



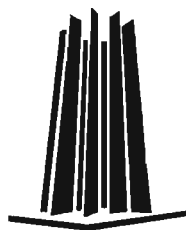
**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
CAMPUS ARAGÓN**

**“GESTIÓN DE SERVICIOS DEL GOBIERNO DEL  
DISTRITO FEDERAL EN LÍNEA ATRAVÉS DE WEB”**

**T R A B A J O E S C R I T O  
EN LA MODALIDAD DE SEMINARIOS  
Y CURSOS DE ACTUALIZACIÓN Y  
CAPACITACIÓN PROFESIONAL  
QUE PARA OBTENER EL TÍTULO DE:  
I N G E N I E R O E N C O M P U T A C I Ó N  
P R E S E N T A :  
E D G A R R I O S L I N A R E S**

**ASESOR: ING. VÍCTOR RAMÓN AGUILAR OCAMPO**



**MÉXICO, 2005.**

m. 341595



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.

A mi madre

Puesto que muero existo  
Xavier Villaurrutia

# INDICE

<b>OBJETIVO GENERAL DEL DIPLOMADO DE DESARROLLO DE SISTEMAS EN WEB .....</b>	<b>1</b>
<b>CONTENIDOS A TRATAR EN EL DIPLOMADO .....</b>	<b>1</b>
<b>MÓDULO 1. FUNDAMENTOS DEL DESARROLLO DE SISTEMAS EN WEB .....</b>	<b>2</b>
<b>MÓDULO 2. ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADOS AL WEB .....</b>	<b>3</b>
<b>MÓDULO 3. INTERFASE DE USUARIO .....</b>	<b>4</b>
<b>MÓDULO 4. CAPA DE DATOS.....</b>	<b>5</b>
<b>MÓDULO 5. CAPA DE REGLAS DE NEGOCIO .....</b>	<b>6</b>
<b>MÓDULO 6. IMPLEMENTACIÓN .....</b>	<b>7</b>
<b>CASO PRÁCTICO. GESTIÓN DE SERVICIOS DEL GOBIERNO DEL DISTRITO FEDERAL EN LÍNEA A TRAVÉS DE WEB..</b>	<b>8</b>
<b>TEMA 1 ANTECEDENTES .....</b>	<b>8</b>
1.1    DEFINICIÓN DE SERVICIO.....	8
1.2    SOLICITUDES DE SERVICIOS POR INTERNET .....	8
1.3    SERVICIOS DEL MANUAL DE TRÁMITES Y SERVICIOS AL PÚBLICO.....	12
<b>TEMA 2 PLANTEAMIENTO DEL PROBLEMA .....</b>	<b>14</b>
2.1    ARQUITECTURA DE LOS SISTEMAS DE INFORMACIÓN.....	14
2.1.1    Arquitectura en Dos Capas .....	14
2.1.2    Arquitectura en Tres Capas .....	15
2.2    APLICACIONES DE BASES DE DATOS EN WEB.....	18
2.2.1    Arquitectura .....	18
2.2.2    Capa de Cliente .....	19
2.2.3    Capa de Aplicación .....	20
2.2.4    Capa de Datos .....	20
2.3    INVESTIGACIÓN DEL PROBLEMA .....	23
2.3.1    Definición del Problema .....	23
2.3.2    Objetivo General .....	24
2.3.3    Objetivos Específicos.....	24
2.3.4    Alcance y Limites .....	24
2.3.5    Costo de Desarrollo .....	25
2.3.6    Costo de Operación .....	26
2.3.7    Beneficios.....	27
<b>TEMA 3 ANÁLISIS Y DISEÑO .....</b>	<b>29</b>
3.1    DETERMINACIÓN DE REQUERIMIENTOS .....	29
3.2    SUBSISTEMAS DEL SISTEMA .....	31
3.3    CASOS DE USO.....	31
3.4    CONTRATOS DE LOS CASOS DE USO.....	41
3.5    DIAGRAMAS DE SECUENCIA.....	59
<b>TEMA 4 DISEÑO DE INTERFAZ.....</b>	<b>61</b>
4.1    INTRODUCCIÓN .....	61
4.2    SUBSISTEMAS.....	61
4.2.1    Módulo de Registro .....	63
4.2.2    Módulo de Seguimiento .....	71
4.2.3    Módulo de Reportes.....	75
4.2.4    Módulo de Catálogos .....	76
4.2.5    Módulo de Administración.....	93
4.2.6    Módulo de Orientación .....	99

<b>TEMA 5 DISEÑO DE LA BASE DE DATOS .....</b>	<b>100</b>
5.1 DISEÑO DE LA BASE DE DATOS .....	100
5.2 DICCIONARIO DE DATOS .....	101
5.3 SCRIPT DE CREACIÓN DE LA BASE DE DATOS .....	111
5.4 SCRIPTS DE CONSULTA Y ELIMINACIÓN .....	117
<b>TEMA 6 DISEÑO DE LAS REGLAS DE NEGOCIO.....</b>	<b>119</b>
6.1 REGLAS DEL MODELO DE DATOS .....	119
6.2 REGLAS DE RELACIÓN.....	121
6.3 REGLAS DE RESTRICCIÓN .....	122
6.4 REGLAS DE FLUJO .....	122
6.5 VALIDACIONES.....	124
<b>TEMA 7 IMPLEMENTACIÓN.....</b>	<b>125</b>
7.1 PLAN DE IMPLEMENTACIÓN.....	125
7.2 ESTÁNDARES DE PROGRAMACIÓN .....	131
<b>CONCLUSIONES DEL DIPLOMADO .....</b>	<b>138</b>
<b>ANEXO A. SERVICIOS DEL MTYSP .....</b>	<b>139</b>
<b>BIBLIOGRAFÍA.....</b>	<b>143</b>

## **OBJETIVO GENERAL DEL DIPLOMADO DE DESARROLLO DE SISTEMAS EN WEB**

Proporcionar a los participantes el conocimiento y habilidad requeridos para analizar, diseñar e implementar aplicaciones de uso interactivo y con intercambio de datos a través de una interfase WWW, de tal forma que se ofrezca una ventaja competitiva con respecto a los otros entornos donde no existe el WWW:

### **CONTENIDOS A TRATAR EN EL DIPLOMADO**

MÓDULO 1. FUNDAMENTOS DEL DESARROLLO DE SISTEMAS EN WEB.

