificar los datos incorrectos

Caso de uso: Borrar Averiguación Previa
Nombre: Borrar Averiguación Previa
Actores: Administrador, Ministerio público, Oficial secretario y Conauro
Propósito: Borrar un registro de averiguación previa en el sistema
Tipo: Secundario
Descripción: Cada vez que el usuario accede a este caso, es para borrar el registro de una averiguación previa en el sistema
Referencias cruzadas: R3,R12,R13,14,R15

Caso de uso: Consultar
Nombre: Consultar
Actores: Administrador, Ministerio público, Oficial secretario, conauro, Fiscal, Directores generales y Responsable de agencia
Propósito: Consultar el rol de guardia, denunciante, probable responsable y averiguaciones previas
Tipo: Secundario
Descripción: Cada vez que el usuario accede a este caso, es para consultar dentro del sistema el rol de guardia de determinada fecha, denunciante, probables responsables y averiguaciones previas
Referencias cruzadas: R3, R17,R18,R19,R20,R30

Cursos de eventos: Consultar

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Muestra la Página de la opción deseada donde aparecen los diversos criterios de consulta.
3	El usuario selecciona y llena los criterios de búsqueda para la consulta según sean sus necesidades	4	Valida los criterios y busca en el sistema dicha información, dándole un formato la presenta en pantalla y despliega la opción de imprimir dicha consulta o regresar a la página principal de trabajo.
5	El usuario selecciona Imprimir	6	Imprime dicho reporte, mandando un mensaje al usuario y regresa a la Página principal de trabajo.
		7	Regresa a la Página principal de trabajo
8	El usuario selecciona regresar	9	Regresa a la Página principal de trabajo

Caso de uso: Perfiles
Nombre: Perfiles
Actores: Administrador
Propósito: Agregar, modificar, borrar, consultar y dar permisos a los diversos usuarios del sistema
Tipo: Primario
Descripción: Cada vez que el usuario accede a este caso, podrá agregar, borrar, modificar y consultar a los usuarios dados de alta en el sistema, además de asignar los permisos a los que tiene derecho cada usuario.
Referencias cruzadas: R3, R21,R22,R23,R24

Cursos de eventos:

Perfiles

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Muestra la Página de la opción deseada donde aparecen las diversas opciones a seleccionar
3	El usuario selecciona la opción agregar	4	Muestra el formulario con los datos requeridos para dar de alta a un usuario
5	El usuario llena el formulario con los datos requeridos por el sistema y le da la opción de guardar	6	El sistema valida los datos y los agrega a la base de datos
		7	Despliega un mensaje al usuario avisando el guardado exitoso del registro y regresa a la pantalla principal de trabajo
8	El usuario selecciona la opción de modificar	9	Solicita el nombre del usuario para la localización del registro
10	El usuario proporciona el nombre de usuario en cuestión	10	Localiza y presenta la información en pantalla para su modificación
11	El usuario modifica la información deseada y selecciona guardar	12	Envía un mensaje de confirmación de que los datos son correctos.
13	El usuario acepta el mensaje de confirmación	14	El sistema guarda los cambios y despliega un mensaje informando que se guardo exitosamente.
		15	Regresa a la Página principal de trabajo
16	El usuario rechaza el mensaje de confirmación	17	El sistema regresa al formulario para que el usuario vuelva a modificar la información errónea.
18	El usuario elige la opción de borrar usuario	19	Solicita el nombre del usuario para la localización del registro
20	El usuario proporciona el nombre de usuario en cuestión	21	Localiza y presenta la información en pantalla para su borrado
22	El usuario elige la opción borrar	23	Solicita una clave de seguridad para el borrado de dicho registro
24	El usuario da la clave de seguridad	25	El sistema borra el registro en cuestión
		26	Envía mensaje de borrado exitoso y regresa a la pantalla principal de trabajo
27	El usuario no proporciona la clave de seguridad	28	El sistema no borra el registro
		29	Envía mensaje de clave errónea y regresa a la pantalla principal de trabajo
30	El usuario elige la opción de consultar	31	Solicita el nombre del usuario para la localización del registro
32	El usuario proporciona el nombre de usuario en cuestión	33	Localiza y presenta la información en pantalla
34	El usuario elige la opción de limpiar	35	Limpia la pantalla y solicita nuevamente el nombre de usuario para su localización
36	El usuario elige la opción de regresar	37	El sistema regresa a la pantalla principal de trabajo.

Caso de uso: Catálogos
Nombre: Catálogos
Actores: Administrador
Propósito: Agregar, modificar, borrar, y consultar los datos de la información catalogada dentro del sistema
Tipo: Primario
Descripción: Cada vez que el usuario accede a este caso, podrá agregar, borrar, modificar y consultar la diversa información catalogada en el sistema (delegaciones, coordinaciones territoriales, colonias, delitos, modalidades del delito, turnos, etc.).
Referencias cruzadas: R3, R25, R26, R27, R28

Cursos de eventos: Catálogos

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando el administrador accede a esta opción del sistema	2	Muestra la Página de la opción deseada donde aparecen las diversas opciones a seleccionar (delegaciones, coordinaciones, colonias, tipo de A.P., determinación de A.P., remitente, delito, modalidad de delito, marca de vehículo, submarca de vehículo, color de vehículo y turno)
3	El administrador selecciona agregar en catálogos	4	Muestra el formulario con los datos requeridos por el sistema para agregar dentro de catálogos (dependiendo del catálogo)
5	El administrador llena el formulario y selecciona guardar	6	Valida la información y Despliega un mensaje al usuario avisando el guardado exitoso del registro
		7	Regresa a la página principal de trabajo de catálogos
8	El administrador llena el formulario y selecciona limpiar	9	Borra la información ingresada al formulario y espera el nuevo llenado del mismo
10	El administrador selecciona modificar en catálogos	11	Solicita el nombre dependiendo del tipo de catálogo para localizar el registro
12	El administrador proporciona el nombre	13	Localiza el registro y lo muestra en pantalla para modificación
14	El administrador modifica la información y selecciona modificar	15	Valida la información y modifica el registro
		16	Regresa a la Página principal de trabajo de catálogos
17	El administrador selecciona borrar	18	Solicita el nombre (dependiendo del catálogo) para localizar el registro
19	El administrador proporciona el nombre (dependiendo del catálogo)	20	Localiza el registro, valida relación del mismo con otras tablas y lo muestra en pantalla
21	El administrador selecciona borrar	22	Solicita una contraseña de seguridad
23	Da la contraseña requerida	24	Valida la contraseña
		25	Borra el registro si no se encuentra relacionado con otras tablas y la contraseña es válida

No.	Acción del actor	No.	Respuesta del sistema
		26	Regresa a la página principal de trabajo de catálogos
		27	Si se encuentra relacionado con otras tablas despliega un aviso al usuario indicando que no se puede borrar porque existe información relacionada
		28	Regresa a la página principal de trabajo de catálogos
		29	Si la contraseña no es valida no borra el registro envía un mensaje de "contraseña invalida"
		30	Regresa a la página principal de trabajo de catálogos
31	El administrador selecciona consultar	32	Solicita el nombre (dependiendo del catálogo) para localizar el registro
33	El administrador proporciona el nombre (dependiendo del catálogo)	34	Localiza el registro y lo muestra en pantalla
35	Selecciona regresar	36	Regresa a la página principal de trabajo de catálogos

Caso de uso: Imprimir
Nombre: Imprimir
Actores: Oficial Secretario, Conauro, Ministerio Público, Responsable de Agencia, Fiscal, Directores Generales, Administrador
Propósito: Imprimir consultas, rol de guardia y reportes generados en el sistema
Tipo: Primario
Descripción: Cada vez que el usuario accede a este caso, podrá imprimir las consultas, rol de guardia y reportes generados por el sistema
Referencias cruzadas: R3,R29,R30,R31

Cursos de eventos: Imprimir

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Muestra la información en pantalla para consulta del usuario
3	El usuario selecciona la opción imprimir	4	Imprime la información desplegada en pantalla
		5	Regresa al área de trabajo donde se encuentre el usuario
	El usuario selecciona la opción regresar	6	Regresa al área de trabajo donde se encuentre el usuario

Caso de uso: Cambiar Turno
Nombre: Cambio de Turno
Actores: Oficial Secretario, Conauro, Ministerio Público, Administrador
Propósito: Cambiar el turno que se encuentra en la guardia, generar rol de la guardia que termina y regenerar la información del rol de guardia para el turno que entra.
Tipo: Primario
Descripción: Cada vez que el usuario accede a este caso, podrá cambiar el turno de la guardia
Referencias cruzadas: R3, R32,R33

Cursos de eventos: Cambiar Turno

No.	Acción del actor	No.	Respuesta del sistema
1	El caso de uso inicia cuando los usuarios acceden a esta opción dentro del sistema	2	Solicita contraseña de seguridad
3	El usuario proporciona la contraseña	4	Si la contraseña es correcta, actualiza la información y cambia de turno
		5	Genera rol de guardia
		6	Despliega un mensaje al usuario "Actualizando Información"
		7	Regresa a la Página principal de trabajo
		9	Si la contraseña no es valida no actualiza información, ni cambia de turno
		10	Envía mensaje de "contraseña invalida, vuelva a intentar"
		11	Regresa a la solicitud de contraseña
12	El usuario elige la opción de regresar	13	Regresa a la página principal de trabajo

III.3.2 COMPORTAMIENTO DEL SISTEMA.

El comportamiento del sistema es una descripción de lo que el sistema realiza, sin importar, cómo lo hace; los contratos representan un documento útil para describir el comportamiento del sistema en función del cambio de estado que experimenta el sistema cuándo se realiza una operación.

CONTRATOS.

Ayudan a definir el comportamiento del sistema, describen el efecto de las operaciones sobre el sistema, a continuación, se dá la descripción de los contratos de caso de uso "página principal", los faltantes los puede consultar en el anexo II.

Contratos para el caso de uso: Página Principal

Contrato: valida_usuario.	
Nombre	valida_usuario
Responsabilidades	Dar acceso al sistema.
Tipo	Sistema.
Referencias cruzadas	R1.
Notas	Debe de existir el usuario en la base de datos.
Excepciones	Enviará mensaje de usuario no registrado si no existe en la base de datos.
Salida	Consultará el perfil del usuario para asignar permisos.
Pre-condiciones	El usuario debe de estar dado de alta en el sistema.
Post-condiciones	Asignará permisos al usuario.

Contrato: Asignar_permisos_usuario.	
Nombre	Asigna_permiso
Responsabilidades	Dar acceso a las diversas áreas de trabajo dependiendo del perfil asignado.
Tipo	Sistema.
Referencias cruzadas	R2.
Notas	Debe de existir el usuario en la base de datos.
Excepciones	Enviará mensaje de usuario no registrado si no existe en la base de datos.
Salida	Enviará al usuario a su área de trabajo según el perfil.
Pre-condiciones	El usuario debe de estar dado de alta en el sistema.
Post-condiciones	Mostrará la Página principal de trabajo dependiendo de su perfil.

Contrato: Salir de Sesión()	
Nombre	Salir_sesion
Responsabilidades	Salir del sistema.
Tipo	Sistema.
Referencias cruzadas	R3
Notas	Se debe de encontrar dentro del sistema.
Excepciones	
Salida	Enviará al usuario a la pantalla de validación de usuario.
Pre-condiciones	El usuario debe de estar dentro del sistema.
Post-condiciones	Mostrará la Página de validación de usuarios.

III.3.3 DIAGRAMÁS DE SECUENCIAS

El diagrama de secuencia, forma parte del modelado dinámico del sistema. Se modelan las llamadas entre clases desde un punto concreto del sistema. Es útil para observar la vida de los objetos en sistema, identificar llamadas a realizar o posibles errores del modelado estático, que imposibiliten el flujo de información o de llamadas entre los componentes del sistema.

DIAGRAMA GENERAL DE SECUENCIAS

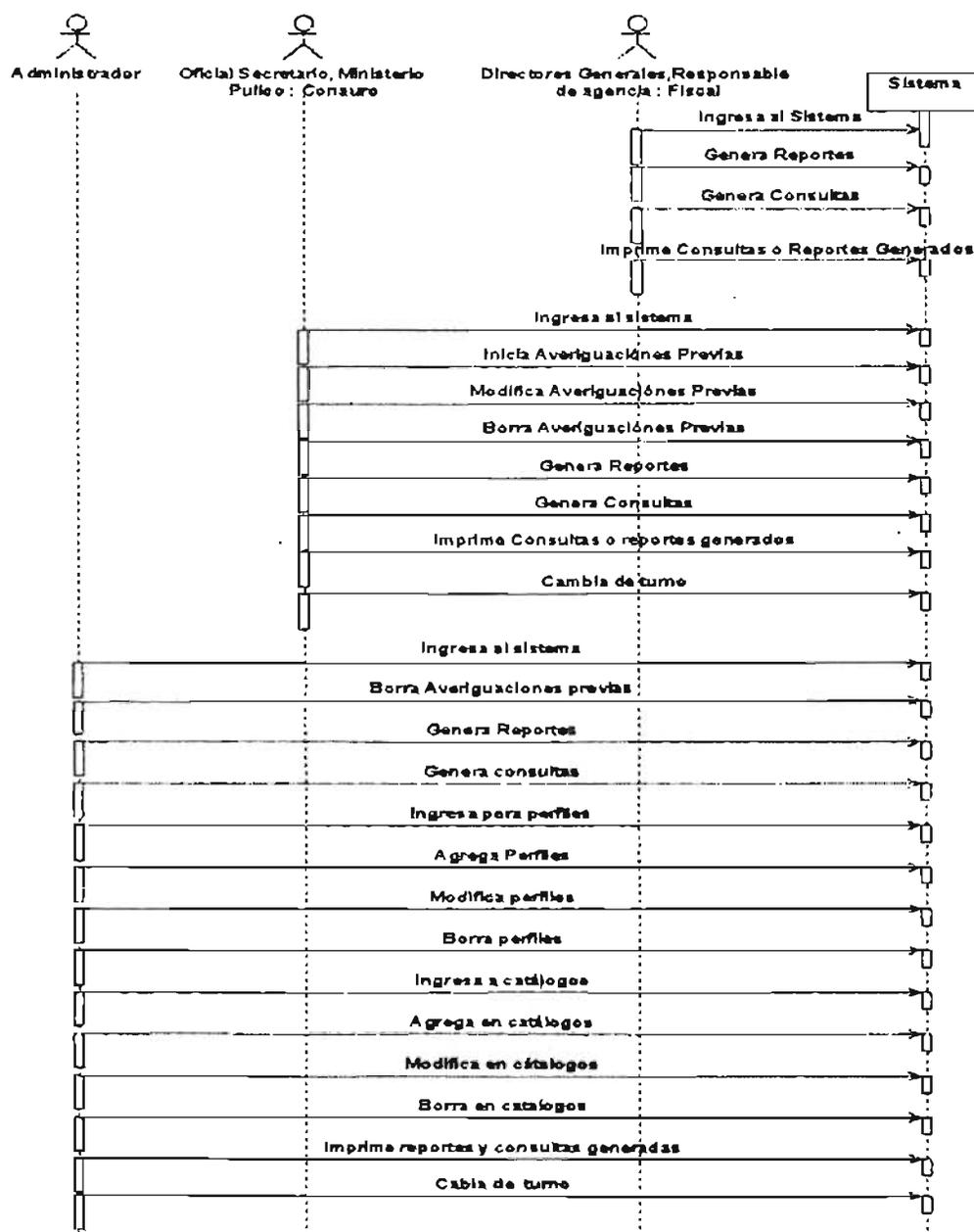
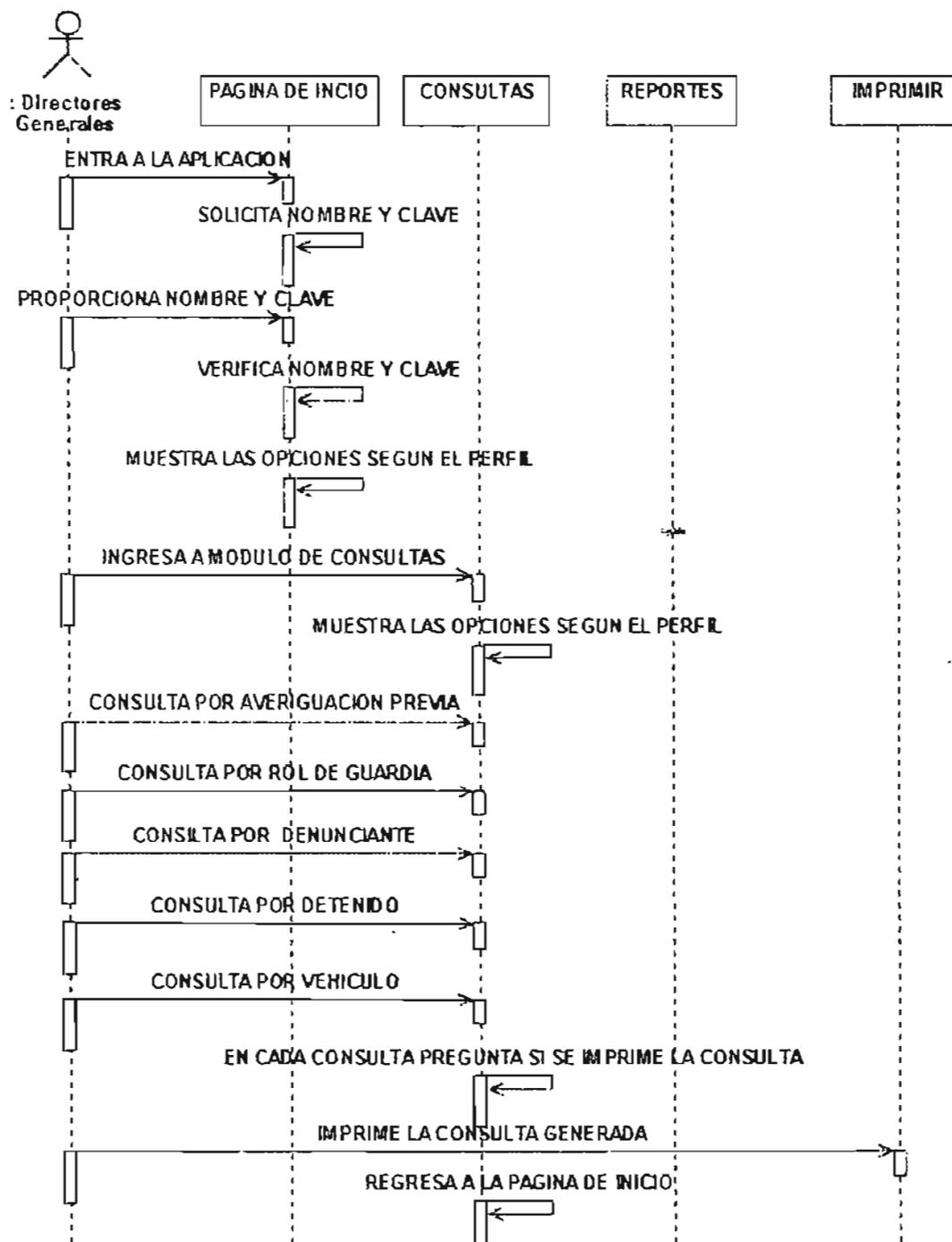