MÓDULO 2. ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADOS AL WEB

MÓDULO 3. INTERFASE DE USUARIO

MÓDULO 4. CAPA DE DATOS

MÓDULO 5. CAPA DE REGLAS DE NEGOCIO

MÓDULO 6. IMPLEMENTACIÓN

## **MÓDULO 1. FUNDAMENTOS DEL DESARROLLO DE SISTEMAS EN WEB**

### **OBJETIVOS**

- Comprender la mecánica del diplomado.
- Analizar, proponer y diseñar una infraestructura de Hardware y Software para dar soporte a un Sistema que será utilizado en línea vía WWW.
- Conocer los fundamentos del modelo en 3 capas.

Se llevo acabo un análisis del software libre y en específico sobre el sistema operativo Linux, del cual se realizó un análisis de su historia, características generales, las distribuciones más conocidas y aplicaciones más comunes. Así mismo se realizó su instalación, configuración y utilización de comandos básicos.

Se estudió la instalación y configuración del servidor web Apache, del modulo Interprete del Lenguaje Script PHP y el servidor de bases de datos MySQL.

Como último punto del módulo se estudio la tecnología de 3 capas: presentación, datos y reglas de negocio; En donde se detallaron todos los elementos que componen cada una de las antes mencionadas capas y los distintos enfoques para la implementación de un sistema de base de datos en WWW.

## **MÓDULO 2. ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADOS AL WEB**

### **OBJETIVOS**

- El alumno analizará el sistema y los recursos para desarrollarlo.
- Diseñará la base de datos y documentará el sistema.

En el módulo 2 se revisó como debe realizarse el análisis de un sistema de información, se estudió la metodología Yourdon y el Ciclo de Vida de un sistema de información en cada una de sus diferentes etapas: La encuesta, el análisis de sistemas, el diseño, la implantación, la generación de pruebas de aceptación, la garantía de calidad, la descripción del procedimiento, la conversión de base de datos y la instalación.

También pudimos ver lo que se debe analizar al momento de diseñar un sistema de información, la forma en que se debe realizar el análisis de Hardware y Software, la forma en que se debe realizar el análisis de factibilidad, la estimación de tiempo, la estimación de costos, estimación de peligros y las reglas para la estimación, y la forma de realizar el análisis de costo / beneficio.

Otro punto que se abarcó en el módulo se refirió a los elementos que se deben diseñar para la construcción de un sistema, nos abocamos al diseño de la base de datos desde un punto de vista conceptual, la forma de identificar las entidades involucradas en el sistema, la forma de identificar los atributos y características de dichas entidades, revisamos como identificar las tablas y sus campos de un sistema, como realizar las distintas formas de relaciones entre entidades, todo lo anterior para obtener un Diagrama Entidad - Relación.

Se realizó una revisión de lo que se debe modelar en un sistema, para lo cuál se estudiaron los diagramas de flujo de datos y sus componentes: Proceso, flujo, almacén, terminador. El Manual Técnico y el manual de Usuario como parte de la documentación que se debe generar sobre el sistema fue otro de los temas que se vieron.

Y como último punto se estudió la Metodología UML para el modelado de sistemas de información, se analizaron los diagramas de casos de uso y diagramas de secuencia, se estudiaron los casos de uso, sus componentes y la forma de hacerlos; así mismo se estudiaron los contratos de los casos de uso y su creación.



### **MÓDULO 3. INTERFASE DE USUARIO**

#### **OBJETIVOS**

- El alumno analizará los requerimientos de interacción entre el usuario y el sistema.
- Diseñará las interfases del sistema.
- Construirá las interfases del sistema.

En éste módulo se explicó las características de la interfase de un sistema de información y como funciona una interfase de un sistema en un ambiente Web, se analizó la forma de diseñar un sistema basado en Web y los principios que se deben considerar para el diseño de una interfase: diseño, congruencia con los procesos, distribución lógica, navegación y ayuda; se estudiaron las estructuras posibles de las páginas que componen una interfase y las recomendaciones para organizar los elementos de una interfase y así mismo se analizaron los elementos de una interfase Web y como emplearlos: encabezados, imágenes, frames, listas, formularios, tablas, botones, cuadros de diálogo y campos de texto.

Se estudió el lenguaje HTML para la creación de páginas Web y propiamente para la construcción de la interfase de sistemas Web, se estudiaron los diversos elementos que componen al lenguaje para la creación de interfases: listas, tablas, imágenes, hipervínculos y formularios, y como implementar Javascript dentro de páginas hechas con HTML.

Como último punto se estudió los inconvenientes de las interfases diseñadas con HTML puro, por lo cuál se analizó las características de las Hojas de Estilo mejor conocidas como CSS, se estudió además de sus características, sus ventajas, la sintaxis y su construcción e implementación.

## **MÓDULO 4. CAPA DE DATOS**

### **OBJETIVOS**

- Entender y utilizar el lenguaje SQL para diseñar consultas y manipular información de una base de datos.
- Explicar el propósito del Acceso Universal a datos.

En el módulo 4 se estudió el modelo de datos, el significado y la importancia de las Bases de Datos, la clasificación de los Sistemas Gestores de Bases de Datos: modelo relacional, modelo de red, modelo jerárquico, modelo orientado a objetos; El objetivo, la arquitectura, las características y funciones de un Sistema Gestor de Bases de Datos.

Se estudió el SQL, como lenguaje de gestión con la base de datos, en sus 2 partes: Lenguaje Definición de Datos y Lenguaje de Manejo de Datos. Con el Lenguaje Definición de Datos se estudió la forma de crear bases de datos, tablas con sus campos, índices y cambio de la definición de campos; y con el Lenguaje de Manejo de Datos se estudió la forma de realizar distintos tipos de consultas, agregar registros, modificar y borrar datos.

Se estudió la normalización de base de datos, su importancia y las 3 reglas de normalización de bases de datos.

Para terminar el módulo se estudió MySQL como Sistema Gestor de Bases de Datos, su estructura, la seguridad, como crear Bases de Datos, tablas y consultas. Así mismo se realizó un estudio sobre las características del Sistema Gestor de Bases de Datos llamado PostgreSQL.

## MÓDULO 5. CAPA DE REGLAS DE NEGOCIO

### OBJETIVOS

- El alumno identificará las necesidades principales del proceso que se desea incorporar al Web e implementará los productos y/o servicios que integran un sistema de información de este tipo.
- Reconocerá la necesidad de mantener y controlar un sitio apoyado por una base de datos.

En éste módulo se estudiaron las Reglas de Negocio que se deben aplicar a un sistema de información: origen, importancia y clasificación (Reglas del modelo de datos, reglas de relación, reglas de derivación, reglas de restricción y reglas de flujo). Se estudió en que capa de un sistema cliente / servidor deben ir las reglas de negocio y la logística de un sitio WWW. Se analizaron las mejores prácticas en análisis y diseño orientado a objetos.

Se estudió PHP como lenguaje de programación para la implementación de las reglas de negocio, se revisaron: la sintaxis, variables, arrays, cadenas, funciones, control de flujo, operadores, pasar variables por URL y formularios, la forma de conexión a bases de datos para ingreso, consulta, borrado y actualización de registros.

Se instaló y trabajó con un módulo de acceso a datos denominado **adoDB** para facilitar la conexión para presentación, consulta, borrado y actualización de registros en una base de datos.

Se instaló y trabajó con un módulo de seguridad desarrollado en php denominado **phpSecurePages** para poder implementar la seguridad en un sistema de información con manejo de bases de datos.

Se instaló y trabajó con un módulo para la creación de documentos en formato PDF denominado **FPDF** para poder generar documentos de consulta formateados al vuelo, con la finalidad de generar reportes que sean fáciles de imprimir, almacenar y distribuir.

## **MÓDULO 6. IMPLEMENTACIÓN**

### **OBJETIVOS**

- Implementar los productos y servicios de software que fueron conceptualizados en las etapas previas del diplomado aplicando los conocimientos adquiridos.

En lo correspondiente a éste módulo se aplicaron las técnicas, metodologías y recomendaciones para el diseño y desarrollo de los sistemas de información, a través del análisis de casos prácticos y concretos con 3 ejercicios: Tienda de libros, administración de autobuses de viaje y administración de estacionamientos.

Cabe mencionar que así mismo se mencionaron una serie de sugerencias para la presentación de la implementación de un sistema de información.

## **CASO PRÁCTICO. GESTIÓN DE SERVICIOS DEL GOBIERNO DEL DISTRITO FEDERAL EN LÍNEA A TRAVÉS DE WEB**

### **TEMA 1 ANTECEDENTES**

#### **1.1 DEFINICIÓN DE SERVICIO**

Antes de iniciar con el análisis del problema del caso práctico en cuestión es importante definir algunos conceptos sobre el tema, para ello comenzaremos con una disertación sobre el significado de servicio; Ya que no existe una definición clara de lo que es un servicio, lo que debemos hacer es diferenciar entre lo que es un servicio y lo que es un trámite, para ello definiremos al trámite como la gestión que debe realizar el gobernado ante la Administración Pública<sup>1</sup> con objeto de que se declare la generación, modificación, transmisión, revalidación o extinción de un derecho, en virtud de estar previsto en la ley, como ejemplos de trámites tenemos la expedición y revalidación de licencias, autorizaciones o permisos, la transmisión de derechos generados por éstos, la expedición de certificados, entre otros; y como ejemplo de servicios tenemos la limpieza de calles, la recolección de desechos, el suministro de agua potable, la orientación y asesoría, la atención a quejas y denuncias, el transporte público, etc.

Para los efectos de éste trabajo se tomarán en cuenta todos aquellos servicios incluidos en el Manual de Trámites y Servicios al Público del Distrito Federal publicado en la Gaceta Oficial del Distrito Federal No 44-Bis del día 3 de Junio del 2003 con la modificación del 30 de Julio del 2004 publicado en la Gaceta Oficial del Distrito Federal No 74-Bis.

#### **1.2 SOLICITUDES DE SERVICIOS POR INTERNET**

La Administración Pública juega un papel central en el funcionamiento de nuestra sociedad al ser la encargada de proporcionar servicios y autorizar trámites a los gobernados, dichos trámites y servicios son procedimientos administrativos<sup>2</sup> perfectamente descritos, aunque no siempre bien conocidos por los ciudadanos.

Dichos procesos se diseñan de modo que las operaciones que realizan cumplan y hagan cumplir la legislación vigente y aplicable en caso concreto. Una parte esencial de dichos procesos es el valor probatorio de cada uno de los documentos con los que se cubren los requisitos.

---

<sup>1</sup> Según la Ley de Procedimiento Administrativo para el Distrito Federal en su Artículo 2º: "Administración Pública: Dependencias y entidades que integran a la Administración Central y Paraestatal del Distrito Federal, en los términos de la Ley Orgánica de la Administración Pública del Distrito Federal".

<sup>2</sup> Según la Ley de Procedimiento Administrativo para el Distrito Federal en su Artículo 2º: "Procedimiento Administrativo: Conjunto de trámites y formalidades jurídicas que preceden a todo acto administrativo, como su antecedente y fundamento, los cuales son necesarios para su perfeccionamiento, condicionan su validez y persiguen un interés general".

Cuando un ciudadano solicita un trámite o servicio ante la administración, la primera etapa, en cualquier caso, es solicitarla a través de los correspondientes formatos en los que se piden datos de identidad, que tipo de servicio y/o trámite se solicita, así como todos aquellos documentos que le acrediten para tal solicitud, lo mismo sucede para el caso en que los servidores públicos son los que deben cumplir con sus obligaciones ante la Administración Pública, también deben llenar formatos y cumplir con ciertas disposiciones para poder dar respuesta al ciudadano.

En ambos casos los procesos deben pasar por una etapa de identificación tanto del ciudadano como del servidor público, así mismo debe existir un medio para poder probar posteriormente por parte del ciudadano que entregó la documentación requerida, y para que el funcionario pruebe que dio respuesta a dicha petición, en términos generales deben existir disposiciones que regulen las peticiones de los ciudadanos a la autoridad. En este sentido el Distrito Federal emite disposiciones para regular los actos y procedimientos de la Administración Pública, dichas disposiciones se encuentran en la Ley de Procedimiento Administrativo para el Distrito Federal, que señala en su artículo 1º : “Las disposiciones de la presente Ley son de orden e interés públicos y tienen por objeto regular los actos y procedimientos de la Administración Pública del Distrito Federal...”