DIAGRAMA DE SECUENCIAS DE DIRECTORES GENERALES, RESPONSABLE DE AGENCIA Y FISCAL



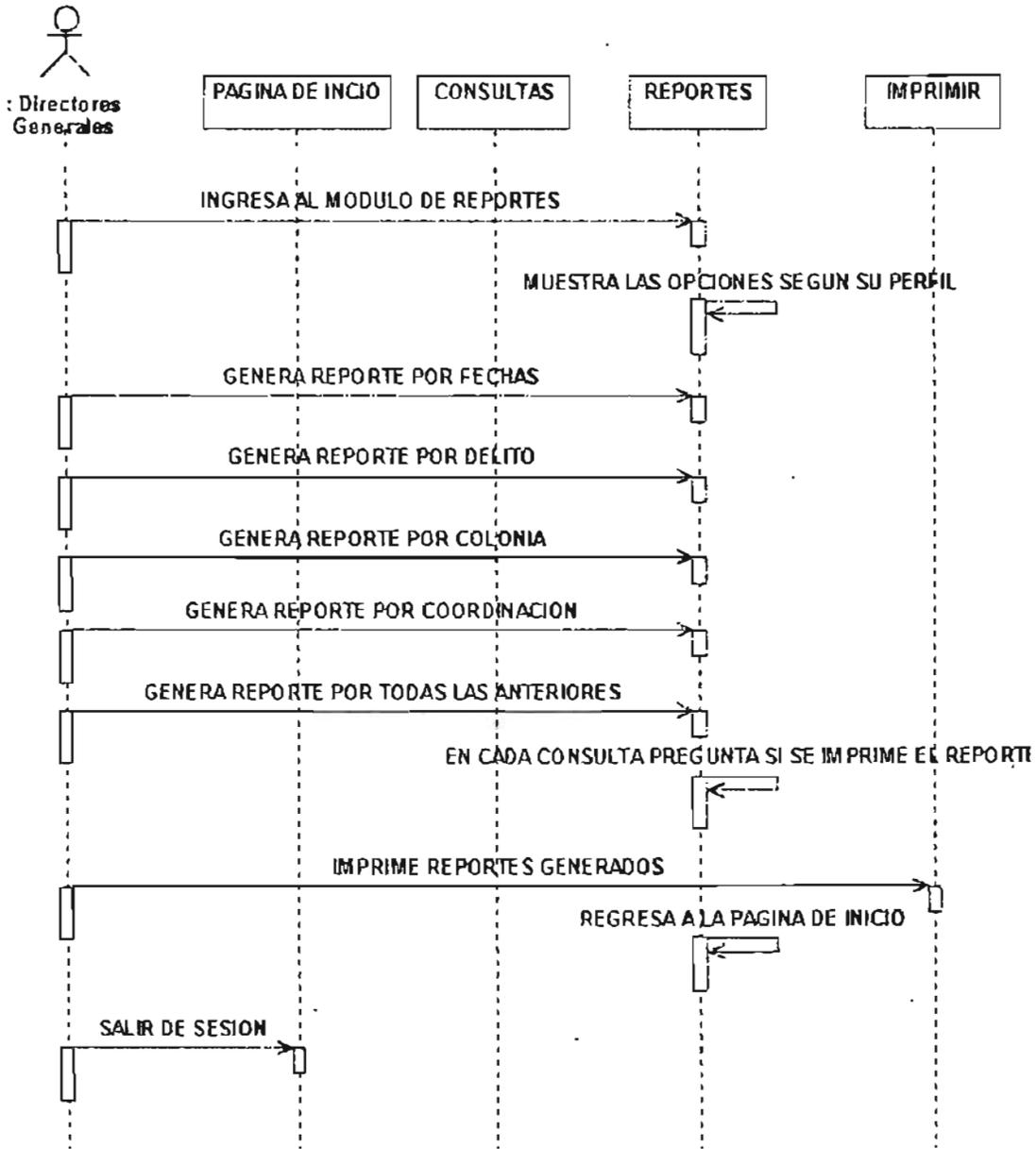
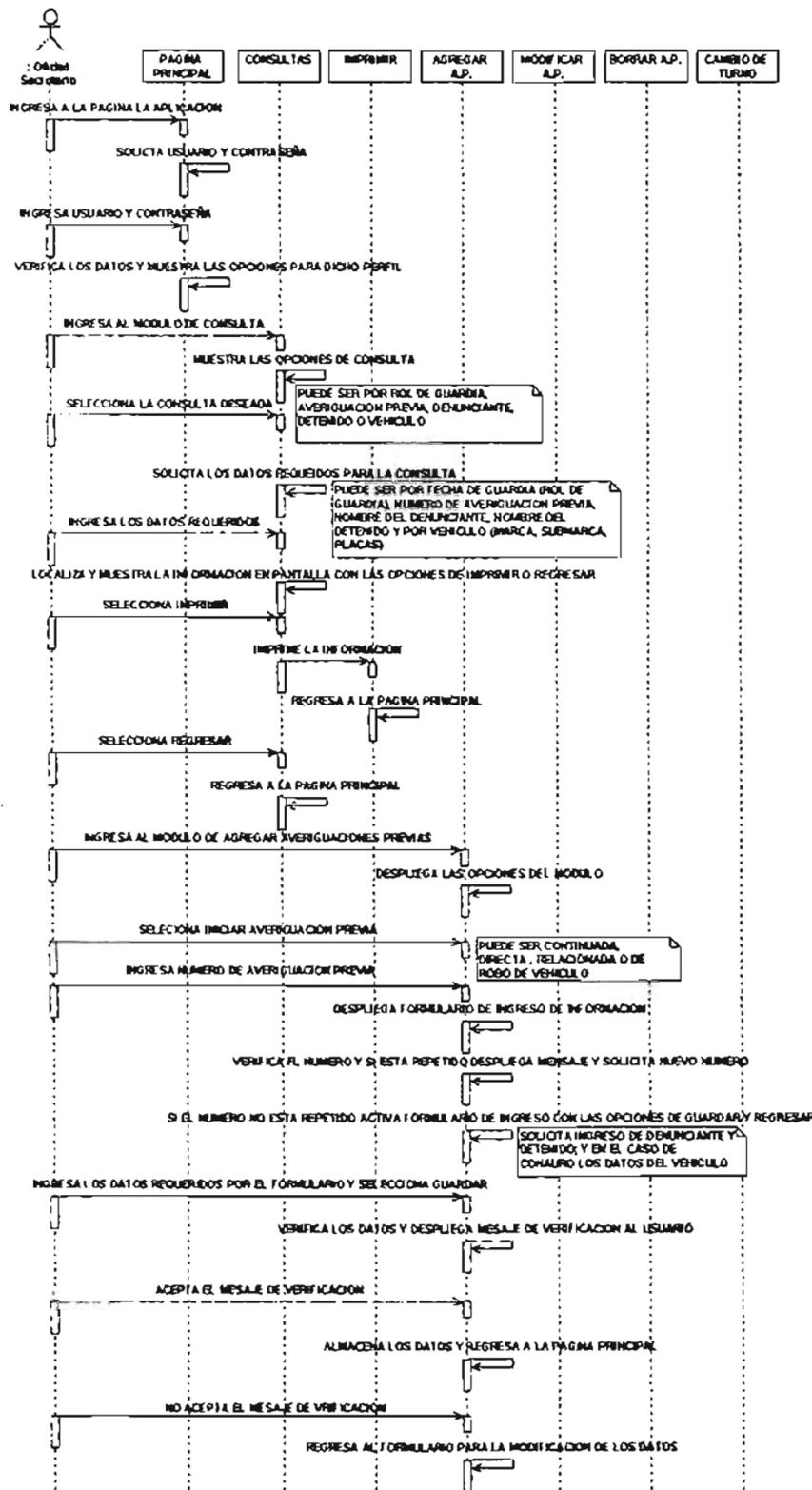
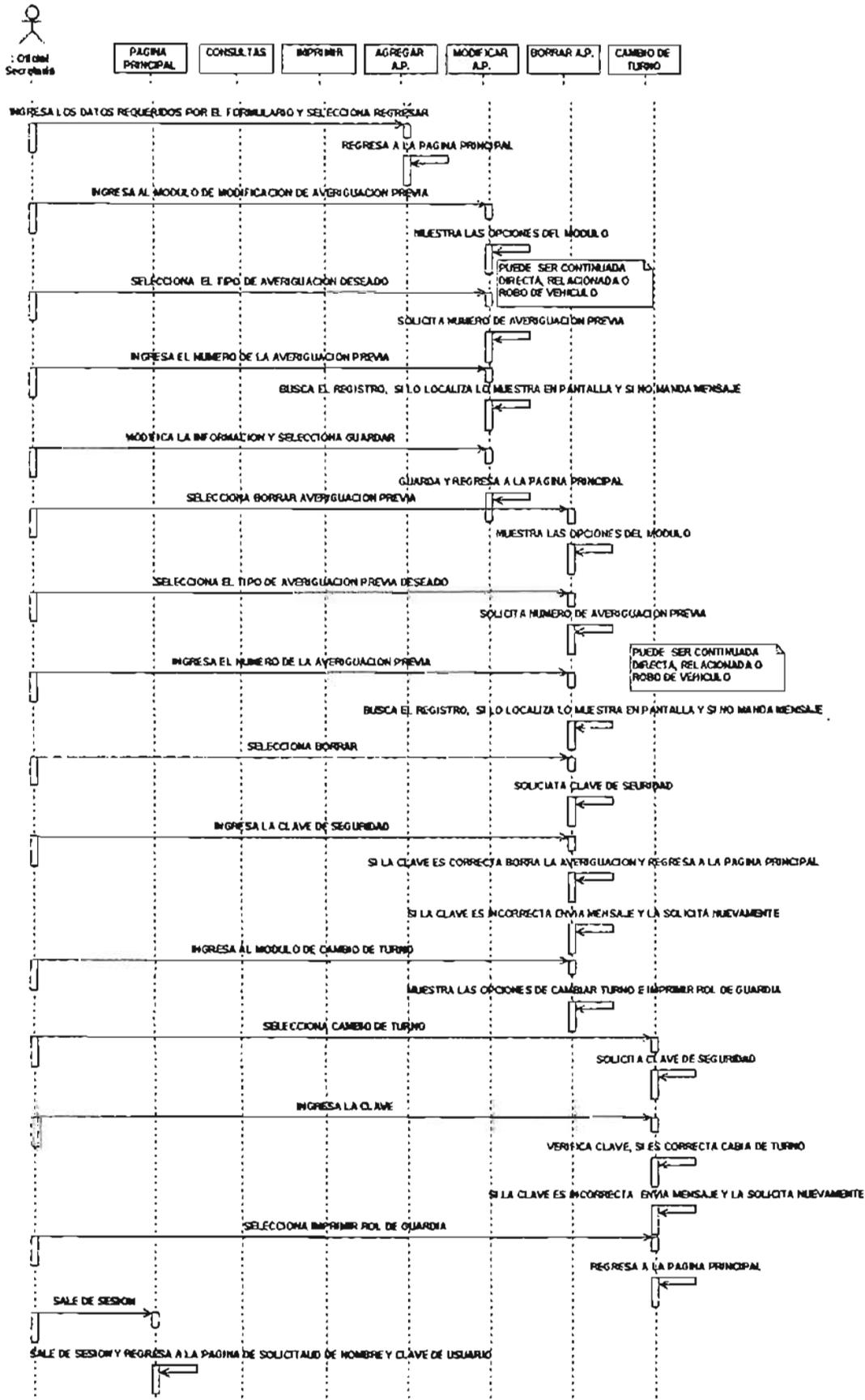


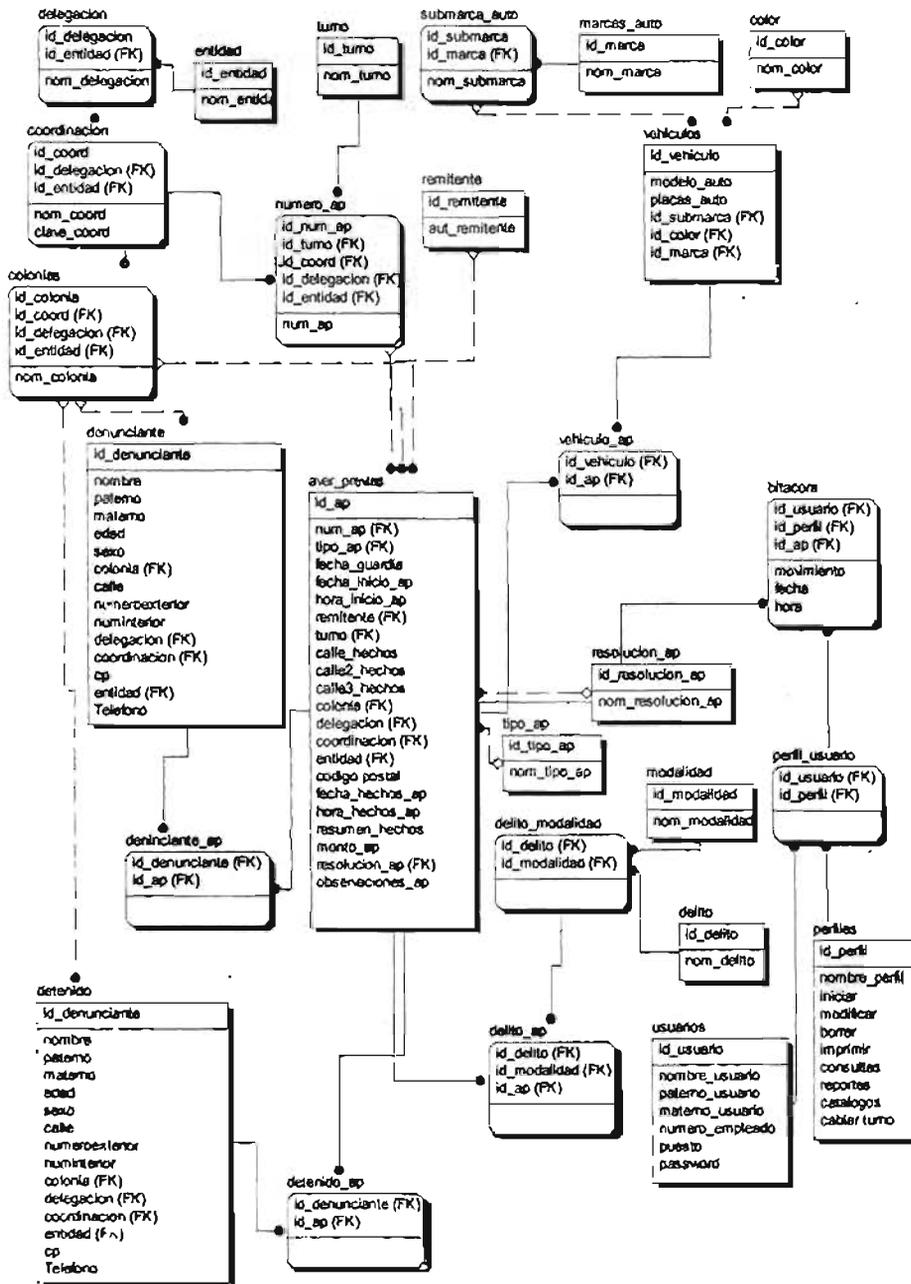
DIAGRAMA DE SECUENCIAS DE OFICIAL SECRETARIO





III.3.4 BASE DE DATOS

DIAGRAMA ENTIDAD RELACION DE LA BASE DE DATOS



El diccionario de datos lo podrá consultar en el anexo III.

III.3.5 INTERFASE DE USUARIO Y REGLAS DE NEGOCIO.

III.3.5.1 ESQUEMÁS DE NAVEGACIÓN.

Se presentarán a continuación, los esquemas de navegación de los módulos principales del sistema, los esquemas de los submódulos se encuentran contenidos en el anexo IV.

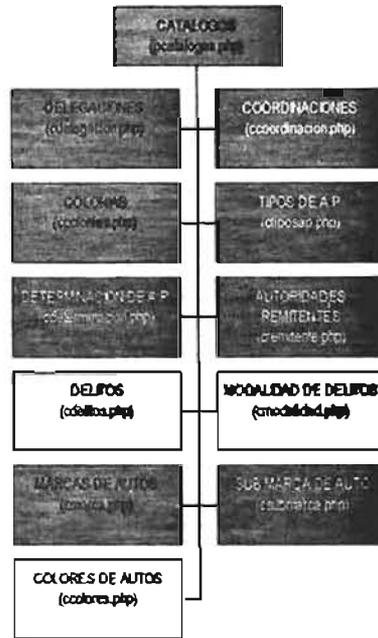
ESQUEMA GENERAL DEL SISTEMA.



ESQUEMA DEL MÓDULO DE AVERIGUACIONE PREVIAS.



ESQUEMA DEL MÓDULO DE CATÁLOGOS.



ESQUEMA DEL MÓDULO DE CONSULTAS.

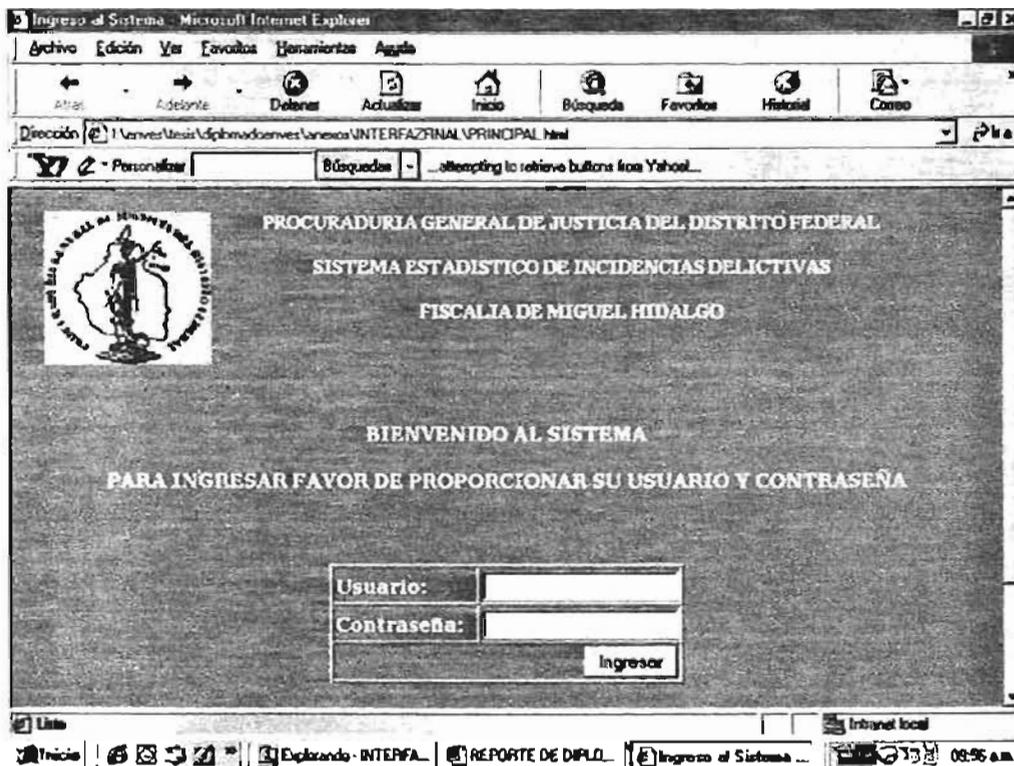


ESQUEMA DEL MÓDULO DE REPORTES.