Derivado de lo anterior es importante revisar la legitimidad de las solicitudes de trámites y servicios por medios electrónicos, específicamente por Internet.

Primeramente revisemos en que consiste solicitar un trámite o servicio por Internet: Cuando el interesado<sup>3</sup> ingresa a la página Web de la Dependencia o Unidad Administrativa para solicitar un trámite o servicio, lo hace a través de una computadora, acto seguido debe ingresar la información necesaria para cubrir con los requisitos contemplados para dicho trámite o servicio, al momento de realizar el ingreso el ciudadano debe proporcionar información que demuestre su personalidad, es decir, que demuestre que efectivamente el interesado es quien dice ser, estaríamos hablando que deberá ingresar una firma electrónica, después debe obtener un número de folio del registro del trámite o servicio y adicionalmente debe obtener un acuse con el cual pueda comprobar el ingreso del trámite o servicio, para esto último estaríamos hablando que el acuse debe ser electrónico, puede ser una clave o archivo que certifique el ingreso del trámite y/o servicio; ahora bien una vez recibida la solicitud el servidor público debe dar respuesta al ciudadano, es decir, notificar al ciudadano de la resolución a su petición, ya sea favorable o no.

Derivado de lo anterior vemos dos conceptos interesantes: firma electrónica y acuse electrónico, el primero se refiere a poder firmar a través de una computadora mediante algún mecanismo que de certeza de que la persona que dice ser a través de Internet es efectivamente ésta; el segundo concepto se refiere a la notificación que se le daría al ciudadano, de que efectivamente está registrada la solicitud, normalmente estos acuses

---

<sup>3</sup> Según la Ley de Procedimiento Administrativo para el Distrito Federal en su Artículo 2º: “Interesado: Particular que tiene un interés legítimo respecto de un acto o procedimiento administrativo, por ostentar un derecho legalmente tutelado”.

son generados automáticamente a través del sistema de cómputo, esta notificación puede ser a través de una página Web o mediante un correo electrónico; y entonces aquí vemos que aparece un tercer concepto: el correo electrónico, el cual se refiere a una dirección de Internet mediante la cuál una persona puede enviar y/o recibir mensajes de datos, semejante en su funcionamiento al correo convencional, con la diferencia principal de que en el correo electrónico se envía y/o recibe a través de una computadora, adicionalmente en el correo convencional se pueden firmar los documentos que se envían con una firma autógrafa y en el correo electrónico se firman con una firma electrónica; la diferencia importante radica en que la firma autógrafa se realiza de puño y letra del signatario, situación que no se da en la firma electrónica.

Después de describir brevemente el procedimiento que debiera darse para poder ingresar trámites y servicios a través de Internet, pasaremos a revisar las disposiciones que contempla la Ley de Procedimiento Administrativo del Distrito Federal.

El solicitar un trámite o servicio por Internet primeramente sería improcedente ya que como lo menciona el artículo 46 de la Ley de Procedimiento Administrativo del Distrito Federal (LPADF), éstas deben presentarse en las unidades administrativas autorizadas: "Las promociones deberán presentarse en las unidades receptoras autorizadas para tales efectos por la dependencia o entidad; las subsecuentes promociones dentro del procedimiento administrativo podrán presentarse en las oficinas de correos, salvo en el caso de los escritos iniciales los cuales deberán presentarse precisamente en las oficinas administrativas correspondientes", o en su caso por correo certificado como lo indica el artículo 48 de la LPADF: "Los escritos que la Administración Pública del Distrito Federal reciba por vía de correo certificado con acuse de recibo, se considerarán presentados en la fecha que los ingrese la autoridad competente..."

Ahora bien se entiende que las unidades receptoras autorizadas para recibir los trámites y servicios que se regulen por la LPADF deben dar a conocer su domicilio publicándolo en la Gaceta Oficial del Distrito Federal (GODF) como lo indica el artículo octavo transitorio de la LPADF, en este sentido en caso de que opere el ingreso de trámites y servicios por Internet, se tendrá que publicar en la GODF la dirección de Internet.: "Los domicilios de las dependencias y entidades competentes para conocer de los trámites administrativos que estén regulados por esta Ley, deberán ser publicados en la Gaceta Oficial del Distrito Federal, dentro de los 180 días naturales posteriores a la entrada en vigor de la presente ley".

Así mismo el hecho de poder ingresar trámites y servicios por Internet implica la cuestión de la definición de los días de atención de las actuaciones y diligencias, como lo indica el artículo 71 de la LPADF: "Las actuaciones y diligencias previstas en esta Ley se practicarán en días y horas hábiles. Para los efectos de esta Ley se considerarán días inhábiles: Los sábados, domingos, 1º de enero, 5 de febrero, 21 de marzo, 1º de mayo, 16 de septiembre, 20 de noviembre, 1º de diciembre de cada seis años, cuando corresponda a la transmisión del Poder Ejecutivo Federal; y el 25 de diciembre, así como los días en que tengan vacaciones generales las autoridades competentes o aquéllos en que se suspendan las labores, los que se harán del conocimiento público mediante acuerdo del titular de la dependencia o entidad respectiva, que se publicará en la Gaceta Oficial del Distrito Federal", razón por la cual debe contemplarse que los sitios en Internet están normalmente disponibles todas las

horas de todos los días del año, lo que conlleva a la necesidad que en caso de operar el ingreso de trámites y servicios por Internet se contemple dicha disposición.

Continuando con las implicaciones que tendría el ingresar trámites y servicios por Internet, se encuentra el inconveniente de la firma electrónica, ya que el artículo 33 de la LPADF hace mención de firma autógrafa o firma electrónica, pero en ningún momento hace una definición de lo que se debe entender por firma electrónica siendo entonces discrecional y ambiguo. Así mismo el artículo 44 de la LPADF nos señala que “Las promociones deberán hacerse por escrito. ...” pero no se indica que por escrito se acepta la emisión de datos por medios electrónicos.

Ahora bien en lo que respecta al acuse electrónico, que debe tener el ciudadano para poder comprobar el ingreso de su trámite o servicio ante la autoridad, no se encuentra claramente definido en la Ley de Procedimiento Administrativo del Distrito Federal, ya que menciona en su artículo 6º fracción VI que solo se considerarán válidos los actos administrativos que indiquen la autoridad de que emana y que contenga la firma autógrafa o electrónica del servidor público, pero sigue habiendo un vacío en cuanto a la definición y forma de operar de la firma electrónica.