III.3.5.2 INTERFASE DE USUARIO Y REGLAS DE NEGOCIO

ACCESO AL SISTEMA

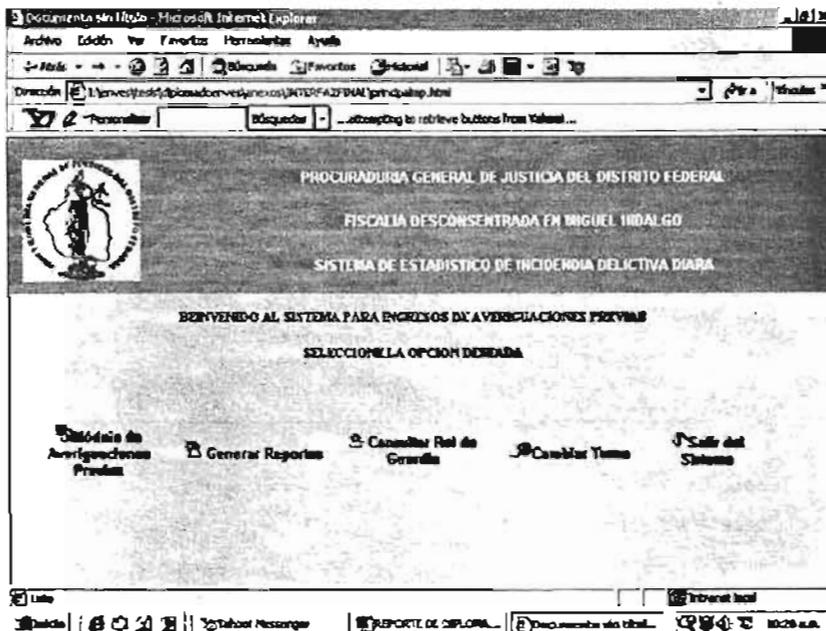


Reglas de negocio:

- Sólo pueden ingresar usuarios registrados.
- Se asignará un perfil a cada usuario registrado.
- Se presentaran diferentes áreas de trabajo dependiendo del perfil asignado a cada usuario.

PÁGINA PRINCIPAL

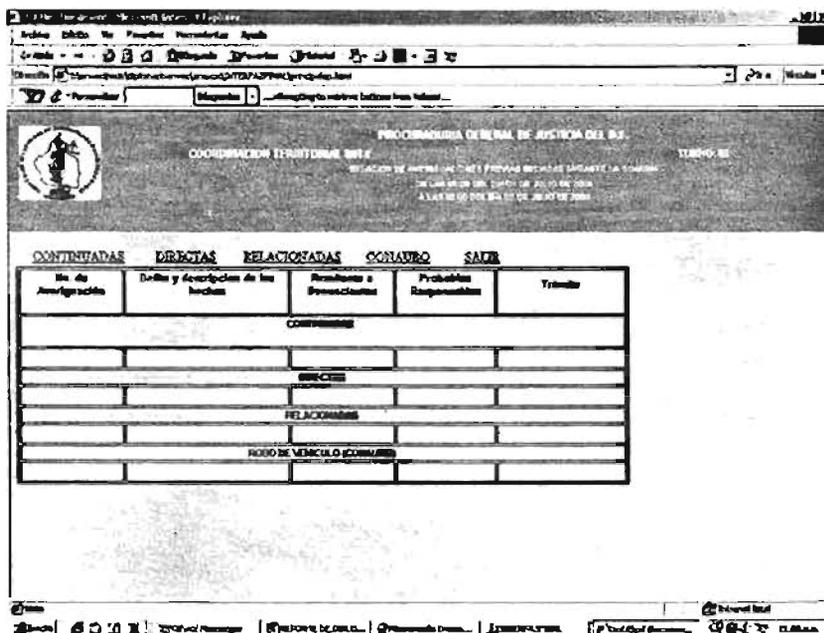
El sistema mostrará la página principal de trabajo para cada uno de los usuarios, dependiendo del perfil con que se haya ingresado, para ejemplificarlo se muestra la pantalla principal de trabajo para los ministerios públicos.



Reglas de negocio:

- Sólo mostrará las opciones del perfil con que se haya ingresado al sistema.

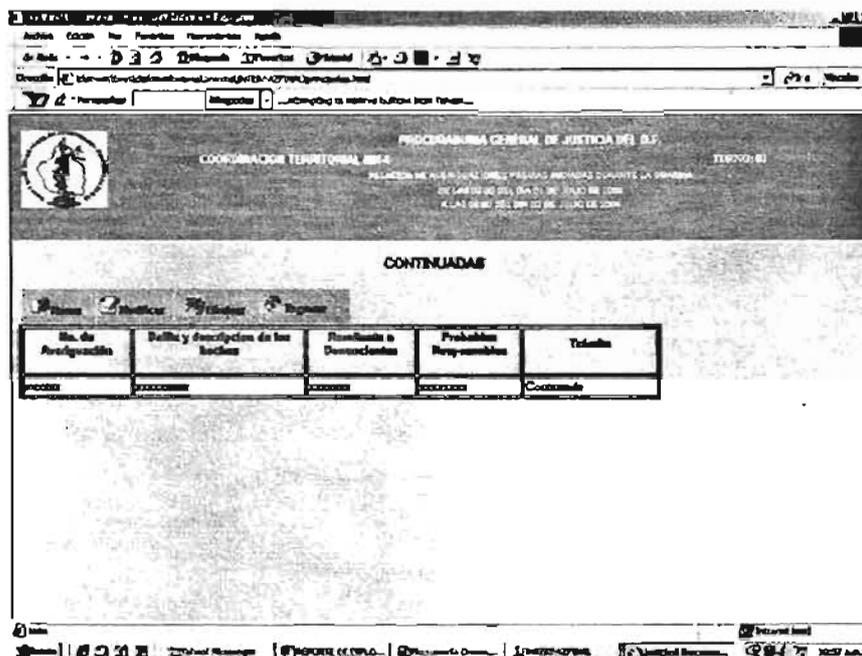
Página principal de trabajo del módulo de averiguaciones previas (iniciar, modificar y borrar)



Reglas de negocio:

- La coordinación territorial se actualizará dependiendo de la asignación que tenga el usuario en su perfil.
- El turno se actualizará automáticamente tomando la información del turno activo.
- La fecha se actualizará automáticamente tomando la fecha corriente del momento en que se hace el cambio de turno.
- Los datos de las averiguaciones del área de trabajo se actualizarán cada vez que se haga un cambio en las averiguaciones previas (alta, baja, cambio de averiguaciones y cambio de turno).

Página principal para agregar Averiguaciones Previas. (Continuada, directa, relacionada o robo de vehículo)



Reglas de negocio:

- Se actualizará automáticamente la información de las averiguaciones previas que se han modificado, agregado o borrado, en el área de trabajo para el tipo de la averiguación.

Después de seleccionar la opción agregar, se mostrará la siguiente ventana con el formulario para el ingreso de la nueva averiguación previa.

COORDINACIÓN TERRITORIAL N°14
RELACION DE AVERIGUACIONES PREVIAS INICIADAS DURANTE LA SESIÓN DE LA JUNTA DEL CRUP DE JULIO DE 2004
A 11 PM 00:00, DIA 04 DE JULIO DE 2004

DATOS GENERALES DE LA AVERIGUACION PREVIA			
NUMERO DE AVERIGUACION PREVIA	TIPO DE RP	FECHA DE INICIO	HORA DE INICIO
14-1-11-0000-04-06	<input type="radio"/> CD <input type="radio"/> SO	01/06/04	17:50
CALLE		DELITOS AGREGADOS	
Agregar		ROBO A CASA HABITACION SIN VIOLENCIA	
DATOS GENERALES DEL LUGAR DE LOS HECHOS			
CALLE		NUMERO EXTERIOR	NUMERO INTERIOR
COLORS		DELEGACION	C.P.
ANZURES		MIGUEL HIDALGO	
ENTRE LA CALLE		Y LA CALLE	
DATOS GENERALES DE LOS HECHOS			
FECHA DE LOS HECHOS		MOMENTO DE OCURRENCIA DE LOS HECHOS	
01/06/04			
HORA DE LOS HECHOS			
12:58			

Reglas de Negocio:

Datos generales de la averiguación previa:

- Automáticamente se llevara el tipo de la averiguación previa a ingresar (directa, continuada, relacionada o robo de vehículo).
- El llenado de todos los formularios serán con mayúsculas.
- El formato del número de averiguación previa es el siguiente: agencia/turno/folio/año-mes bis.
- Se verificará que el número de averiguación previa no exista en el sistema.
- Se deberá seleccionar de manera obligatoria si es una averiguación con o sin detenido.
- El formato de fecha de inicio será dd/mm/aa y por default asignará la fecha del día corriente.
- El formato de hora de inicio será hh/mm y por default asignará la hora corriente.

Ventana para agregar delitos:

Documentos sin título - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

← Atrás → Búsqueda Favoritos Historial Ir a Vínculos

Dirección I:\enves\tesis\diplomadoenves\anexos\INTERFAZFINAL\adddelito.html

Personalizar Búsquedas ...attempting to retrieve buttons from Yahoo!...

DATOS GENERALES DE LOS DELITOS	
DELITO	MODALIDAD
ROBO A TRANSEUNTE	CON VIOLENCIA

DELITOS AGREGADOS	
DELITO	MODALIDAD
ROBO A CASA HABITACION	CON VIOLENCIA

Agregar Quitar Regresar

Listo Intranet local

Reglas de Negocio:

- Se deberá agregar obligatoriamente un delito con su modalidad.
- Solo se pueden eliminar delitos si hay más de uno.
- se Mostrará la información de los delitos agregados en el formulario principal.

Datos generales del lugar de los hechos:

- Toda la información requerida en el apartado de Datos generales del lugar de los hechos no serán requeridos obligatoriamente (Calle, número interior y exterior, entre la calle y la calle, colonia, delegación, y código postal), porque muchas veces los denunciantes no recuerdan en qué lugar fueron los hechos, llenándose dichos campos como nulos.

Datos Generales de los Hechos:

- El formato de fecha de los hechos será dd/mm/aa y por default asignará la fecha del día corriente.
- El formato de hora de los hechos será hh/mm y por default asignará la hora corriente.

Ventana para agregar denunciante:

The screenshot shows a Microsoft Internet Explorer browser window displaying a web form. The browser's address bar shows the URL: `L:\www\tes\aplonadadon\est\apexos\INTERFAZ\FINAL\adddenunciante.html`. The form is titled "DATOS GENERALES DE DENUNCIANTES" and contains the following fields:

DATOS GENERALES DE DENUNCIANTES			
NOMBRE(S)		APELLIDO PATERNO	
SEXO		EDAD	
C M F		IDENTIFICACION	
		CREDENCIAL DEL IFE	
DOMICILIO			
CALLE		NUM. EXT.	NUM. INT.
COLUMNA		C.P.	
ANZURES			
DELEGACION		TELEFONO	
MIGUEL HIDALGO			

At the bottom left of the form is a button labeled "Agregar".

Reglas de negocio:

- Se debe agregar obligatoriamente a un denunciante:
- Obligatoriamente se debe ingresar el nombre del denunciante.
- Obligatoriamente se deben ingresar los datos correspondientes al domicilio del denunciante a excepción del teléfono.
- Mostrará únicamente el nombre del denunciante en el formulario principal de captura en forma de lista en caso de tener más de un denunciante.

Ventana para agregar detenidos:

The screenshot shows a Microsoft Internet Explorer browser window displaying a web form. The browser's address bar shows the URL: I:\mvs\test\diplomado\mvs\anexos\INTERFAZ\FINAL\adddenunciante.html. The form is titled "DATOS GENERALES DE DENUNCIANTES" and contains the following fields:

DATOS GENERALES DE DENUNCIANTES			
NOMBRE(S)		APELLIDO PATERNO	APELLIDO MATERNO
SSNO		EDAD	IDENTIFICACION
C M C F			CREDENCIAL DEL IFE
DOMICILIO			
CALLE		NUM. EXT.	NUM. INT.
COLONIA		C.P.	
ANZURES			
DELEGACION		TELEFONO	
MIGUEL HIDALGO			

Below the form is a button labeled "Agregar". The browser's status bar at the bottom shows "Intranet local".

Reglas de Negocio:

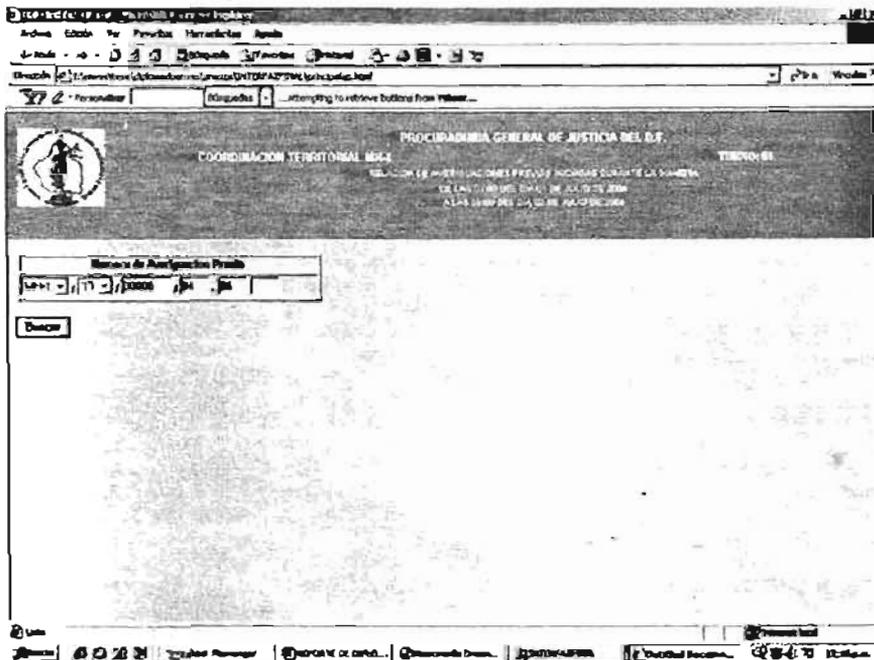
- Si se tiene seleccionado el campo S/D el sistema pondrá por default la leyenda "Q.R.R. (Quien Resulte Responsable)", en el apartado de detenidos
- Se debe agregar obligatoriamente a un detenido en el caso de tener seleccionado el campo C/D.
- Obligatoriamente se debe ingresar el nombre del detenido.
- Mostrará únicamente el nombre del detenido en el formulario principal de captura en forma de lista en caso de tener más de un denunciante.

Al momento de guardar la información del formulario principal:

- Se validarán todos los campos requeridos.

- En caso de no ser la información requerida o ser errónea, Enviará mensaje de verificación de datos.
- En caso de ser valida la información, enviará mensaje de confirmación de datos y guardará el registro regresando al área de trabajo para averiguaciones previas.
- Actualizará automáticamente la información de las averiguaciones previas en el área de trabajo.

Página para modificar la información de una averiguación previa:



Reglas de negocio:

- Deberá ingresar necesariamente el número a averiguación previa a modificar
- Si no se localiza el número de averiguación previa desplegará un mensaje informando que no existe el número.
- Si localiza la averiguación mostrará en pantalla el formulario con la información ingresada.

COORDINACIÓN TERRITORIAL III-4
PROCURADURÍA GENERAL DE JUSTICIA DEL D.F.
RELACION DE AVERIGUACIONES PREVIAS INICIADAS DURANTE LA SEMANA DE LAS 00:00 HRS. DEL 01 DE JUNIO DE 2004 AL 00:00 HRS. DEL 07 DE JUNIO DE 2004

DATOS GENERALES DE LA AVERIGUACIÓN PREVIA

NUMERO DE AVERIGUACION PREVIA	TIPO DE A.P.	FECHA DE INICIO	HORA DE INICIO
1471	C O F S B	01/06/04	12:50
DELITO		DELITOS AGREGADOS	
ROBO A CASA HABITACION SIN VIOLENCIA			
DATOS GENERALES DEL LUGAR DE LOS HECHOS			
CALLE	NUMERO EXTERIOR	NUMERO INTERIOR	
La piedad	12		
COLORS	DEL GACION	CP.	
ANZURES	MIGUEL HIDALGO	00000	
ENTRE LA CALLE	Y LA CALLE		
And	Medico		
DATOS GENERALES DE LOS HECHOS			
FECHA DE LOS HECHOS	BREVE DESCRIPCION DE LOS HECHOS		
01/06/04	Al regresar a su domicilio se percato de que		
HORA DE LOS HECHOS	de que		
12:50	las chapas estan forzadas		
DENUNCIANTES Y DE TENIDOS			

Reglas de negocio:

- Se actualizará automáticamente la información de las averiguaciones previas que se han modificado, agregado o borrado, en el área de trabajo para el tipo de la averiguación.

Datos generales de la averiguación previa:

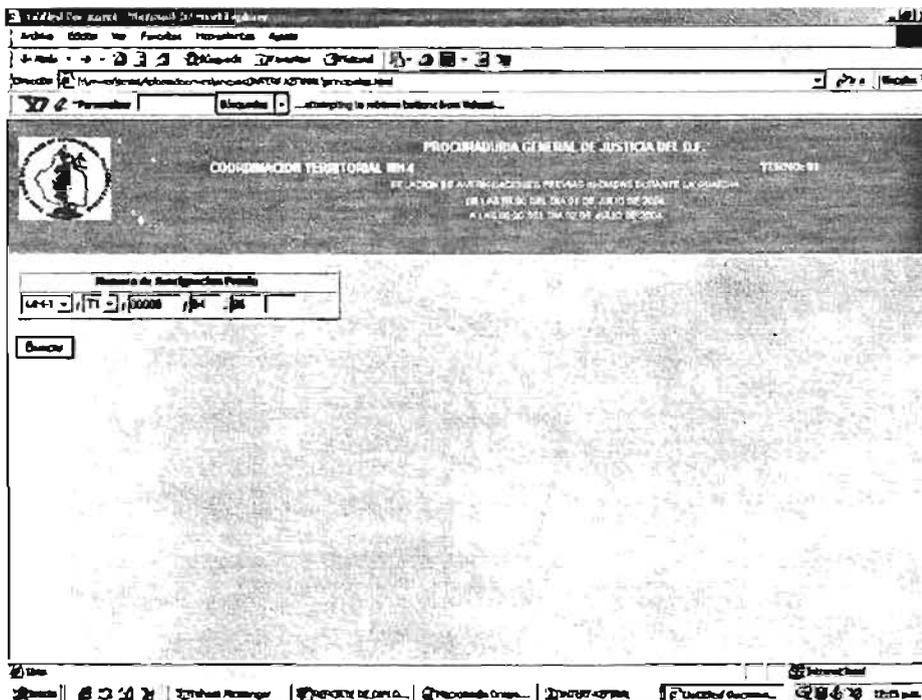
- Podrá modificar todos los datos de la averiguación previa, excepto el número de la averiguación.
- El llenado de todos los formularios serán con mayúsculas.
- Podrá agregar o quitar delitos con su modalidad.
- Sólo se pueden eliminar delitos si hay más de uno.
- Se mostrará la información de los delitos agregados o modificados en el formulario principal.

- Se podrán modificar los datos de los denunciantes y detenidos.
- Se podrá cambiar una averiguación previa "sin detenido" a "con detenido".
- Si se cambia el tipo de averiguación, obligatoriamente se debe ingresar el nombre del detenido.
- Mostrará únicamente el nombre del detenido en el formulario principal de captura en forma de lista en caso de tener más de un denunciante.

Al momento de guardar la información del formulario principal:

- Se validarán todos los campos requeridos.
- En caso de no ser la información requerida o ser errónea, Envió mensaje de verificación de datos.
- En caso de ser válida la información, enviará mensaje de confirmación de datos y guardará el registro regresando al área de trabajo para averiguaciones previas.
- Actualizará automáticamente la información de las averiguaciones previas en el área de trabajo.

Página para el borrado de averiguaciones Previas:



Reglas de negocio:

- Deberá ingresar necesariamente el número a averiguación previa a modificar
- Si no se localiza el número de averiguación previa, desplegará un mensaje informando que no existe el número.
- Si localiza la averiguación mostrará en pantalla el formulario con la información ingresada.

Reglas de negocio:

- Los botones del formulario y los datos estarán desactivados para modificaciones.
- Sólo borrará la averiguación previa si se da correctamente la clave de acceso.

Reglas de Negocio:

- Si la clave no es la correcta el sistema enviará mensaje de aviso y no borrará el registro del sistema.
- Si la clave es correcta mandará un mensaje de verificación para el borrado del registro.
- Borrara el registro y modificará la infomación del área de trabajo para las averiguaciones previas.

Por cuestiones de espacio, la interfaz de usuario y las reglas de negocio de los módulos faltantes se encuentran contenidos en el anexo V.

CAPITULO IV DESARROLLO E IMPLEMENTACIÓN.

IV.1 DESARROLLO

Para comenzar con el desarrollo de la aplicación se debe preparar primeramente, el equipo donde se desarrollara la aplicación, el cual es un equipo con las siguientes características:

- Pentium 4 a 1.6 Ghz.
- 256 MB de ram
- disco duro de 120 Gb. de almacenamiento

Para el desarrollo se instalará el siguiente software:

- sistema operativo Windows xp
- servidor apache como servidor web
- Php como lenguaje de desarrollo
- Mysql como manejador de base de datos
- Phpmyadmin como herramienta auxiliar para el manejo de la base de datos
- Dreamweaver como auxiliar para la creación de la interfaz de usuario.

La aplicación se desarrollará de manera modular; en la cual primeramente se crearán los catálogos, los cuales son la parte medular del sistema; en segundo termino los formularios de ingreso de datos del sistema y por último la generación de reportes diversos; en el presente trabajo sólo se explicará de forma breve el modo de cómo se creo el módulo de catálogos.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

IV.1.1 MÓDULO DE CATÁLOGOS

Para este módulo, es necesario primeramente analizar nuestro diseño de la base de datos, para definir qué tablas serán parte de nuestros catálogos y cuáles no pertenecerán a los mismos; es muy importante identificarlos porque ellos son la parte medular del sistema y de ellos se tomará la mayor parte de la información que se ingresará en el mismo.

Se han identificado tres tablas como catálogos que serán estáticos y con información que no cambiará, por lo que se ha decidido hacer la inserción de los datos por medio de un script y no se creará vista de los mismos en la aplicación, dichos catálogos serán:

- Entidad: Son los 31 estados de la república más el distrito federal, los cuales no cambiarán.
- Turno: sólo serán tres turnos, ya que con estos turnos se cubren los 365 días del año en horarios de 24 horas de trabajo por 48 horas de descanso.
- Tipo_ap: no hay más que cinco tipos de averiguación previa los cuales, no pueden ser cambiados.

En cuanto a las demás tablas se identificaron doce tablas que pertenecerán a los catálogos; los cuales, serán dinámicos por el manejo de información de las mismas; dichas tablas son:

- Delegación: aunque las 16 delegaciones del Distrito Federal no cambiarán, se optó por ponerlo dinámico para poder agregar delegaciones de las distintas entidades.
- Coordinación: en cualquier momento pueden ser creadas nuevas coordinaciones o en su caso desaparecer.
- Colonias: En cualquier momento se pueden crear nuevas colonias o cambiar de nombre.

- Marca_auto: pueden ser cambiantes las marcas de los vehículos de un año a otro.
- Submarca_auto: pueden desaparecer o salir nuevas submarcas de autos en cualquier momento.
- Color: cambian en cada momento.
- Remitente: puede crearse algún nuevo grupo policial para la prevención del delito en cualquier momento.
- Resolución_ap: no están definidos en su totalidad y en su caso pueden crearse nuevas resoluciones.
- Delito: no existe un catálogo definido de todos los delitos.
- Modalidad: la modalidad del delito es cambiante a cada momento.
- Usuarios: el personal es reubicado en cualquier momento.
- Perfiles: previene la creación de nuevos perfiles para nuevos usuario dentro del sistema.

Para el manejo de información de estos catálogos se utilizarán las librerías Adodb proporcionadas en el diplomado, las cuales se modificaron en su código original para ser adecuadas a cada uno de los catálogos.

La creación de estos los catálogos y la inserción de datos en los catálogos estáticos se hizo con el script que se muestra a continuación:

```
#Script para la creación de la base de datos y las tablas
```

```
#crea la base de datos PRUEBA
```

```
DROP DB `prueba`  
CREATE DB `prueba`
```

Estructura de tabla para la tabla `colonias`

```
CREATE TABLE `colonias` (  
  `id_colonia` int(2) NOT NULL auto_increment,  
  `entidad` int(2) NOT NULL default '0',  
  `delegacion` int(2) NOT NULL default '0',  
  `colonia` varchar(40) NOT NULL default '',  
  PRIMARY KEY (`id_colonia`)  
) TYPE=MyISAM AUTO_INCREMENT=4 ;
```

Estructura de tabla para la tabla `color`

```
CREATE TABLE `color` (  
  `id_color` smallint(2) NOT NULL auto_increment,  
  `color` varchar(30) NOT NULL default '',  
  PRIMARY KEY (`id_color`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

Estructura de tabla para la tabla `coordinacion`

```
CREATE TABLE `coordinacion` (  
  `id_coordinacio` smallint(2) NOT NULL auto_increment,  
  `delegacion` smallint(2) NOT NULL default '0',  
  `entidad` smallint(2) NOT NULL default '0',  
  `nom_coordinacion` varchar(50) NOT NULL default '',  
  `clave_coord` varchar(20) NOT NULL default '',  
  PRIMARY KEY (`id_coordinacio`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

Estructura de tabla para la tabla `delegacion`

```
CREATE TABLE `delegacion` (  
  `id_delegacion` int(2) NOT NULL auto_increment,  
  `id_entidad` int(2) NOT NULL default '0',  
  `nombre_del` varchar(40) NOT NULL default '',  
  PRIMARY KEY (`id_delegacion`)  
) TYPE=MyISAM AUTO_INCREMENT=23 ;
```

Estructura de tabla para la tabla `delito`

```
CREATE TABLE `delito` (  
  `id_delito` smallint(3) NOT NULL auto_increment,  
  `delito` varchar(50) NOT NULL default '',  
  PRIMARY KEY (`id_delito`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `entidad`
```

```
CREATE TABLE `entidad` (  
  `id_entidad` int(2) NOT NULL auto_increment,  
  `nom_entidad` varchar(40) NOT NULL default "",  
  PRIMARY KEY (`id_entidad`)  
) TYPE=MyISAM AUTO_INCREMENT=5 ;
```

```
# Estructura de tabla para la tabla `marca_auto`
```

```
CREATE TABLE `marca_auto` (  
  `id_marca` smallint(2) NOT NULL auto_increment,  
  `marca` varchar(30) NOT NULL default "",  
  PRIMARY KEY (`id_marca`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `modalidad`
```

```
CREATE TABLE `modalidad` (  
  `id_modalida` smallint(2) NOT NULL auto_increment,  
  `modalidad` varchar(50) NOT NULL default "",  
  PRIMARY KEY (`id_modalida`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `perfil`
```

```
CREATE TABLE `perfil` (  
  `id_perfil` smallint(2) NOT NULL auto_increment,  
  `perfil` smallint(2) NOT NULL default '0',  
  PRIMARY KEY (`id_perfil`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `remite`
```

```
CREATE TABLE `remite` (  
  `id_remite` smallint(2) NOT NULL auto_increment,  
  `remite` varchar(30) NOT NULL default "",  
  PRIMARY KEY (`id_remite`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `resolucion_ap`
```

```
CREATE TABLE `resolucion_ap` (  
  `id_resolucion` smallint(2) NOT NULL auto_increment,  
  `resolucion` varchar(40) NOT NULL default "",  
  PRIMARY KEY (`id_resolucion`)  
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `submarca_auto`

CREATE TABLE `submarca_auto` (
  `id_submarca` smallint(2) NOT NULL auto_increment,
  `marca` smallint(2) NOT NULL default '0',
  `submarca` varchar(30) NOT NULL default "",
  PRIMARY KEY (`id_submarca`)
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `tipo_ap`

CREATE TABLE `tipo_ap` (
  `id_tipo` smallint(2) NOT NULL auto_increment,
  `tipo_ap` varchar(40) NOT NULL default "",
  PRIMARY KEY (`id_tipo`)
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

```
# Estructura de tabla para la tabla `turnos`

CREATE TABLE `turnos` (
  `id_turno` smallint(1) NOT NULL auto_increment,
  `turno` varchar(10) NOT NULL default "",
  PRIMARY KEY (`id_turno`)
) TYPE=MyISAM AUTO_INCREMENT=1 ;
```

A continuación, y para ejemplificar la forma de crear dichos catálogos se tomará sólo uno, el de delegaciones y se mostrará el código para las operaciones básicas (altas, bajas, cambios y consultas) dentro de los catálogos.

Código para la página principal que muestra la lista de los registros en el catálogo de delegaciones.

```
# Uso de las librerías adodb

<?php
# include($DOCUMENT_ROOT."/seguridad/secure.php");

if(empty($DOCUMENT_ROOT))
  $DOCUMENT_ROOT=$HTTP_SERVER_VARS["DOCUMENT_ROOT"];
include_once($DOCUMENT_ROOT."/adodb/adodb.inc.php");
include_once($DOCUMENT_ROOT."/adodb/adodb-ABC.inc.php");
session_start();
?>
```

```

<h2>Delegaciones</h2>
<p><a href="http://informatica_02/catálogos/addDelegacion.php">Registrar</a>
<a href="http://informatica_02/catálogos/salida.php">Salir</a>

</p>
<p>

# Hacer la connexion a la base de datos

<?
$db = NewADOConnection('mysql');
$db->Connect('127.0.0.1','root','', 'prueba');
$sql = "select id_delegacion ID, nombre_del Delgacion from delegacion d,
        entidad e where d.id_entidad=e.id_entidad order by 2";

# cantidad de registros que Mostrará en cada página
$pager = new ADODB_ABC($db,$sql,"true','catálogos','delegacion');
$pager->Render($rows_per_page=8);
?>

```

Código para el alta de una delegación

```

<?php
if(empty($DOCUMENT_ROOT))
    $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];
include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$conn = &ADONewConnection('mysql');
$conn->PConnect('127.0.0.1','root','', 'prueba');
$sql = 'select nom_entidad, id_entidad from entidad';
$rs = $conn->Execute($sql);
$laentidad = $rs->GetMenu('id_entidad');
$conn->Close();
?>

<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link href="tirulo.css" rel="stylesheet" type="text/css">
</head>
<body>
<form
                name="form1"
                method="post"
action="http://informatica_02/catálogos/raddDelegacion.php">
<table width="84%" border="1">

```

```

<tr class="titulo1">
  <td colspan="2"><div align="center"><font size="3" face="Arial, Helvetica,
sans-serif"><strong>Registrar
  una Delegacion </strong></font></div></td>
</tr>
<tr class="titulo1">
  <td width="41%">Nombre de la Delegacion</td>
  <td width="59%"><input name="nombre_del" type="text" id="nombre_del"></td>
</tr>
<tr class="titulo1">
  <td>Entidad federativa</td>
  <td>
    <?php
      echo $laentidad;
    ?>
  </td>
</tr>
<tr bgcolor="#D4D0C8">
  <td colspan="2"><div align="center">
    <input name="Submit" type="submit" class="titulo1" value="Enviar">
  </div></td>
</tr>
</table>
</form>
</body>
</html>

```

#Operación de Inserción de un Registro de delegaciones

```

<?
if(empty($DOCUMENT_ROOT))
  $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];
include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$db = NewADOConnection('mysql');
$db->Connect('127.0.0.1','root','prueba');

$sql = "select * from delegacion where id_delegacion=-1";
$rs = $db->Execute($sql);

$UnaDelegacion = array();

$id_delgacion = $db->GenID('seq_delegacion');

$UnaDelegacion['id_delegacion']=$id_delegacion;
$UnaDelegacion['nombre_del']=$HTTP_POST_VARS['nombre_del'];
$UnaDelegacion['id_entidad']=$HTTP_POST_VARS['id_entidad'];

```

```

$sql_Inserta = $db->GetInsertSQL($rs,$UnaDelegacion);

#print "INSERTAR:[" . $sql_Inserta."]";
$db->Execute($sql_Inserta);
$db->Close();
header("Location: index.php");
?>

```

Código para los cambios de delegaciones.

```

<?php
if(empty($DOCUMENT_ROOT))
    $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];

include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$conn = &ADONewConnection('mysql');
$conn->PConnect('127.0.0.1','root','','prueba');

#Obtener el registro a modificar

$sql = "select nombre_del, nom_entidad from delegacion d,";
$sql.= "entidad e where id_delegacion=" . $HTTP_GET_VARS['id'];
$sql.= " and d.id_entidad=e.id_entidad";
$rs = $conn->Execute($sql);

if (!$rs) {
    print $conn->ErrorMsg();
    exit();
}
else {
    $NOMBRE_DEL = $rs->fields[0];
    $NOM_ENTIDAD= $rs->fields[1];
}

#Ahora las entidades

$sql = 'select nom_entidad, id_entidad from entidad';
$rs = $conn->Execute($sql);
$laentidad = $rs->GetMenu('id_entidad',$id_entidad,false);
$conn->Close();

?>
<html>
<head>

```

```
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link href="tirulo.css" rel="stylesheet" type="text/css">
</head>

<body>
<form name="form1" method="post"
action="http://informatica_02/catálogos/rupdDelegacion.php">
  <table width="75%" border="1">
    <tr bgcolor="#D4D0C8">
      <td colspan="2"><div align="center"><font size="3" face="Arial, Helvetica,
sans-serif" class="titulo1"><strong> Cambiar una
Delegación</strong></font></div></td>
    </tr>
    <tr>
      <td width="30%" class="titulo1">Nombre Delegación:</td>
      <td width="70%" bgcolor="#D4D0C8"><input name="nombre_del" type="text"
class="titulo1"
id="nombre_del" value="<?php echo $NOMBRE_DEL;?>"></td>
    </tr>
    <tr>
      <td class="titulo1">Entidad Federativa:</td>
      <td bgcolor="#D4D0C8">
        <?php
          echo $laentidad;
        ?>
      </td>
    </tr>
    <tr>
      <td colspan="2" class="titulo1"><div align="center">
        <input name="Submit" type="submit" class="titulo1" value="Enviar">
        <input type="hidden" name="id_delegacion" value="<?php echo
$HTTP_GET_VARS['id'];?>">
      </div></td>
    </tr>
  </table>
  <div align="center"></div>
</form>

</body>
</html>
```

Operación de Actualización de un Registro

```

<?
if(empty($DOCUMENT_ROOT))
    $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];
include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$db = NewADOConnection('mysql');
$db->Connect('127.0.0.1','root','','prueba');

$sql = "select * from delegacion where
id_delegacion=".$HTTP_POST_VARS['id_delegacion'];
$rs = $db->Execute($sql);

$UnaDelegacion = array();

$UnaDelegacion["nombre_del"]=$HTTP_POST_VARS['nombre_del'];
$UnaDelegacion["id_entidad"]=$HTTP_POST_VARS['id_entidad'];

$sql_Actualiza = $db->GetUpdateSQL($rs,$UnaDelegacion);

#print "Actualizar:[" . $sql_Actualiza."]";
$db->Execute($sql_Actualiza);
$db->Close();

header("Location: index.php");
?>

```

Código para la consulta de delegaciones.

```

<?php
if(empty($DOCUMENT_ROOT))
    $DOCUMENT_ROOT = $HTTP_SERVER_VARS['DOCUMENT_ROOT'];
include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$conn = &ADONewConnection('mysql');
$conn->PConnect('127.0.0.1','root','','prueba');

$sql = "select nombre_del, nom_entidad from delegacion d,";
$sql.= "entidad e where id_delegacion=" . $HTTP_GET_VARS['id'];
$sql.= " and d.id_entidad=e.id_entidad";

$rs = $conn->Execute($sql);

if (!$rs) {
    print $conn->ErrorMsg();
}

```

```
        exit();
    }
    else {
        # $ID_DELEGACION = $rs->fields[0];
        $NOMBRE_DEL = $rs->fields[0];
        $NOM_ENTIDAD= $rs->fields[1];

    }
    $conn->Close();
?>
<html>
<head>
<title><?php echo $NOMBRE_DEL?></title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link href="tirulo.css" rel="stylesheet" type="text/css">
</head>

<body>
<table width="75%" border="1">
<tr bgcolor="#D4D0C8">
<td colspan="2"><div align="center"><font size="3" face="Arial, Helvetica,
sans-serif" class="titulo1"><strong>Consultar
una Delegacion</strong></font></div></td>
</tr>
<tr>
<td width="29%" class="titulo1">Delegación</td>
<td width="71%" class="titulo1"><?php echo $NOMBRE_DEL; ?></td>
</tr>
<tr>
<td class="titulo1">Entidad Federativa</td>
<td class="titulo1"><?php echo $NOM_ENTIDAD; ?></td>
</tr>
<tr bgcolor="#D4D0C8">
<td colspan="2"><div align="center"><a
href='http://informatica_02/catálogos/index.php' class="titulo1">Regresar</a>
</div></td>
</tr>
</table>
</form>

</body>
</html>
```

Código para el borrado de un registro de delegaciones.