Es importante considerar que el ingreso de trámites y servicios por Internet como ya lo hemos visto requiere de formalidades adicionales como son la firma electrónica, en este sentido la LPADF en su artículo 35 menciona que la autoridad no puede exigir mayores formalidades que los expresamente establecidos en los ordenamientos jurídicos y en el Manual: “La Administración Pública del Distrito Federal en los procedimientos y trámites respectivos, no podrá exigir mayores formalidades y requisitos que los expresamente establecidos en los ordenamientos jurídicos de cada materia y en el Manual...”; entendiéndose por éste último el Manual de Trámites y Servicios al Público.<sup>4</sup>

La finalidad del análisis de las implicaciones en las solicitudes de trámites y servicios por Internet, es evitar incumplir con las normas y disposiciones legales aplicables, toda vez que se debe buscar que todos los actos de la Administración Pública se realicen conforme a Derecho, es por ello que una vez terminado dicho análisis se propone que para el registro de solicitudes de servicios vía Internet basándonos en los principios del Artículo 5 de la LPADF “...simplificación, agilidad, información, precisión, legalidad, transparencia, imparcialidad y buena fe”, sólo se tomen en cuenta aquellos servicios que cumplan con las siguientes características:

- No generen el pago de derechos.
- Que no requieran muchas formalidades (Presentación de documentos).
- Que sean de asesoría y orientación.
- Satisfagan necesidades colectivas y de interés general.
- Que se deban cumplir sin la petición o consentimiento de los gobernados.

---

<sup>4</sup> Según la Ley de Procedimiento Administrativo para el Distrito Federal en su Artículo 2º: “Manual: Manual de Trámites y Servicios al Público del Distrito Federal, que contiene las características de diversos trámites y procedimientos, de acuerdo a los requisitos y plazos que establecen las Leyes y Reglamentos aplicables...”



### 1.3 SERVICIOS DEL MANUAL DE TRÁMITES Y SERVICIOS AL PÚBLICO

El Manual de Trámites y Servicios al Público (MTYSP) es un documento normativo y de difusión que compila diversos trámites y servicios que se gestionan ante la Administración Pública del Distrito Federal. El manual se encuentra dividido en 2 secciones: la de trámites y la de servicios, la sección de servicios (que es la de interés de este trabajo) está organizada por categorías que se denominan materias y cada materia contiene una serie de servicios. Las materias de servicios del Manual se muestran en la Tabla 1.

Agraria	Salud
Agua Potable y Servicios Hidráulicos	Servicios Legales y Archivo de Notarias
Asesoría Telefónica	Servicios Urbanos
Asesoría y Asistencia Social	Trabajo
Industria	Turismo
Medio Ambiente	Uso de Suelo
Registro Público	Vehículos Automotores Pasajeros

Tabla 1

Las características de cada servicio están contenidas en las denominadas Cédulas, que no son otra cosa que un formato con la siguiente información:

1. Nombre del servicio.
2. Tiempo de respuesta.
3. Descripción del servicio.
4. Requisitos para obtener el servicio.
5. Beneficios para el usuario.
6. Fundamento Jurídico-Administrativo.
7. Observaciones sobre el servicio.
8. Oficinas donde se solicita el servicio.
9. Costo.
10. Lugares donde se efectúa el pago.
11. Nota Importante (Reportar anomalías a la Contraloría General).

Dichas cédulas gráficamente tienen el aspecto que se muestra en la Figura 1-1

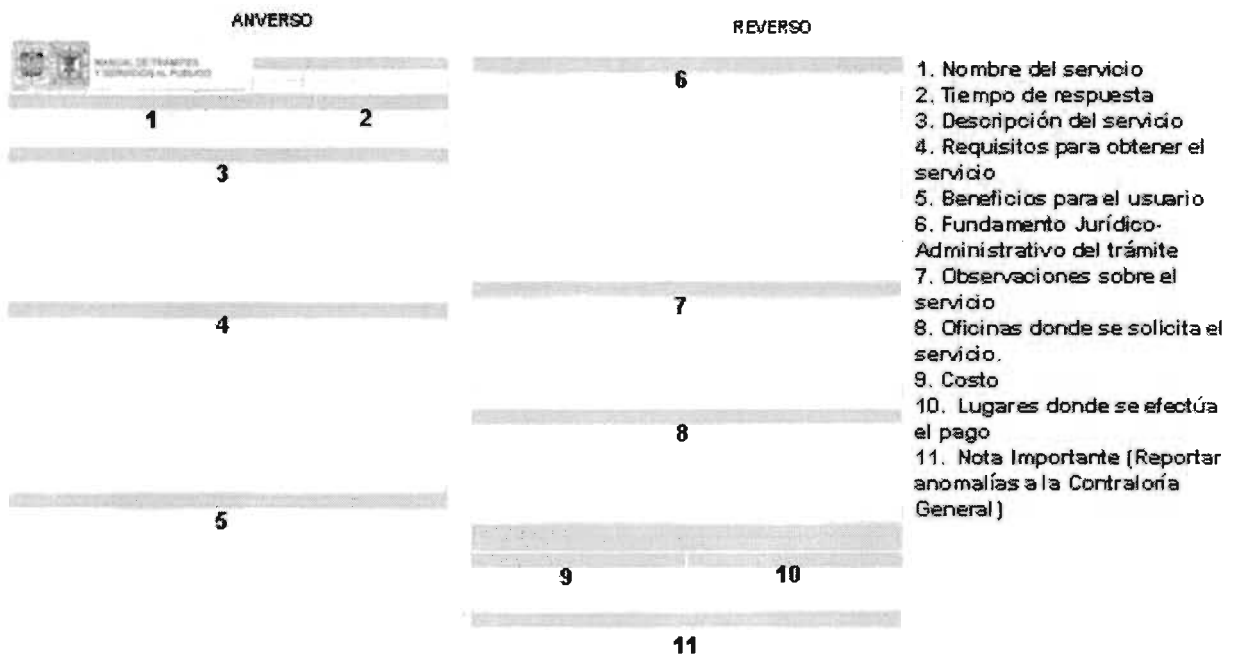


Figura 1-1

La explicación sobre el Manual y sus servicios es importante para el diseño del Programa de Gestión de Servicios a través de Web, la lista de servicios de cada materia de puede consultar en el Anexo A.