```
<?php
  if(empty($DOCUMENT_ROOT))
    $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];
  include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
  $conn = &ADONewConnection('mysql');
  $conn->PConnect('127.0.0.1','root','', 'prueba');

  #Obtener el registro a eliminar

  $sql = "select nombre_del, nom_entidad from delegacion d,";
  $sql.= "entidad e where id_delegacion=" . $HTTP_GET_VARS['id'];
  $sql.= " and d.id_entidad=e.id_entidad";

  $rs = $conn->Execute($sql);

  if (!$rs) {
    print $conn->ErrorMsg();
    exit();
  }
  else {
    $NOMBRE_DEL = $rs->fields[0];
    $NOM_ENTIDAD= $rs->fields[1];
  }

  #Ahora las entidades

  $sql = 'select nom_entidad, id_entidad from entidad';
  $rs = $conn->Execute($sql);
  $lasCalif = $rs->GetMenu('nom_entidad',$nom_entidad,false);
  $conn->Close();

?>
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<link href="tirulo.css" rel="stylesheet" type="text/css">
</head>

<body>
<form name="form1" method="post"
action="http://informatica_02/catálogos/rdelDelegacion.php">
```

```

<table width="75%" border="1">
  <tr>
    <td colspan="2"><div align="center" class="titulo1"><font size="3" face="Arial,
Helvetica, sans-serif"><strong>Consultar
una Delegacion</strong></font></div></td>
  </tr>
  <tr>
    <td width="28%" class="titulo1">Delegación</td>
    <td width="72%" class="titulo1"><?php echo $NOMBRE_DEL; ?></td>
  </tr>
  <tr>
    <td class="titulo1">Entidad Federativa</td>
    <td class="titulo1"><?php echo $NOM_ENTIDAD; ?></td>
  </tr>
  <tr bgcolor="#F0F0F0">
    <td colspan="2" class="titulo1">
      <div align="center">
        <input name="Submit" type="submit" class="titulo1" value="Confirmar
borrado">
        <input type="hidden" name="id_delegacion" value="<?php echo
$http_get_vars[id];?>">
      </div></td>
  </tr>
</table>
</form>

</body>
</html>

```

Operación de borrado de un Registro

```

<?
if(empty($DOCUMENT_ROOT))
  $DOCUMENT_ROOT=$HTTP_SERVER_VARS['DOCUMENT_ROOT'];
include($DOCUMENT_ROOT."/adodb/adodb.inc.php");
$db = NewADoConnection('mysql');
$db->Connect('127.0.0.1','root','','prueba');

$sql = "delete from delegacion where
id_delegacion=".$HTTP_POST_VARS[id_delegacion];
$rs = $db->Execute($sql);

$db->Close();
header("Location: index.php");
?>

```

Una vez hechas las funciones anteriormente explicadas, se anexarán a la estructura de las páginas de trabajo para el módulo de catálogos, las cuales se muestran el capítulo 3, en la interfaz de usuario.

IV.2 IMPLEMENTACIÓN.

La Fiscalía de Miguel Hidalgo, cuenta con cinco coordinaciones territoriales, de las cuales, sólo dos de ellas se encuentran físicamente en el edificio sede y las restantes distribuidas en el área geográfica correspondiente a la delegación; tomando en cuenta que se trabaja las 24 horas del día los 365 días del año, se debe diseñar un programa de implementación que no afecte la productividad de la fiscalía.

En una primera etapa sólo se implementará el sistema en la sede de la fiscalía, porque en ella se encuentra personal de informática las 24 horas para brindar apoyo a los usuarios y resolver cualquier problemática presentada.

Para la preparación del servidor solo se podrán utilizar 2 horas diarias en un horario de 15:00 a 17:00, puesto que es el lapso de tiempo que el personal toma sus alimentos y la carga de trabajo es baja; se debe tomar en cuenta además que se pueden presentar casos relevantes que impedirán el uso del servidor para la preparación.

El programa de preparación e instalación del sistema en su primera etapa para el servidor se describe a continuación:

Acciones	Noviembre			
	1 al 7	8 al 14	15 al 21	21 la 28
Diagnóstico del estado actual del servidor				
Preparación para la instalación de Linux				
instalación de linux con el módulo de PHP y apache				
Creación de la Base de Datos				
Pruebas de comunicación con la base de datos				
Capacitación sobre la administración del sistema al personal de informática				
Ingreso de los catálogos al sistema				

Tomando en cuenta que los clientes no necesitan un gran cambio en su sistema operativo y que todos los equipos cuentan con una suite de office 2000, la cual proporciona la herramienta de navegación Internet Explorer en su versión 5., sólo se revisarán, depurarán y en su caso se instalará la herramienta de navegación; será una tarea que se puede realizar en conjunto con la preparación del servidor.

El programa de preparación de los clientes se presenta, a continuación:

Acciones	Noviembre			
	1 al 7	8 al 14	15 al 21	21 la 28
Diagnóstico del estado actual de los clientes				
Preparación de los clientes para instalación de las herramientas necesarias para el uso del sistema.				
Pruebas de conectividad y funcionalidad de la red				
Pruebas de comunicación con la base de datos				
Capacitación sobre la administración del sistema al personal de informática				

Se usará el sistema como prueba piloto por un período de un mes, llevando el control estadístico en la forma anterior y se comparará con el control estadístico del nuevo sistema para localizar y modificar los errores que se desprendan de la prueba piloto; se tendrá un período de dos meses para corrección y adecuación del sistema, en caso de ser necesarios.

En su etapa final se implantará el sistema en las cinco coordinaciones para poder hincar el uso del sistema el 1 de febrero de 2006, esto es debido al control de numeración de las averiguaciones previas y a un acuerdo con el C. Fiscal de la delegación.

El programa para la implementación del sistema en las coordinaciones restantes, se muestra a continuación, tomando en cuenta que se tiene el personal de informática capacitado para este fin:

Acciones	Enero			
	1 al 7	8 al 14	15 al 21	26 la 30
Diagnóstico del estado actual de los servidores				
Preparación para la instalación de Linux				
instalación de linux con el módulo de PHP y apache				
Creación de la Base de Datos				
Pruebas de comunicación con la base de datos				
Carga de la información de los catálogos				
Diagnostico del estado actual de los clientes				
Preparación de los clientes para instalación de las herramientas necesarias para el uso del sistema.				
Pruebas de conectividad y funcionalidad de la red				
Pruebas de comunicación con la base de datos				
Capacitación sobre la administración del sistema al personal de informática				

CONCLUSIONES.

Las experiencias aportadas por el Diplomado de Desarrollo de Sistemas Web me permitieron tener una nueva herramienta para el desarrollo de sistemas; es un buen comienzo para la comunidad universitaria la apertura de estos diplomados, los cuales, aportarán herramientas y metodologías modernas para el mejor desempeño profesional en el ámbito laboral.

Crear sistemas basados en arquitecturas de tres capas con tecnología web, nos brinda grandes ventajas sobre los sistemas tradicionales, porque se pueden hacer cambios, adecuaciones y anexos a los sistemas sin que esto represente grandes costos y en un menor tiempo.

Los conocimientos adquiridos, me permitieron analizar el sistema de roles y hacer una propuesta profesional de un nuevo sistema, buscando explotar al máximo los recursos que con anterioridad no se utilizaban, anexando grandes beneficios para la institución, porque hasta la fecha no hay un sistema estadístico en el que se puedan generar reportes al momento con información real y verídica; destacando que si el sistema es institucionalizado, será el primer sistema con arquitectura web en la institución, en el cual estarán conectadas todas las Fiscalías y áreas centrales involucradas con el control estadístico de la incidencia delictiva diaria en el Distrito Federal.

En un futuro, este sistema se puede adecuar para una gran gama de diversos proyectos, entre los cuales de manera personal podría destacar los siguientes:

- **Control de Gestión de las Averiguaciones Previas.** Un sistema en que se registren los estados de las averiguaciones previas, con el cual las secretarías o el público en general podrá consultar en que estado se encuentra su averiguación previa y quien la esta trabajando.

- **Sistema Geográfico de Incidencia Delictiva.** Un sistema aunado a un GIS (Sistema de Información Geográfica), en el cual se puedan marcar los puntos de mayor incidencia delictiva, para poder implementar diversos operativos para combatir la delincuencia.

BIBLIOGRAFÍA:

El lenguaje unificado de modelado.

Addison Wesley.

Grady Booch, James Rumbaugh, Ivar Jacobson.

Building Web Applications With UML.

Adisson Wesley.

Jim Connallen.

Construcción de software orientado a objetos.

Prentice may.

Bertrand Meyer.

Tech TourSelf UML in 24 Hours.

Sams.

Joseph Schmulerr.

Web Application Development with PHP 4.0.

New Riders Publishing.

Tobias Ratschiller & Till Gerken.

Web Database Applications with PHP & MySQL.

O'Rilley, 2002.

Hugh E. Williams & David Lane.

Linux 6ª Edición.

Prentice Hall, 2001.

David Bandel & Rober Napier.

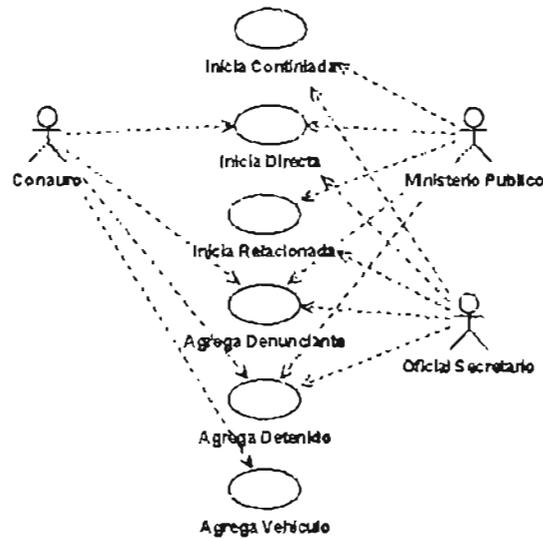
MySQL/PHP Database Applications.

M&T Books, 2001.

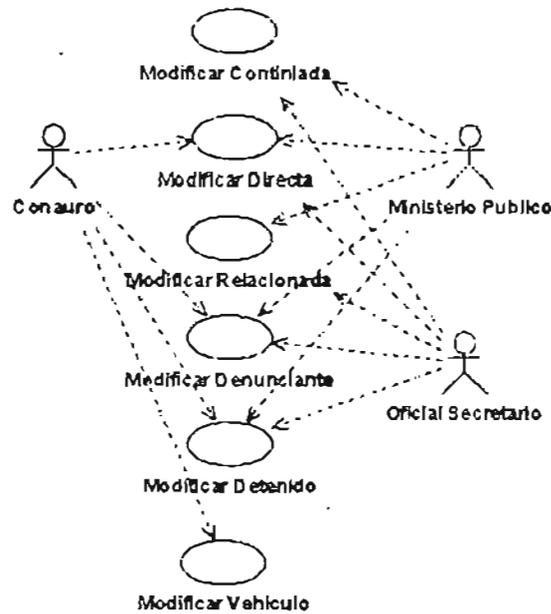
Jay Greenspan & Brad Bulger.

ANEXO I CASOS DE USO

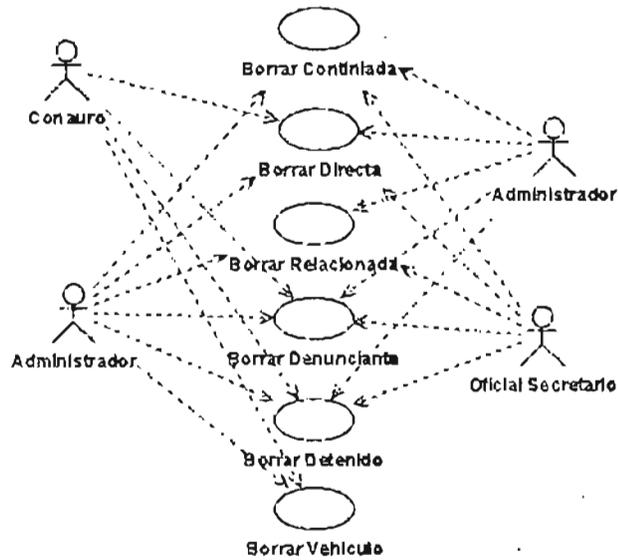
MÓDULO DE INICIAR AVERIGUACIÓN PREVIA



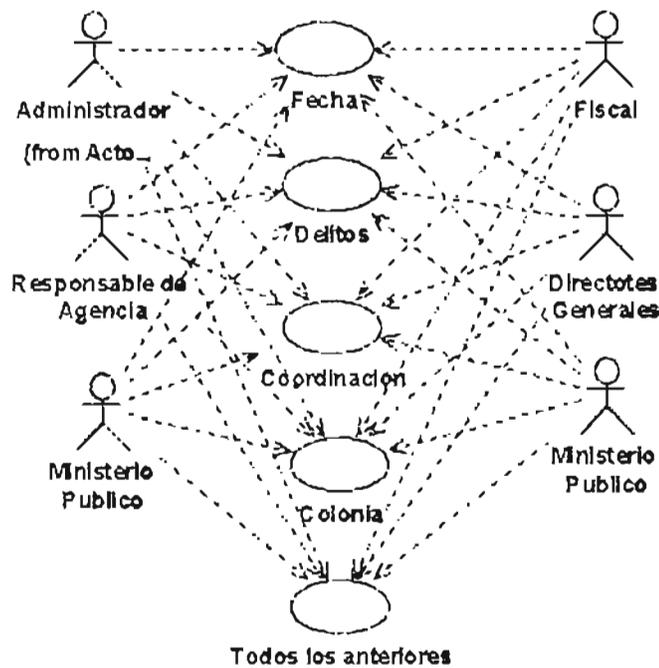
MÓDULO DE MODIFICAR AVERIGUACIÓN PREVIA



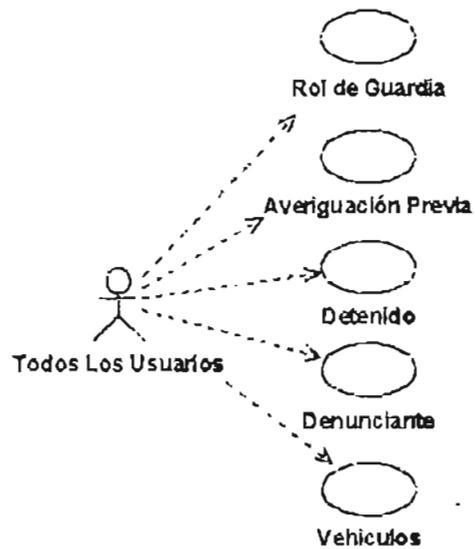
MÓDULO DE BORRAR AVERIGUACIÓN PREVIA



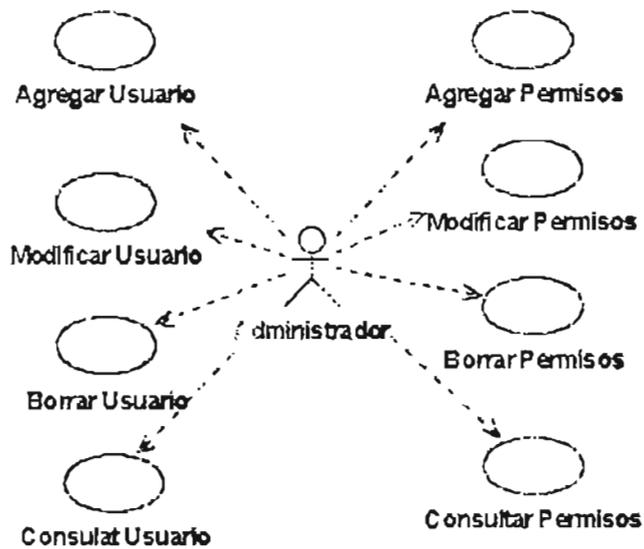
MÓDULO DE REPORTES



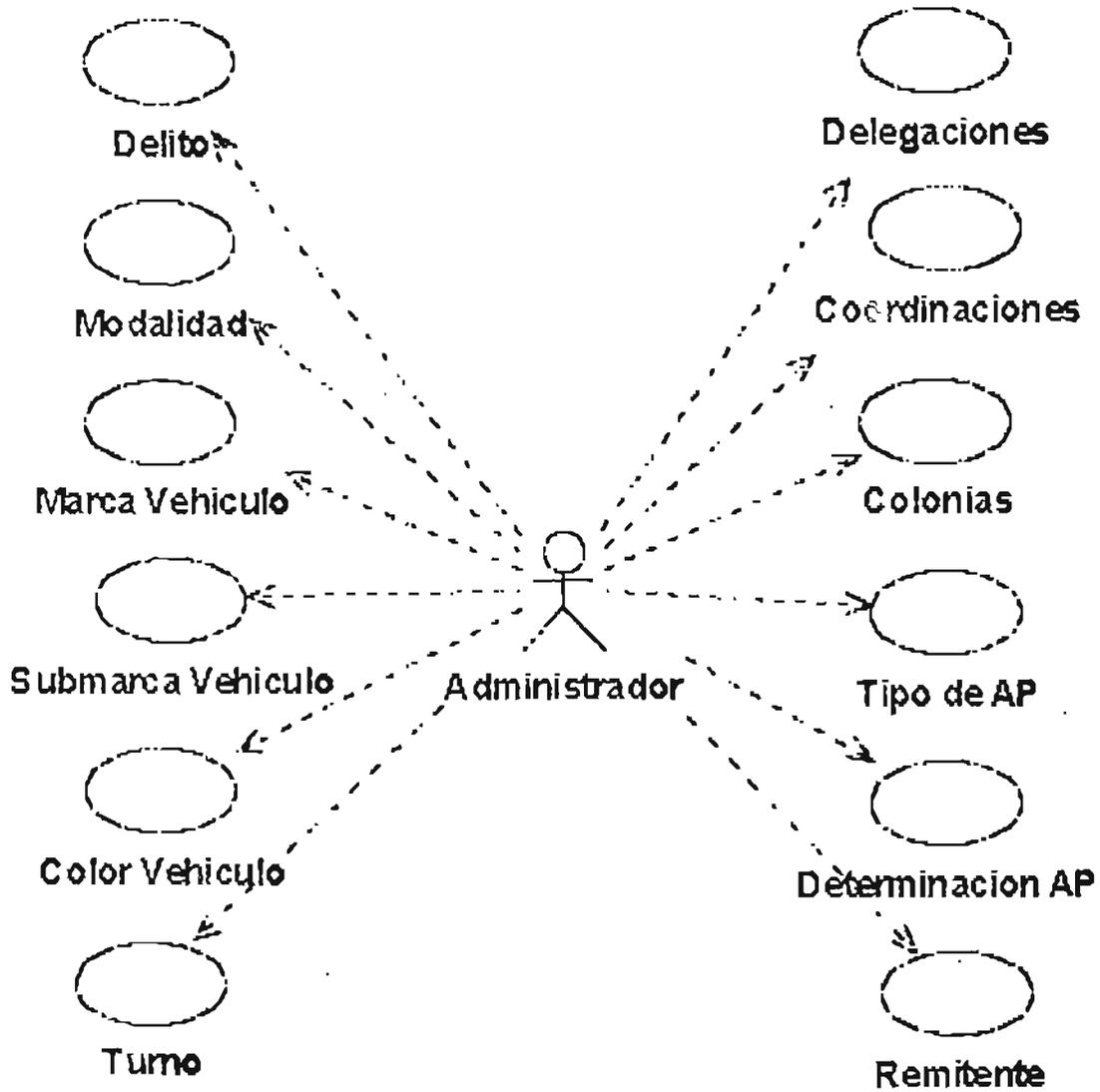
MÓDULO DE CONSULTAS



MÓDULO DE PERFILES



MÓDULO DE CATÁLOGOS



ANEXO II CONTRATOS DEL SISTEMA

Contratos para el caso de uso: **Iniciar Averiguación Previa**

Contrato: <code>Agrega_datos_ap()</code>	
Nombre	<code>Agregar_datos_de_ap</code>
Responsabilidades	Agregar un registro de averiguación previa al sistema.
Tipo	Sistema.
Referencias cruzadas	R4
Notas	Debe de haber un delito y al menos un denunciante para iniciar una averiguación previa.
Excepciones	Si el número de averiguación previa existe Envió mensaje de aviso.
Salida	Mostrará la información ingresada en el área de trabajo del usuario.
Pre-condiciones	El número de la averiguación previa no debe de existir registrado en el sistema.
Post-condiciones	Agregará un registro a la base de datos en la tabla <code>aver_previa</code> , debiendo previamente tener asignado al menos a un denunciante y un detenido; al guardar el registro y regresará al área de trabajo.

Contrato: <code>Agrega_datos_denunciante_ap()</code>	
Nombre	<code>Agregar_datos_denunciante_ap</code>
Responsabilidades	Agregar un denunciante a una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R5
Notas	Puede existir más de un denunciante por cada averiguación previa y se debe contar al menos con nombre completo y dirección.
Excepciones	Envió mensaje si los datos llenados son insuficientes
Salida	Mostrará la lista de los denunciados asignados a la averiguación previa.
Pre-condiciones	Debe de existir el número de una averiguación previa.
Post-condiciones	Asignará un denunciante a una averiguación previa y regresará al formulario principal de la misma.

Contrato: <code>Agrega_datos_detenido_ap()</code>	
Nombre	<code>Agregar_datos_detenido_ap</code>
Responsabilidades	Agregar un detenido a una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R6
Notas	Puede existir más de un detenido por cada averiguación previa.
Excepciones	Cuando la averiguación previa sea sin detenido el sistema agregará la leyenda Q.R.R.
Salida	Mostrará la lista de detenidos asignados a una averiguación o en su caso la leyenda Q.R.R.
Pre-condiciones	Debe existir el número de una averiguación previa.
Post-condiciones	Asignará un detenido a una averiguación previa y regresará al formulario principal de la misma.

Contrato: <code>Agrega_datos_vehículo_ap()</code>	
Nombre	<code>Agregar_datos_vehiculo_ap</code>
Responsabilidades	Agregar un vehículo a una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R7
Notas	Pueden existir más de un vehículo por cada averiguación previa y sólo podrán ser agregados por el usuario de conauro.
Excepciones	
Salida	Mostrará la lista de los autos asignados a la averiguación previa.
Pre-condiciones	Debe existir el número de una averiguación previa por el delito de robo de vehículo o repartidor.
Post-condiciones	Asignará un vehículo a una averiguación previa y regresará al formulario principal de la misma.

Contratos para el caso de uso: Modificar Averiguación Previa

Contrato: <code>Modifica_datos_ap()</code>	
Nombre	<code>Modifica_datos_de_ap</code>
Responsabilidades	Modificar un registro de averiguación previa en el sistema.
Tipo	Sistema.
Referencias cruzadas	R8
Notas	Debe de existir la averiguación previa a modificar.
Excepciones	Enviará mensaje de Averiguación Previa no localizada si no existe la misma en el sistema.
Salida	Modificará los datos en el área de trabajo.
Pre-condiciones	Debe de existir la averiguación previa en el sistema.
Post-condiciones	Modificará el registro de la averiguación previa y los datos generales de la misma, regresando al área de trabajo.

Contrato: Modifica_datos_denunciante_ap()	
Nombre	Modifica_datos_denunciante_ap
Responsabilidades	Modificar los datos de un denunciante en una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R9
Notas	Se debe de contar al menos con el nombre completo y la dirección del denunciante.
Excepciones	Enviará mensaje si los datos son insuficientes.
Salida	Modificará los datos del denunciante tanto en el formulario principal como en el área de trabajo.
Pre-condiciones	Debe de existir la averiguación previa en el sistema.
Post-condiciones	Modificará los datos de un denunciante de una averiguación previa y regresa al formulario principal de la misma.

Contrato: Modifica_datos_detenido_ap()	
Nombre	Modifica_datos_detenido_ap
Responsabilidades	Modificar los datos de un detenido en una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R10
Notas	
Excepciones	Cuando la averiguación previa sea sin detenido no se podrá modificar la leyenda Q.R.R.
Salida	Modificará los datos del detenido tanto en el formulario principal como en el área de trabajo.
Pre-condiciones	Debe existir la averiguación previa y no ser sin detenido.
Post-condiciones	Modificará los datos de un detenido en una averiguación previa y regresará al formulario principal.

Contrato: Modifica_datos_vehículo_ap()	
Nombre	Modifica_datos_vehiculo_ap
Responsabilidades	Modificar los datos de un vehículo en una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R11
Notas	Pueden existir más de un vehículo por cada averiguación previa y sólo podrán ser modificados por el usuario de conauro.
Excepciones	
Salida	Modifica los datos del vehículo tanto en el formulario principal como en el área de trabajo.
Pre-condiciones	Debe existir la averiguación previa y ser de robo de vehículo o repartidor.
Post-condiciones	Modificará los datos de un vehículo en una averiguación previa y regresa al formulario principal.

Contratos para el caso de uso: **Borrar Averiguación Previa**

Contrato: Borra_datos_ap()	
Nombre	Borra_datos_de_ap
Responsabilidades	Elimina un registro de averiguación previa al sistema.
Tipo	Sistema.
Referencias cruzadas	R12
Notas	Se deben de eliminar primeramente los denunciante y los detenidos.
Excepciones	No Eliminará la averiguación previa si la contraseña de seguridad es incorrecta.
Salida	Elimina el registro de la base de datos y del área de trabajo
Pre-condiciones	El número de la averiguación debe de existir en el sistema y la contraseña de seguridad solicitada debe ser correcta.
Post-condiciones	Eliminará un registro de averiguación previa de la base de datos y modifica la información del área de trabajo.

Contrato: Borra_datos_denunciante_ap()	
Nombre	Borra_datos_denunciante_ap
Responsabilidades	Eliminar un denunciante de una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R13
Notas	Puede existir más de un denunciante por cada averiguación previa.
Excepciones	Solo se Eliminarán todos los denunciante si se ha dado la instrucción de eliminar la averiguación previa.
Salida	Eliminará los datos del denunciante de las tablas denunciante_ap y denunciante
Pre-condiciones	Debe de existir la averiguación previa y tener asignados uno o más denunciante.
Post-condiciones	Eliminará los denunciante asignados a una averiguación previa.

Contrato: Borra_datos_detenido_ap()	
Nombre	Borra_datos_detenido_ap
Responsabilidades	Eliminar un detenido asignado a una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R14