Como se mencionó anteriormente, existen inconvenientes para el registro de servicios a través de Web, pero la determinación de los servicios que podrán recibirse mediante el sistema lo hará cada área donde se instale y opere. El sistema será diseñado de tal forma que pueda darse de alta cualquier servicio siempre que se apegue al diseño del Manual de Trámites y Servicios al Público.

## TEMA 2 PLANTEAMIENTO DEL PROBLEMA

### 2.1 ARQUITECTURA DE LOS SISTEMAS DE INFORMACIÓN

La arquitectura de un sistema de información es la forma en que se proyecta y construye él mismo, esta forma de elaboración nos dice las partes que constituyen al sistema y el modo en que éstas se relacionan e interactúan entre sí.

Todo sistema de información tiene código de presentación, de reglas y de manejo de datos. La arquitectura de un sistema dependerá de cómo esté distribuido este código, entendiendo por código una serie de instrucciones que nos permiten darle órdenes a la computadora para que ejecute ciertas operaciones y procesos.

Antes de disponerse a programar y escribir código es importante determinar cuál es la mejor arquitectura para nuestro sistema, lo anterior nos lleva a realizar consideraciones importantes tales como: a) satisfacer las necesidades presentes, b) preparar el sistema para posibles cambios y c) poder implementar mejoras. Es decir, nuestro sistema debe ser flexible.

#### 2.1.1 ARQUITECTURA EN DOS CAPAS

La arquitectura en 2 capas está constituida de las siguientes capas:

Capa 1: Aquí básicamente se encuentran los datos del sistema y en la práctica se refiere a la base de datos.

Capa 2: En esta capa se encuentra los procesos, algoritmos, reglas, procedimientos y restricciones aplicables al usuario del sistema, todo lo anterior se conoce como la lógica de negocios.

También se encuentra en ésta capa la interfaz de usuario que básicamente está formada por ventanas, menús, cuadros de diálogo, botones, reportes y todos los controles necesarios para que el usuario interactúe con el sistema.

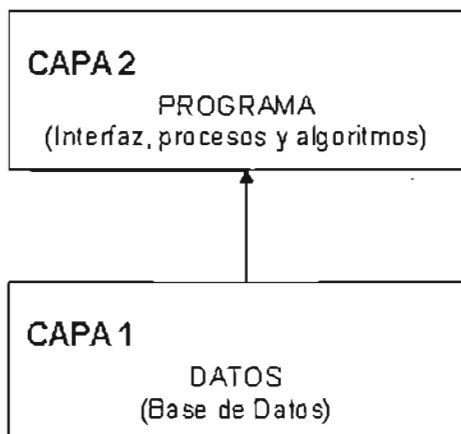


Figura 2-1

Es decir, las bases de datos se encuentran en archivos de texto y/o manejadores de bases de datos, y los procesos, algoritmos e interfaz se encuentran en otro archivo.

La tecnología en 2 capas presenta los siguientes problemas:

Cliente pesado, ya que realiza una gran parte del procesamiento de la aplicación localmente.

El mantenimiento es difícil y costoso en cada cliente, debido a que la revisión de los archivos que contienen la interfaz y la lógica de negocios se vuelve complicada por el extenso número de líneas de código y se complica más aún si dicho código no se encuentra documentado.

Difícil manejo de la Interfaz debido a que para cambiar la interfaz es necesario cambiar también la lógica de negocios.

Difícil migración ya que sería necesario cambiar prácticamente todo: la forma en que se conecta uno a los datos, la interfaz de usuario y la lógica de negocios.

Es cerrado, ya que en caso de aparecer nuevas tecnologías de información es complicado diseñar nuevamente la interfaz para utilizarse con las antes mencionadas tecnologías, de lo anterior podemos poner como ejemplo Internet, cuando necesitemos hacer que el sistema funcione en Internet será necesario diseñar una nueva interfaz para ésta, pero en este nuevo diseño también deberá estar la lógica de negocios, luego entonces cuando necesitemos modificar la lógica de negocios deberemos modificarla en ambas interfaces, lo anterior puede crear problemas debido a que es muy grande la probabilidad de que la lógica de negocios de cada interfaz sea distinta.

La aplicación queda sobrecargada de información de bajo nivel si hay que extraer los datos de varias bases de datos, posiblemente con estructuras diferentes.

Genera mucho tráfico en la Red en que se ejecute la aplicación.

Posibilidad de clientes desfasados, toda vez que el mantenimiento de los mismos es complicado.

En términos generales esta arquitectura suele ser costosa de mantener, difícil de escalar y pesada de depurar. Muchas soluciones actualmente utilizan esta arquitectura y sobre todo aquellas que son cliente/servidor no Internet. Poner reglas de negocio ligados a un interfaz es una mala idea ya que fuerza a que cada cambio en la aplicación nos lleve a ir usuario por usuario cambiándole la aplicación.

### **2.1.2 ARQUITECTURA EN TRES CAPAS**

Derivado de los problemas antes mencionados a parece la tecnología de 3 capas, la cual es una filosofía de trabajo compuesta de las siguientes capas:

Capa de datos .

Capa lógica de negocios .

Capa de Interfaz de usuario.

Es importante mencionar que esta filosofía de trabajo se encuentra directamente relacionada con la arquitectura cliente-servidor.

1.- Capa de datos: Es ésta encontramos todo lo relacionado a la administración, acceso, almacenamiento, actualización, consulta, y recuperación de datos contenidos en el sistema. Típicamente la capa esta conformada por la Base de Datos compuesta por sus respectivas tablas y registros que son administrados por un Manejador de Bases de Datos (DBMS por sus siglas en inglés DataBase Management System) y siempre se localizará en el servidor.

Ésta capa sólo se comunica con la Capa Lógica de Negocios para el manejo de los datos que puede incluir el ingreso y consulta de información.

La capa de datos puede estar en el mismo servidor que las de lógica de negocio y presentación, o en un servidor independiente (incluso puede consistir en un cluster de servidores).

2.- Capa lógica de negocios: Aquí encontramos todos los procesos, algoritmos, procedimientos, reglas, lógica, validaciones, cálculos y restricciones aplicables que requiere el sistema de información para tener una funcionalidad específica. Ejemplos de estas restricciones y/o reglas pueden ser:

En un banco: abrir y cerrar cuentas, disposición de efectivo al día, etc.