Notas	Puede existir más de un detenido por cada averiguación previa.
Excepciones	Solo se Eliminarán todos los detenidos si se ha dado la instrucción de eliminar la averiguación previa.
Salida	Eliminará los datos del detenido de las tablas detenido_ap y detenido.
Pre-condiciones	Debe existir el número de la averiguación previa y tener asignado a un detenido.
Post-condiciones	Eliminará los detenidos asignados a una averiguación previa.

Contrato: Borra_datos_vehículo_ap()	
Nombre	Borra_datos_vehículo_ap
Responsabilidades	Borra un vehículo asignado a una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R15
Notas	Puede existir más de un vehículo por cada averiguación previa.
Excepciones	Solo se Eliminarán todos los vehículos si se ha dado la instrucción de eliminar la averiguación previa.
Salida	Eliminará los datos del vehículo de las tablas vehículo_ap y vehículos.
Pre-condiciones	Debe existir la averiguación previa por el delito de robo de vehículo o repartidor y tener asignado un vehículo.
Post-condiciones	Eliminará los vehículos asignados una averiguación previa.

Contratos para el caso de uso: Generar Reportes

Contrato: Genera_reporte ()	
Nombre	Genera_reporte
Responsabilidades	Generar Reportes por medio de los criterios predefinidos.
Tipo	Sistema.
Referencias cruzadas	R16
Notas	Debe de seleccionarse al meno un criterio de búsqueda para el reporte.
Excepciones	Enviará mensaje si son erróneos los datos para generar reporte.
Salida	Mostrará en pantalla un formato con la información seleccionada.
Pre-condiciones	Debe de tener acceso al módulo y seleccionar al menos uno de los criterios de búsqueda.
Post-condiciones	Mostrará la información del reporte con formato en pantalla dando la opción de imprimir.

Contratos para el caso de uso: Consultas

Contrato: Consulta_ap ()	
Nombre	Consulta_ap
Responsabilidades	Hacer una consulta de una averiguación previa.
Tipo	Sistema.
Referencias cruzadas	R17
Notas	Debe dar el número de la averiguación previa deseada para consultar.
Excepciones	Enviará mensaje si no se encuentra la averiguación previa deseada.
Salida	Mostrará en pantalla la averiguación previa localizada.
Pre-condiciones	Debe de tener acceso al módulo y proporcionar el número de la averiguación previa deseada.
Post-condiciones	Mostrará la información de la averiguación previa en pantalla con la opción de imprimir.

Contrato: Consulta_denunciante ()	
Nombre	Consulta_denunciante
Responsabilidades	Hacer una consulta de un denunciante.
Tipo	Sistema.
Referencias cruzadas	R18
Notas	Debe proporcionar al menos el nombre del denunciante a consultar.
Excepciones	Enviaré mensaje si no se encuentra el denunciante.
Salida	Mostraré en pantalla los denunciados encontrados en el sistema.
Pre-condiciones	Debe tener acceso al módulo y proporcionar al menos el nombre del denunciante deseado.
Post-condiciones	Mostraré la información de las averiguaciones previas localizadas con el nombre del denunciante en pantalla con la opción de imprimir.

Contrato: Consulta_detenido ()	
Nombre	Consulta_detenido
Responsabilidades	Hacer una consulta de un detenido.
Tipo	Sistema.
Referencias cruzadas	R19
Notas	Debe proporcionar al menos el nombre del detenido a consultar.
Excepciones	Enviaré mensaje si no se encuentra el detenido.
Salida	Mostraré en pantalla los detenidos encontrados en el sistema.
Pre-condiciones	Debe tener acceso al módulo y proporcionar al menos el nombre del detenido deseado.
Post-condiciones	Mostraré la información de las averiguaciones previas localizadas con el nombre del detenido en pantalla con la opción de imprimir.

Contrato: Consulta_rol_guardia ()	
Nombre	Consulta_rol
Responsabilidades	Hace una consulta del rol de guardia.
Tipo	Sistema.
Referencias cruzadas	R20
Notas	Debe proporcionar la fecha del rol de guardia a consultar.
Excepciones	Enviaré mensaje si no se encuentra el rol de guardia.
Salida	Mostraré en pantalla el rol de guardia de la fecha deseada.
Pre-condiciones	Debe tener acceso al módulo y proporcionar la fecha del rol de guardia deseado.
Post-condiciones	Mostraré la información del rol de guardia en pantalla con la opción de imprimir.

Contratos para el caso de uso: Perfiles

Contrato: Agregar_perfil ()	
Nombre	agrega_perfil
Responsabilidades	Agregar a un usuario al sistema y darle su perfil.
Tipo	Sistema.
Referencias cruzadas	R21
Notas	Debe ser un empleado de la institución.
Excepciones	Enviaré mensaje si los datos requeridos son incorrectos.
Salida	Mostraré en pantalla los datos de los usuarios registrados en el sistema.
Pre-condiciones	Debe ser un empleado de la institución y contar con su número de empleado.
Post-condiciones	Ingresara un registro a las tablas usuario y perfil_usuario, Mostraré en pantalla la información del usuario registrado.

Contrato: Mofidicar_perfil ()	
Nombre	Modifica_perfil
Responsabilidades	Hacer una modificación a los datos del usuario o a su perfil.
Tipo	Sistema.
Referencias cruzadas	R22
Notas	Debe proporcionar al menos el nombre del usuario o su número de empleado.
Excepciones	Enviaré mensaje si no se encuentra al usuario deseado.
Salida	Mostraré en pantalla los datos modificados del usuario.
Pre-condiciones	Debe ser un usuario registrado en el sistema.
Post-condiciones	Mostraré la información modificada en la pantalla.

Contrato: Borrar_perfil ()	
Nombre	Borra_perfil
Responsabilidades	Elimina a un usuario y su perfil del sistema.
Tipo	Sistema.
Referencias cruzadas	R23
Notas	Debe ser usuario registrado en el sistema.
Excepciones	No borrara al usuario si la contraseña de seguridad es errónea.
Salida	Eliminaré de la pantalla los datos del usuario.
Pre-condiciones	Debe ser un usuario registrado en sistema y la contraseña de seguridad correcta.
Post-condiciones	Borrara un registro a las tablas usuario y perfil_usuario, eliminando de la pantalla la información del usuario eliminado.

Contrato: Consultar_perfil ()	
Nombre	Consulta_perfil
Responsabilidades	Hace una consulta de un usuario registrado.
Tipo	Sistema.
Referencias cruzadas	R24
Notas	Debe proporcionar al menos el nombre del usuario o su número de empleado.
Excepciones	Enviará mensaje si no se encuentra al usuario deseado.
Salida	Mostrará en pantalla los datos del usuario deseado.
Pre-condiciones	Debe ser un usuario registrado en el sistema.
Post-condiciones	Mostrará la información completa del usuario deseado en la pantalla con la opción de imprimir.

Contratos para el caso de uso: Catálogos

Contrato: Agregar_en_catálogos ()	
Nombre	Agrega_en_catálogo
Responsabilidades	Agregar un registro en los catálogos del sistema.
Tipo	Sistema.
Referencias cruzadas	R25
Notas	La información necesaria dependerá del catálogo seleccionado.
Excepciones	Enviará mensaje cuando la información no sea suficiente para el registro en catálogos.
Salida	Mostrará la información ingresada en catálogos.
Pre-condiciones	Tener acceso al módulo, contar con la información requerida por el catálogo en cuestión.
Post-condiciones	Ingresara un registro a las tablas del catálogo en cuestión y Mostrará en pantalla la información del registro.

Contrato: Mofdicar_en_catálogos ()	
Nombre	Modifica_en_catálogo
Responsabilidades	Modifica un registro en los catálogos del sistema.
Tipo	Sistema.
Referencias cruzadas	R26
Notas	La información necesaria dependerá del catálogo seleccionado.
Excepciones	Enviará mensaje cuando no se localice el registro deseado en catálogos.
Salida	Mostrará la información modificada en catálogos.
Pre-condiciones	El registro en cuestión exista en catálogo.
Post-condiciones	Modificará un registro en las tablas del catálogo en cuestión y Mostrará en pantalla la información del registro modificado.

Contrato: Borrar_en_catálogos ()	
Nombre	Borra_en_catálogo
Responsabilidades	Elimina un registro en los catálogos del sistema.
Tipo	Sistema.
Referencias cruzadas	R27
Notas	La información necesaria dependerá del catálogo seleccionado.
Excepciones	Enviará mensaje cuando no se localice el registro deseado en catálogos y la contraseña de seguridad sea incorrecta.
Salida	Eliminará la información de la pantalla en catálogos.
Pre-condiciones	El registro en cuestión exista en catálogo y la clave de seguridad sea la correcta.
Post-condiciones	Eliminará un registro en las tablas del catálogo en cuestión y Eliminará de la pantalla la información del registro.

Contrato: Consultar_en_catálogos ()	
Nombre	Consulta_en_catálogo
Responsabilidades	Consultar un registro en los catálogos del sistema.
Tipo	Sistema.
Referencias cruzadas	R28
Notas	La información necesaria dependerá del catálogo seleccionado.
Excepciones	Enviará mensaje cuando no se localice el registro deseado en catálogos.
Salida	Mostrará en pantalla la información completa del registro localizado en catálogos.
Pre-condiciones	El registro en cuestión exista en catálogo y sea localizado
Post-condiciones	Mostrará en la pantalla la información completa del registro localizado en catálogos con la opción de imprimir.

Contratos para el caso de uso: Imprimir

Contrato: Imprime_Reporte_Generado ()	
Nombre	Imprime_reporte
Responsabilidades	Imprimir un reporte generado dentro del sistema.
Tipo	Sistema.
Referencias cruzadas	R29
Notas	Se debió generar previamente un reporte.
Excepciones	Cuando el usuario seleccione la opción de salir.
Salida	Enviará mensaje de envió de información a la impresora.
Pre-condiciones	Tener acceso al módulo, contar con un reporte generado y seleccionar la opción de imprimir.
Post-condiciones	Impresora en línea y configurada.

Contrato: Imprime_rol_de_guardia ()	
Nombre	Imprime_rol
Responsabilidades	Imprimir rol de guardia consultado.
Tipo	Sistema.
Referencias cruzadas	R30
Notas	Se debió consultar previamente el rol de guardia.
Excepciones	Cuando el usuario seleccione la opción de salir.
Salida	Enviará mensaje de envió de información a la impresora.
Pre-condiciones	Tener acceso al módulo, tener en pantalla el rol de guardia consultado y seleccionar la opción de imprimir.
Post-condiciones	Impresora en línea y configurada.

Contrato: Imprimir_consulta ()	
Nombre	Imprime_consulta
Responsabilidades	Imprimir una consulta generada previamente.
Tipo	Sistema.
Referencias cruzadas	R31
Notas	Se debió generar una consulta previamente.
Excepciones	Cuando el usuario seleccione la opción de salir.
Salida	Enviará mensaje de envió de información a la impresora.
Pre-condiciones	Tener acceso al módulo, tener en pantalla la consulta y seleccionar la opción de imprimir.
Post-condiciones	Impresora en línea y configurada.

Contratos para el caso de uso: Cambiar Turno

Contrato: Validar_contraseña ()	
Nombre	Valida_contraseña
Responsabilidades	Verificar que la contraseña para el cambio de turno sea la correcta.
Tipo	Sistema.
Referencias cruzadas	R32
Notas	La contraseña se deberá proporcionar correctamente.
Excepciones	Enviará mensaje cuando la contraseña sea incorrecta o ya se haya hecho el cambio de turno.
Salida	Mostrará en pantalla la información del área de trabajo para el turno activo.
Pre-condiciones	Contar con la contraseña de seguridad y no haber hecho el cambio de turno previamente.
Post-condiciones	Hará el cambio de turno correspondiente.

Contrato: Cambiar_turno ()	
Nombre	Cambia_turno.
Responsabilidades	Actualizar la información para dejar pendientes al turno entrante.
Tipo	Sistema.
Referencias cruzadas	R33
Notas	Modificará información eliminando las averiguaciones con un trámite final y dejando solo las continuadas.
Excepciones	Enviará mensaje cuando la contraseña sea incorrecta.
Salida	Mostrará en pantalla la información del área de trabajo para el turno activo.
Pre-condiciones	Contar con la contraseña de seguridad y no haber hecho el cambio de turno previamente.
Post-condiciones	Modificará la información reorganizándola para el trabajo de la misma por el turno activo

ANEXO III DICCIONARIO DE DATOS

Tabla Entidades.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_entidad	Integer	2	99	Obligatoria	Campo llave para las entidades.
Nom_entidad	VarChar	50	Ninguno	Obligatoria	Nombre de las entidades.

Tabla Delegaciones.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_delegación	Integer	2	99	Obligatoria	Campo llave para las delegaciones.
Id_entidad	Integer	2	99	Obligatoria	Llave foránea para relacionar las delegaciones con la entidad.
Nom_delegación	Varchar	50	Ninguno	Obligatoria	Nombre de la delegación.

Tabla coordinaciones.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_coord.	Integer	4	9999	Obligatoria	Campo llave para las coordinaciones.
Id_entidad	Integer	2	99	Obligatoria	Llave foránea para relacionar las coordinaciones con la entidad.
Id_delegación	Varchar	2	99	Obligatoria	Llave foránea para relacionar las coordinaciones con la delegación.
Nom_coord	Varchar	50	Ninguno	Obligatoria	Nombre completo de la coordinación.
Clave_coord.	Varchar	10	Ninguno	Obligatoria	Abreviatura de la coordinación.

Tabla Colonias.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_colonia	Integer	4	9999	Obligatoria	Campo llave para las colonias.
Id_coord.	Integer	4	9999	Obligatoria	Llave foránea para relacionar las colonias con coordinaciones.
Id_entidad	Integer	2	99	Obligatoria	Llave foránea para relacionar las colonias con la entidad.
Id_delegación	Varchar	2	99	Obligatoria	Llave foránea para relacionar las colonias con la delegación.
Nom_colonia	Varchar	50	Ninguno	Obligatoria	Nombre completo de la colonia.
Codigo_postal	Number	5	99999	Obligatoria	Código postal de la colonia.

Tabla Turno

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_turno	Integer	2	99	Obligatoria	Campo llave para los turnos.
Nom_turno	VarChar	2	A9	Obligatoria	Descripción del turno activo.

Tabla tipo_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_tipo_ap	Integer	2	99	Obligatoria	Campo llave para los tipos de ap.
Nom_tipo_ap	VarChar	50	Ninguno	Obligatoria	Descripción del tipo de averiguación previa (ap).

Tabla resolución_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_resolución_ap	Integer	2	99	Obligatoria	Campo llave para las resoluciones de ap.
Nom_resolución_ap	VarChar	50	Ninguno	Obligatoria	Descripción de las resoluciones de ap.

Tabla Delito.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_delito	Integer	2	99	Obligatoria	Campo llave para los delitos.
Nom_delito	VarChar	50	Ninguno	Obligatoria	Descripción del delito.