En un negocio. Aplicación del IVA, política de descuentos, disponibilidad en almacén, etc.

La capa entra en funcionamiento al recibir peticiones del usuario a través de la Capa de Interfaz de Usuario, una vez procesada dicha petición ésta capa se comunicará con la Capa de Datos para insertar, extraer y/o actualizar información y por último enviará a la Capa de Interfaz de usuario el resultado de la petición.

Puede estar en el servidor o en el cliente, dependerá de las necesidades y naturaleza del sistema; Así mismo nos permite una integración más sencilla y eficaz con sistemas externos.

3.- Capa de interfaz de usuario: Esta capa contiene la interfaz con la cual el usuario interactúa con el sistema. Dicha interfaz contiene propiamente las ventanas, menús, gráficos, cuadros de diálogo, botones, reportes, elementos multimedia en general, modelo de navegación del sistema y los adaptadores para cada modo de acceso (browser, teléfono celular, etc.). Su función principal es la de captura y visualización de datos, que se realiza mediante peticiones que atiende la Capa Lógica de Negocios.

En un sistema basado en Web ésta capa está compuesta por distintas páginas Web que presentan los datos en un navegador (Internet Explorer, Netscape Navigator, Opera, etc) tanto en una PC como en los distintos dispositivos móviles (celulares y palm).

Puede estar en el servidor o en el cliente.

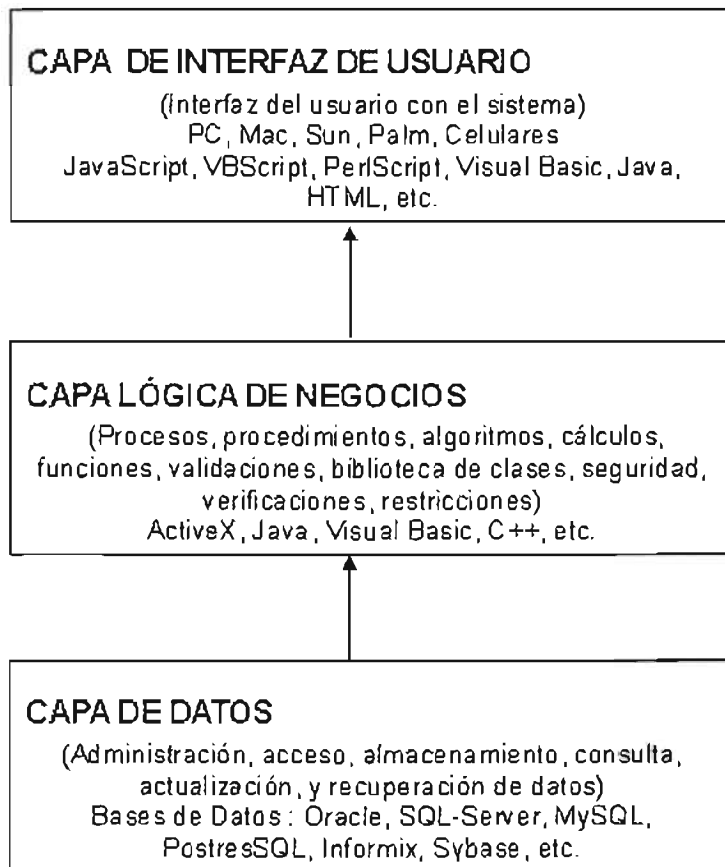


Figura 2-2

Las principales ventajas de esta forma de trabajo en la construcción de sistemas de información son las siguientes:

Mantenimiento fácil y barato.

Se separan los datos de la lógica de negocios.

Se tiene un manejo más fácil de la interfaz.

Flexibilidad.

Reutilización y posibilidad de compartir capas, ya que existe la posibilidad de crear diferentes interfaces para la misma lógica de negocio.

Dependencias se limitan a intra-capa, esto proporciona libertad de los elementos importantes del sistema, ya que se puede modificar una capa sin necesidad de modificar todas las demás capas del sistema y en un momento determinado los cambios en las demás capas es mínimo.

Cliente ligero, ya que solo realiza las tareas de presentación, esto permite que la computadora del cliente sea menos potente.

Arquitectura más escalable, ya que se pueden integrar aplicaciones que accedan a las mismas bases de datos de una forma sencilla.

Permite una variedad de entornos distintos de Sistemas Operativos en el cliente (Como es el caso de los exploradores para Internet)

Reduce la complejidad de administración, instalación, configuración y mantenimiento del software en el cliente.

Separar las reglas de negocio de los interfaces especialmente en entornos multiplataforma permite que las reglas se cambien con un mínimo impacto sobre los usuarios de las aplicaciones.

Es fácil construir nuevas aplicaciones desde los componentes instalados si las reglas del negocio están en unos servidores de aplicaciones más que en en cada aplicación.

Así mismo nos permite una integración más sencilla y eficaz con sistemas externos.

Las aplicaciones en tres capas típicamente tienen mayor capacidad de crecimiento y son más sencillas de mantener, dada su naturaleza altamente modular, aunque cabe aclarar que este tipo de arquitectura es la más compleja.

## 2.2 APLICACIONES DE BASES DE DATOS EN WEB

### 2.2.1 ARQUITECTURA

En el nivel más básico, las aplicaciones Web trabajan en una arquitectura Cliente/Servidor, simplemente se necesita de un servidor central y una aplicación cliente, a diferencia de las aplicaciones tradicionales que no requieren de un servidor para su funcionamiento como sería el caso de las primeras bases de datos como FoxPro, Dbase, Clipper o incluso Access.

La arquitectura de las aplicaciones Web funciona de la forma que se observa en la figura 2-3, un cliente que realiza peticiones al servidor quien las recibe y procesa para posteriormente enviar las respuestas al cliente.

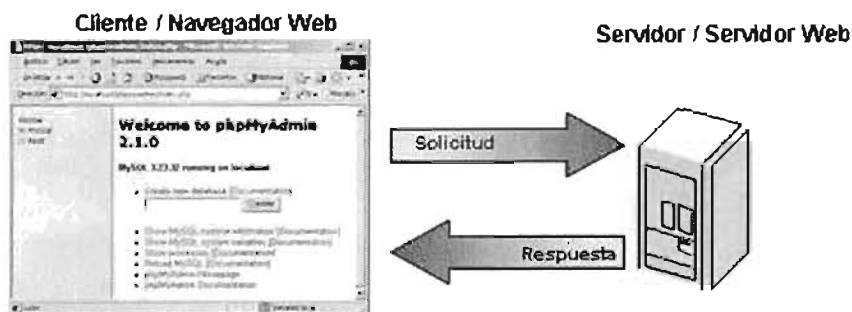


Figura 2-3

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 2-4.