Tabla modalidad.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_modalidad	Integer	2	99	Obligatoria	Campo llave para las modalidades.
Nom_modalidad	VarChar	50	Ninguno	Obligatoria	Descripción de la modalidad.

Tabla Delito_modalidad.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_delito	Integer	2	99	Obligatoria	Llave foránea para la relacionar el delito con la modalidad.
Id_modalidad	Integer	2	99	Obligatoria	Llave foránea para la relacionar la modalidad con el delito.

Tabla marcas_auto.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_marca	Integer	2	99	Obligatoria	Campo llave para las marcas de vehículos.
Nom_marca	VarChar	50	Ninguno	Obligatoria	Descripción de la marca del vehículo.

Tabla color.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_color	Integer	2	99	Obligatoria	Campo llave para los colores de vehículos.
Nom_color	VarChar	50	Ninguno	Obligatoria	Descripción de los colores de vehículos.

Tabla submarca_auto.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_submarca	Integer	2	99	Obligatoria	Campo llave para las modalidades.
Id_marca	Integer	2	99	Obligatoria	Llave Foránea para relacionar la marca con la submarca.
Nom_submarca	VarChar	50	Ninguno	Obligatoria	Descripción de la submarca de vehículos.

Tabla vehículos.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_vehículo	Integer	2	99	Obligatoria	Campo llave para los vehículos.
Id_marca	Integer	2	99	Obligatoria	Llave Foránea para relacionar la marca con el vehículo.
Id_color	Integer	2	99	Obligatoria	Llave Foránea para relacionar el color con el vehículo.
Modelo_auto	Number	4	9999	Obligatoria	Descripción del año del vehículo.
Placas_auto	VarChar	15	Ninguno	Obligatoria	Descripción del número de placas de los vehículos.

Tabla usuarios.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_usuario	Integer	2	99	Obligatoria	Campo llave para los usuarios.
Id_coord.	Integer	2	99	Obligatoria	Llave Foránea para relacionar el usuario con una coordinación.
Nombre_usuario	Varchar	50	Alfabético	Obligatorio	Nombre del usuario.
Paterno_usuario	VarChar	50	Alfabético	Obligatorio	Apellido paterno del usuario.
Materno_usuario	VarChar	50	Alfabético	Obligatorio	Apellido materno del usuario.
Número _ empleado (usuario)	Number	6	99999	Obligatoria	Número de empleado el cual será el nombre de usuario de acceso al sistema.
Cargo	VarChar	50	Alfabético	Obligatorio	Cargo que desempeña el usuario.
Password	VarChar	10	Ninguno	Obligatoria	Clave de acceso al sistema.

Tabla perfiles.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_perfil	Integer	2	99	Obligatoria	Campo llave para los perfiles.
Nombre_perfil	Varchar	20	Alfabético	Obligatoria	Nombre del perfil.
Agregar	Boolean	2	Boolean	Obligatoria	Permiso para agregar.
Modificar	Boolean	2	Boolean	Obligatoria	Permiso para modificar.
Borrar	Boolean	2	Boolean	Obligatoria	Permiso para borrar.
Consultas	Boolean	2	Boolean	Obligatoria	Permiso para consultar.
Agregar	Boolean	2	Boolean	Obligatoria	Permiso para agregar
Imprimir	Boolean	2	Boolean	Obligatoria	Permiso para imprimir.
Reportes	Boolean	2	Boolean	Obligatoria	Permiso para acceder a reportes.
Catálogos	Boolean	2	Boolean	Obligatoria	Permiso para acceder a catálogos.
Cambiar turno	Boolean	2	Boolean	Obligatoria	Permiso para cambiar turno.

Tabla perfil_usuario.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_usuario	Integer	2	99	Obligatoria	Llave foránea para relacionar a un usuario con su perfil.
Id_perfil	Integer	2	99	Obligatoria	Llave Foránea para relacionar el perfil con el usuario.

Tabla número_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_num_ap	Integer	4	9999	Obligatoria	Campo llave para los número de ap.
Id_turno	Integer	2	99	Obligatoria	Llave Foránea para relacionar un turno al número de ap.
Id_coord.	Integer	2	99	Obligatoria	Llave Foránea para relacionar una coordinación al número de ap.
Id_delegacion	Integer	2	99	Obligatoria	Llave Foránea para relacionar una delegación al número de ap.
Num_ap	VarChar	50	Ninguno	Obligatorio	Número completo de la averiguación previa.

Tabla delito_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_delito	Integer	2	99	Obligatoria	Llave foránea para relacionar a un delito con la ap.
Id_ap	Integer	4	9999	Obligatoria	Llave Foránea para relacionar una ap con el delito.

Tabla denunciante_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_denunciante	Integer	4	9999	Obligatoria	Llave foránea para relacionar a un denunciante con una ap.
Id_ap	Integer	4	9999	Obligatoria	Llave Foránea para relacionar una ap con un denunciante.

Tabla detenido_ap.

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_detenido	Integer	4	9999	Obligatoria	Llave foránea para relacionar a un detenido con una ap.
Id_ap	Integer	4	9999	Obligatoria	Llave Foránea para relacionar una ap con un detenido.

Tabla detenido

Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_detenido	Integer	4	9999	Obligatoria	Campo Llave para los detenidos.
Id_colonia	Integer	4	9999	Obligatoria	Llave Foránea para relacionar una colonia con un detenido.
Id_coord.	Integer	2	99	Obligatoria	Llave Foránea para relacionar una coordinación al detenido.
Id_delegacion	Integer	2	99	Obligatoria	Llave Foránea para relacionar una delegación al detenido.
Nombre	Varchar	50	Alfabético	Obligatoria	Nombre del detenido.
Paterno	Varchar	50	Alfabético	Obligatoria	Apellido paterno del detenido.
Materno	Varchar	50	Alfabético	Obligatoria	Apellido materno del detenido.
Edad	Number	2	99		Edad del detenido.
Sexo	Bolean	2			Sexo del detenido.
Identificación	Varchar	30	Alfabético		Tipo de identificación.
Calle	Varchar	30	Alfabético	Obligatoria	Calle del indiciado.
Num_exterior	Varchar	10	Ninguno		Número exterior.
Num_interior	Varchar	10	Ninguno		Número interior.
Cp	Number	6	automatico	Obligatoria	Código postal.
teléfono	Number	12	999999999999		Número telefónico del detenido.

Tabla denunciante

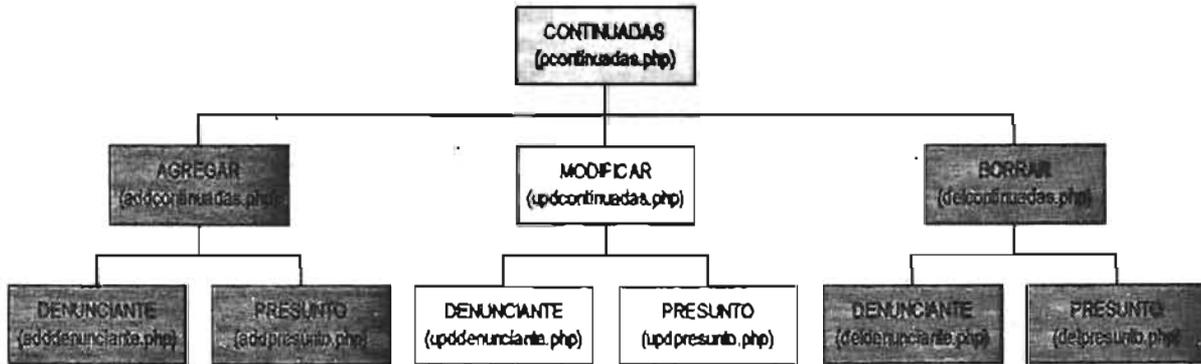
Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_denunciante	Integer	4	9999	Obligatoria	Campo Llave para los detenidos.
Id_colonia	Integer	4	9999	Obligatoria	Llave Foránea para relacionar una colonia con un denunciante.
Id_coord.	Integer	2	99	Obligatoria	Llave Foránea para relacionar una coordinación al denunciante.
Id_delegacion	Integer	2	99	Obligatoria	Llave Foránea para relacionar una delegación al denunciante.
Nombre	Varchar	50	Alfabético	Obligatoria	Nombre del detenido.
Paterno	Varchar	50	Alfabético	Obligatoria	Apellido paterno del denunciante.
Materno	Varchar	50	Alfabético	Obligatoria	Apellido materno del denunciante.
Edad	Number	2	99		Edad del denunciante.
Sexo	Boolean	2			Sexo del detenido.
Identificación	Varchar	30	Alfabético		Tipo de identificación.
Calle	Varchar	30	Alfabético	Obligatoria	Calle del denunciante.
Num_exterior	Varchar	10	Ninguno		Número exterior.
Num_interior	Varchar	10	Ninguno		Número interior.
Cp	Number	6	automatico	Obligatoria	Código postal.
teléfono	Number	12	999999999999		Número telefónico del denunciante.

Tabla ap

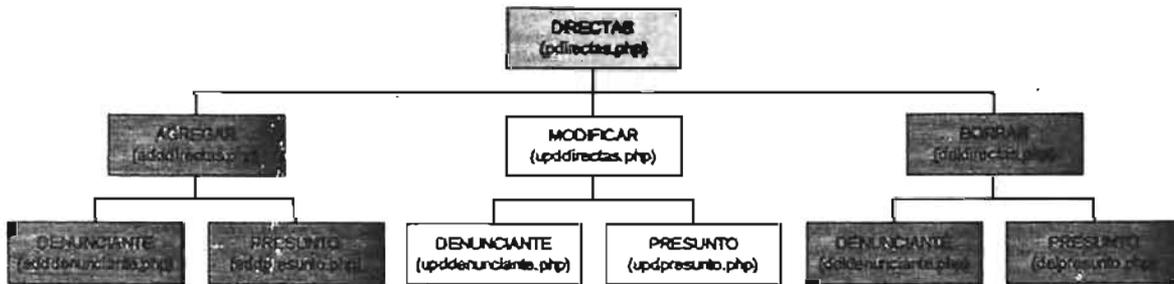
Campo	Tipo	Tamaño	Formato	Validación	Descripción
Id_ap	Integer	4	9999	Obligatoria	Campo Llave para las ap.
num_ap	Integer	4	9999	Obligatoria	Llave foránea para asignar un número de ap a la ap.
Id_tipo_ap	Integer	2	99	Obligatoria	Campo llave para los tipos de ap.
Fecha_guardia	Date	6	ddmmaa	Obligatoria	Fecha de la guardia en que se inicia la ap.
Fecha_inicio	Date	6	ddmmaa	Obligatoria	Fecha en que se inicia la ap.
Hora_inicio	Time	4	Hhmm	Obligatoria	Hora en que se inicia la ap.
turno	Integer	2	99	Obligatoria	Llave Foránea para relacionar un turno a la ap.
Calle_hechos	Varchar	50	Alfabético		Calle de los hechos.
Calle2_hechos	Varchar	50	Alfabético		Entre que calle.
Calle3_hechos	Varchar	50	Alfabético		Y Calle.
Colonia	Integer	4	9999		Llave Foránea para relacionar una colonia la ap.
delegacion	Integer	2	99		Llave Foránea para relacionar una delegación a la ap.
coordinacion	Integer	2	99		Llave Foránea para relacionar una coordinación la ap.
Id_entidad	Integer	2	99		Llave foránea para relacionar las una entidad con la ap.
Codigo_postal	Number	5	99999		Código postal de la colonia.
Fecha_hechos	Date	6	ddmmaa	Obligatoria	Fecha en que ocurrieron los hechos.
Hora_hechos	Time	4	Hhmm	Obligatoria	Hora en que ocurrieron los hechos.
Resumen_hechos	Memo	400	Ninguno	Obligatoria	Resumen de los hechos.
Monto_ap	Number	10	9999999999		Monto de los daños o del robo.
Resolucion_ap	Integer	2	99	Obligatoria	Llave foránea para relacionar la resolución con una ap.
Observaciones	Varchar	50	Alfabético		Observaciones de la ap.

ANEXO IV ESQUEMAS DE NAVEGACION.

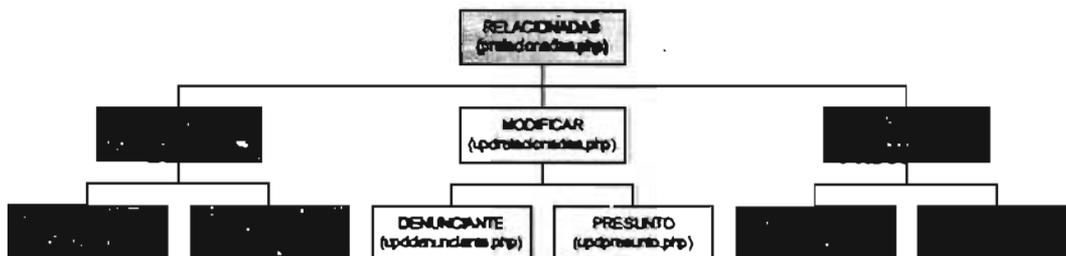
ESQUEMA DEL SUBMÓDULO DE CONTINUADAS.



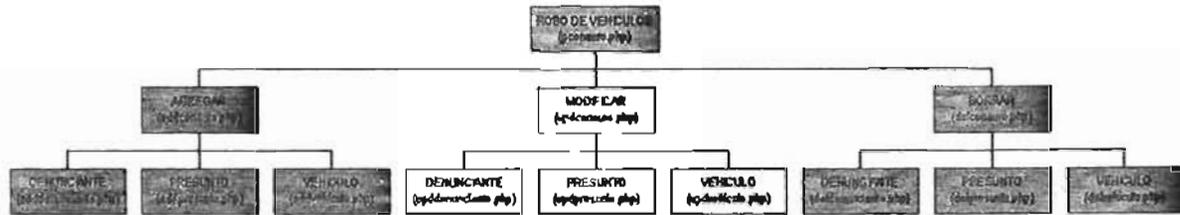
ESQUEMA DEL SUBMÓDULO DE DIRECTAS.



ESQUEMA DEL SUBMÓDULO DE RELACIONADAS.



ESQUEMA DEL SUBMÓDULO DE ROBO DE VEHÍCULOS (CONAURO).



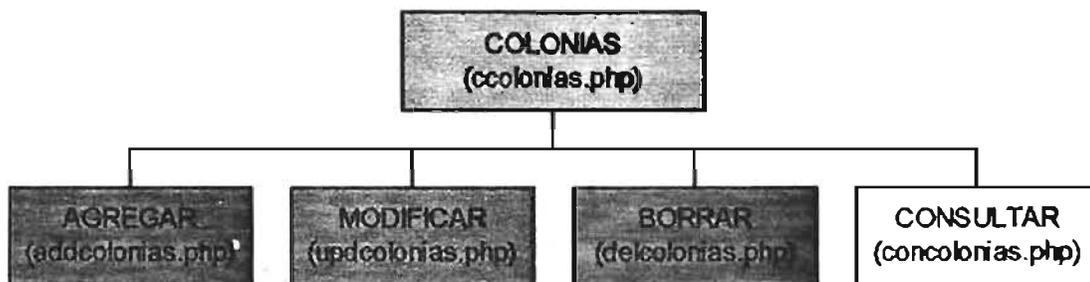
ESQUEMA DEL SUBMÓDULO DELEGACIONES.



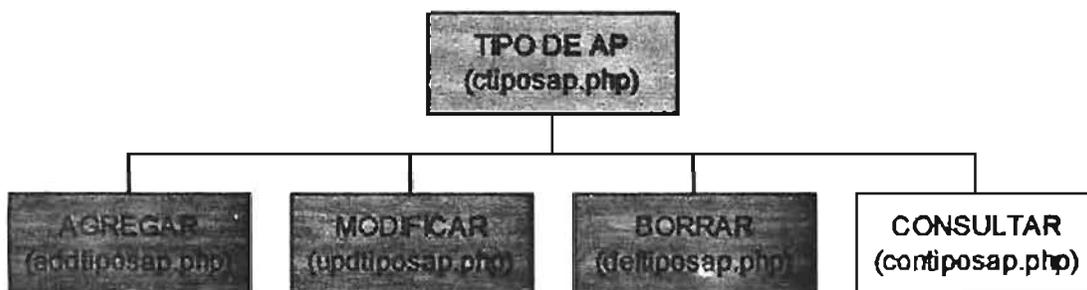
ESQUEMA DEL SUBMÓDULO COORDINACIONES.



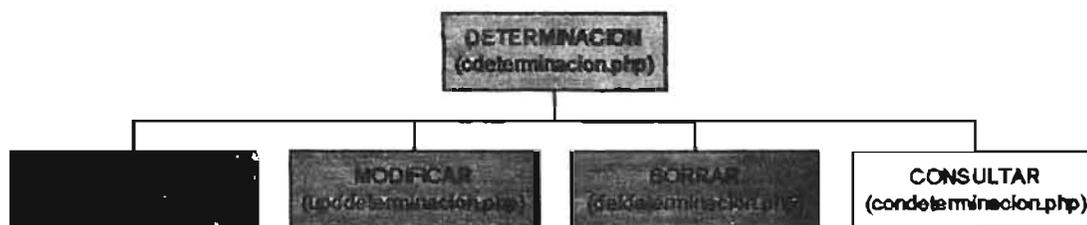
ESQUEMA DEL SUBMÓDULO COLONIAS.



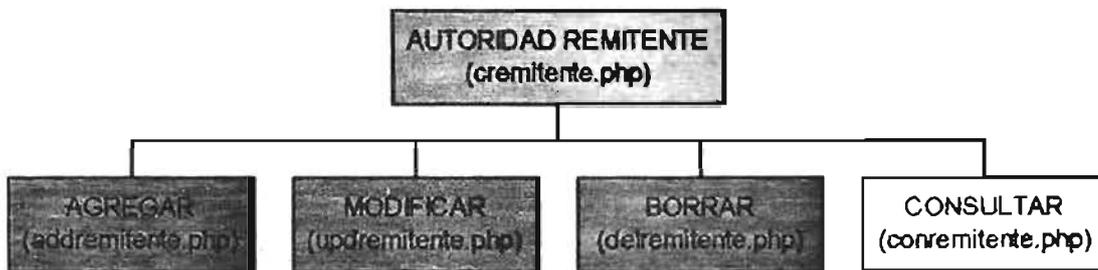
ESQUEMA DEL SUBMÓDULO TIPO DE A.P.



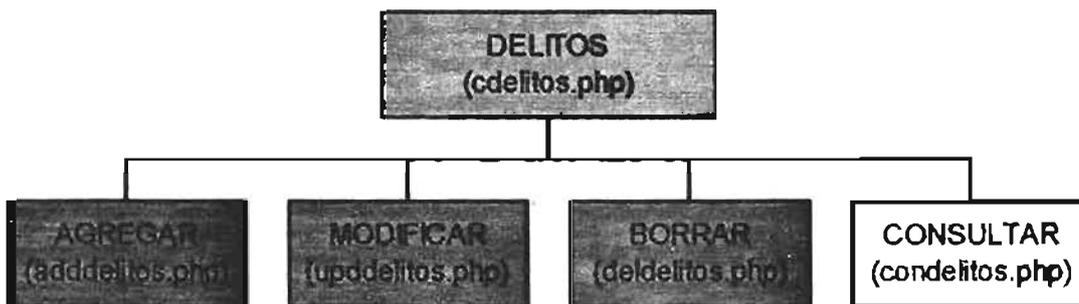
ESQUEMA DEL SUBMÓDULO DETERMINACIÓN DE A.P.



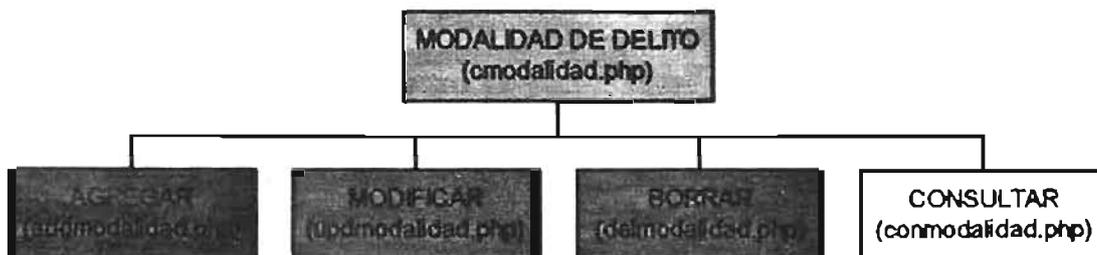
ESQUEMA DEL SUBMÓDULO DE AUTORIDADES REMITENTES.



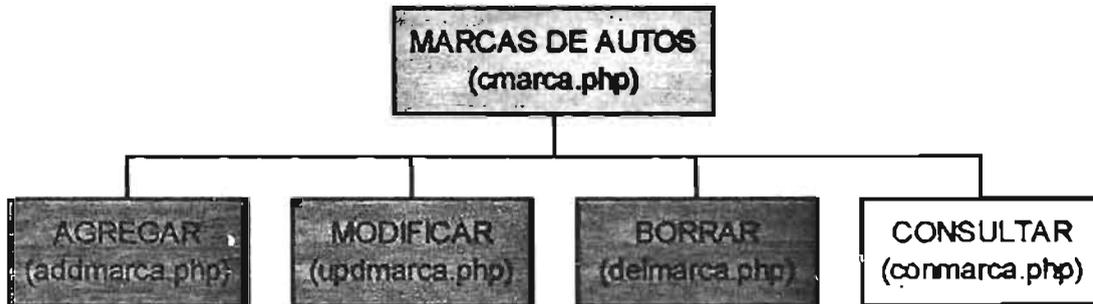
ESQUEMA DEL SUBMÓDULO DELITOS.



ESQUEMA DEL SUBMÓDULO MODALIDAD DE DELITOS.



ESQUEMA DEL SUBMÓDULO MARCAS DE AUTO.



ESQUEMA DEL SUBMÓDULO SUBMARCAS DE AUTO.



ESQUEMA DEL SUBMÓDULO COLORES DE AUTO.



ANEXO V INTERFAZ DE USUARIO Y REGLAS DE NEGOCIO

MÓDULO DE GENERACION DE REPORTES

Página para la generación de reportes:

SELECCIONE LA(S) OPCIÓN(ES) DESIADADA(S)

Ante de seleccionar algunas un criterio de búsqueda

DATOS DE EJERCICACION PREVIA

DELITO
ROBO A TRANSEUNTE

RANGO DE FECHAS
DESDE: 01/06/04 HASTA: 01/06/04

COORDINADOR
COLOMBIA
ANTIOQUIA

DELEGACION
MAGUEL Y CALGO

COORDINACION TERRITORIAL
M+H

Generar Reporte Limpiar Pegar

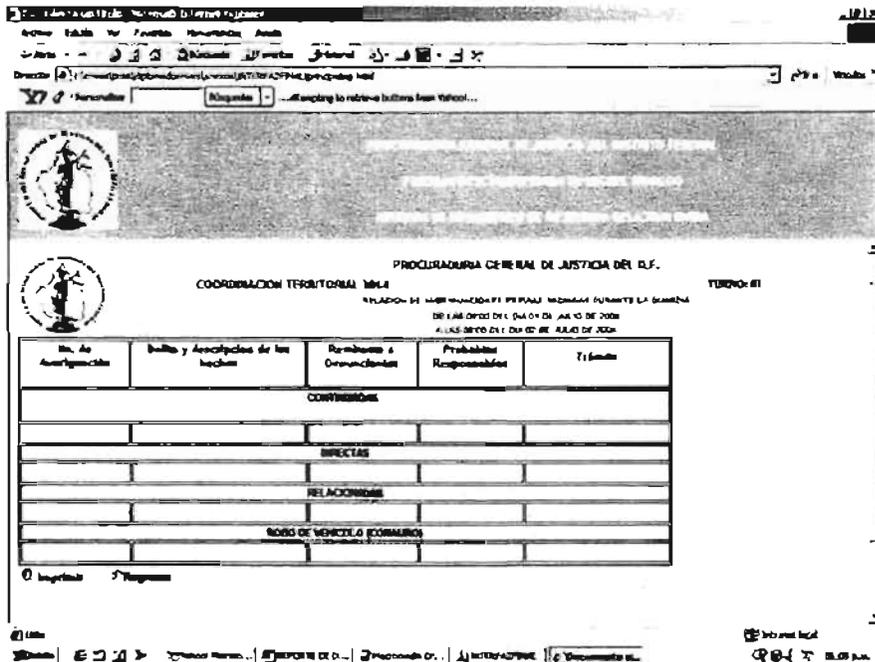
Reglas de negocio:

- El perfil debe tener acceso a la página.
- Seleccionar cuando menos un criterio de búsqueda.
- El formato de fechas será dd/mm/aa.
- Si localiza información con dichos criterios la mostrará en pantalla al usuario con opción para imprimirla.

MÓDULO DE CONSULTAS

Para ejemplificar el módulo de consultas se tomará las consultas que puede realizar un ministerio público.

Página de consulta de rol de guardia para Ministerios Públicos.

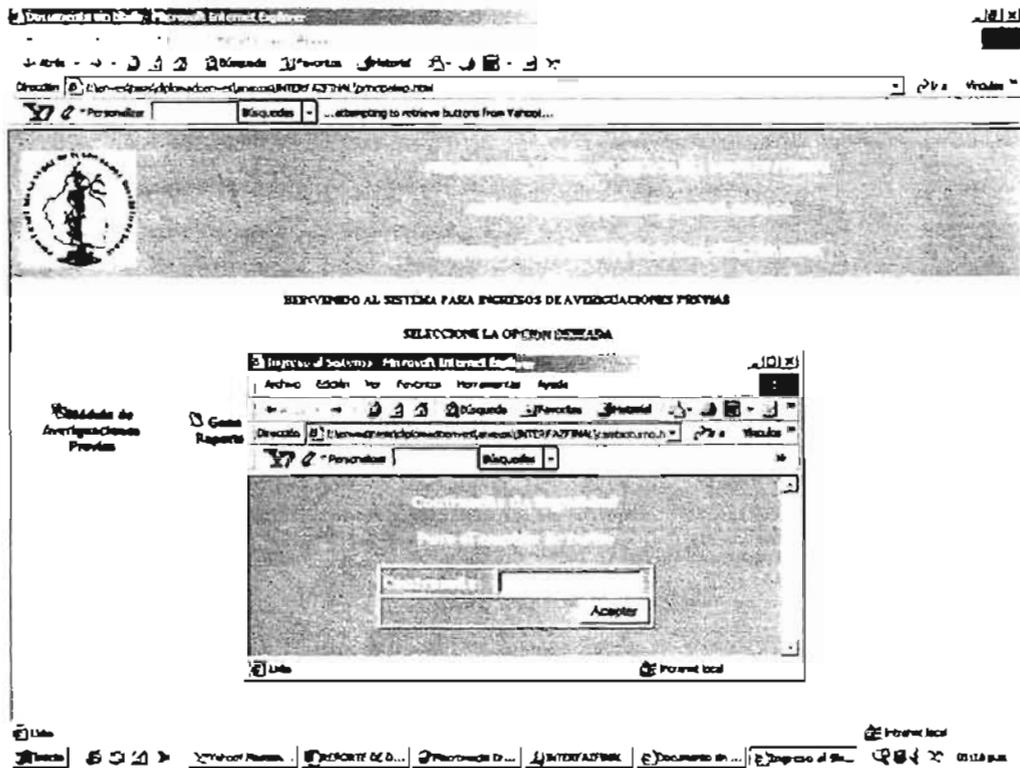


Reglas de Negocio:

- Si selecciona imprimir se enviará el rol de guardia a la impresora y regresará a la pantalla de trabajo para ministerios públicos.
- Si elige regresar irá a la pantalla de ministerios públicos.

MÓDULO DE CAMBIO DE TURNO

PÁGINA PARA EL CAMBIO DE TURNO.



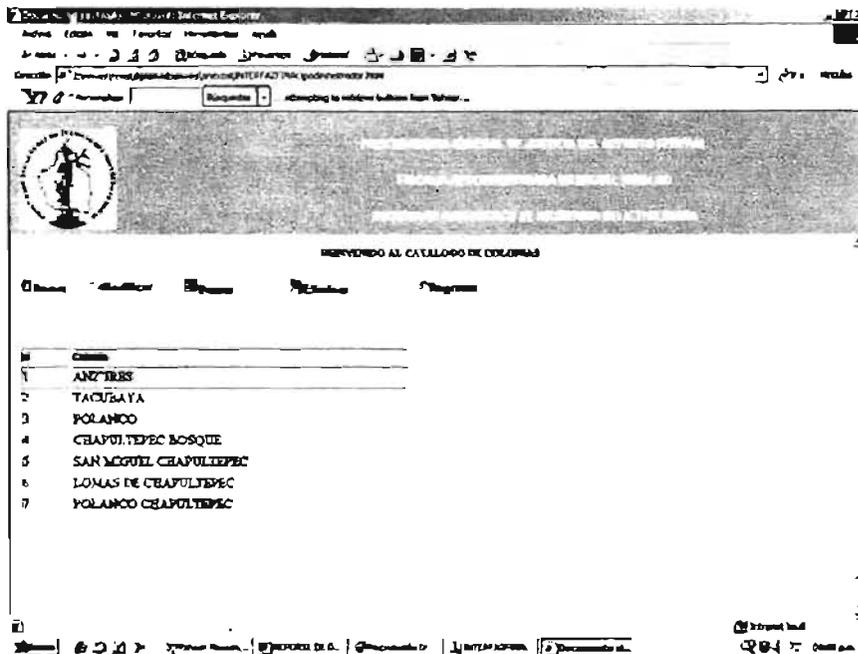
Reglas de Negocio:

- Sólo cambiará de turno si la contraseña es correcta.
- Si la fecha del nuevo turno es la del día corriente, enviará un mensaje de alerta.
- Si el perfil del usuario no pertenece a dicho turno enviará mensaje de aviso al usuario.
- Al hacer el cambio de turno, el sistema actualizará la información de las averiguaciones previas, quitando de pantalla a las que se les ha dado una determinación y dejando las que se encuentran pendientes con el rubro de continuadas.

MÓDULO DE CATÁLOGOS.

Por la extensión de los catálogos, se mostrarán las pantallas para alta, baja, cambio y consulta de un catálogo (Colonias), tomando en cuenta que los catálogos restantes son muy similares.

Página principal de Catálogos (Colonias)



Reglas de negocio:

- Tener perfil con acceso a la Página.
- Actualizará el listado de colonias cuando haya un cambio en la información del catálogo.

Página para agregar colonias.

The screenshot shows a web browser window with the title 'Agregar una colonia'. The browser's address bar shows a URL starting with 'http://www.pgcj.df.gob.mx/'. The page content includes a logo on the left and a form with the following fields and values:

Agregar una colonia	
Entidad	DISTRITO FEDERAL
Delegación	MIGUEL HIDALGO
Coordinación	MH-1
Colonia	

Below the form are three buttons: 'Guardar', 'Limpiar', and 'Regresar'.

Reglas de Negocio:

- Se debe elegir la coordinación donde se encontrará la colonia.
- El sistema hará selección automática dependiendo de la opción elegida, ejemplo: si elige como delegación a Miguel Hidalgo, sólo aparecerán las coordinaciones MH-1, MH-2, etc., y la entidad se posicionará en Distrito federal.
- Se debe llenar obligatoriamente el campo del nombre de la colonia.
- Al guardar la información regresará a la Página principal del catálogo en cuestión, actualizando el listado que se encuentra en la misma.

Página para modificar colonias:

Modificar una colonia

Estado	DISTRITO FEDERAL
Municipio	MIGUEL HIDALGO
Cuartelamiento	M-1
Colonia	ANZURES

Guardar Limpiar Regresar

Reglas de negocio:

- El sistema solicitará el nombre de la colonia para localizar el registro y presentarlo en pantalla.
- Se debe cambiar la información deseada en el catálogo de colonia.
- Se debe llenar obligatoriamente el campo del nombre de la colonia.
- Al guardar la información regresará a la página principal del catálogo en cuestión, actualizando el listado que se encuentra en la misma.

Página para eliminar colonias.

The screenshot shows a web browser window displaying a form titled "Borrar una colonia". The form contains the following fields:

Borrar una colonia	
Estado:	DISTRITO FEDERAL
Delegación:	MIGUEL HIDALGO
Coordinación:	JH-1
Colonia:	ANZURES

Below the form are two buttons: "Borrar" and "Regresar".

The page header includes the logo of the Procuraduría General de Justicia del Distrito Federal and the text: "PROCURADURÍA GENERAL DE JUSTICIA DEL DISTRITO FEDERAL", "FISCALÍA DESCENTRALADA EN MIGUEL HIDALGO", and "SISTEMA DE ESTADÍSTICO DE INCIDENCIA DELICTIVA DIARIA".

Reglas de negocio:

- Se pedirá una clave de seguridad para el borrado de la colonia.
- Sólo se borrara el registro si no existe relación de la misma con otras tablas.
- Enviará mensaje de confirmación para el borrado del registro.
- Actualizará la información de la Página principal del catálogo de colonias.