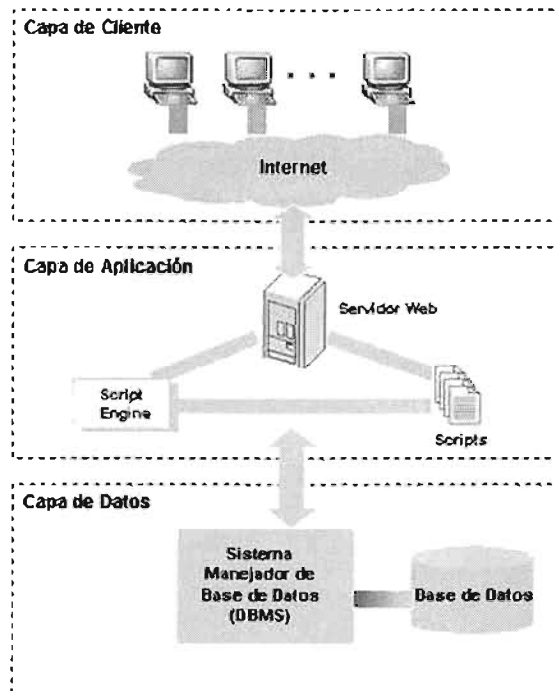


Figura 2-4

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 como respuesta una página igual.

### 2.2.2 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 pero las características más comunes de todos los navegadores son:

Todos los navegadores son clientes de HTTP para enviar solicitudes y recibir las respuestas del servidor Web.

Todos los navegadores interpretan páginas hechas con HTML (Hypertext Markup Language), que pueden contener texto, encabezados, imágenes y enlaces a otros recursos.

Los navegadores pueden mostrar imágenes, reproducir audio y video, y mostrar otro tipo de objetos.

Muchos navegadores pueden ejecutar JavaScript embebido en las páginas HTML.

Dependiendo del navegador seleccionando se pueden ejecutar componentes desarrollados en los lenguajes de programación Java o ActiveX.

Varios navegadores pueden aplicar CSS (Cascading Style Sheets , Hojas de Estilo en Cascada) para tener un mayor control de las páginas en HTML.



### 2.2.3 CAPA DE APLICACIÓN

Esta capa contiene la lógica de la aplicación para integrar la capa de cliente con la capa de datos. 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 2-4.

#### 2.2.3.1 SERVIDORES WEB

Un servidor Web no es otra cosa que un servidor de HTTP, es decir un programa que ofrece un servicio a través de una red para recibir peticiones y enviar respuestas a través del protocolo HTTP, recibe peticiones por el protocolo HTTP que fueron enviadas por un navegador Web y envía las respuestas al navegador Web mediante el mismo protocolo. El servidor Web al recibir peticiones de consulta e inserción de datos en una base de datos mediante los script hechos en un lenguaje de programación, lo que hace es interpretar esos scripts comunicándose al intérprete del lenguaje quien le regresa el resultado de la operación sobre la base de datos, es entonces cuando el Servidor Web devuelve el resultado al cliente Web (normalmente un navegador Web).

Existen varios servidores Web, pero los que dominan el mercado son: Apache e Internet Information Server (IIS).

#### 2.2.3.2 LENGUAJES DE PROGRAMACIÓN SCRIPT E INTÉRPRETES

Los lenguajes de programación script no son otra cosa que un conjunto de expresiones y sentencias que son analizadas por un Intérprete para realizar diferentes funciones que van desde mostrar texto en un navegador Web hasta la consulta e inserción de información en una bases de datos. Entre los más populares para consultar e insertar información en bases de datos encontramos ASP, PHP, Perl y ColdFusión. La característica fundamental de todos ellos es la capacidad de embeberse en las páginas HTML.

Los intérpretes de los lenguajes script no son otra cosa que un módulo que funciona en conjunto con el servidor Web, es decir, se trata de un programa encargado de interpretar las peticiones realizadas al servidor Web por un cliente mediante una página HTML que tiene incrustado el lenguaje script, una vez que interpreta el script donde le solicitan consultas e inserciones de datos en una base de datos, lo que hace es traducir a consultas para inserción y recuperación de datos que envía al manejador de la base de datos quien le devolverá la información solicitada misma que será convertida por el intérprete en una página HTML que enviará al cliente Web a través del servidor Web como respuesta a su petición.

### 2.2.4 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 manejo de datos incluye típicamente el almacenamiento y recuperación de la información, manejo de las actualizaciones, manejo 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 está compuesta de dos elementos: Un Manejador de Base de Datos y un Lenguaje de Gestión con la Base de Datos.

#### 2.2.4.1 MANEJADOR DE BASES DE DATOS

Un Manejador de Bases de Datos o DBMS por sus siglas en inglés (Database Management System), es un programa que sirve para controlar el almacenamiento, recuperación y manejo de la información, dicha información está contenida en Bases de Datos que no son otra cosa que un conjunto de datos relacionados. Los elementos principales que conforman una base de datos son las entidades y sus relaciones entre sí que pueden ser de distinta forma. Las entidades están expresadas en tablas las cuales están compuestas de una serie de campos que forman registros. Como ejemplo de lo anterior podemos tener una tabla (entidad) llamada colonia, la cual está compuesta de los campos: clave de la colonia, nombre de la colonia y código postal, un registro será un reglón que contenga los datos de los 3 campos anteriormente mencionados. Ahora bien, como ejemplo de las relaciones podemos tener una tabla llamada calle, la cual contendrá los campos: clave de la calle, nombre de la calle y clave de la colonia a la que pertenece, la relación será de uno a muchos, es decir, una colonia puede tener muchas calles.

Entre los más populares manejadores de bases de datos tenemos: MS-SQL Server, Oracle, Informix, DB2, PostgreSQL y MySQL, es importante mencionar que la mayoría de ellos tienen una arquitectura típica como se muestra en la Figura 2-5.

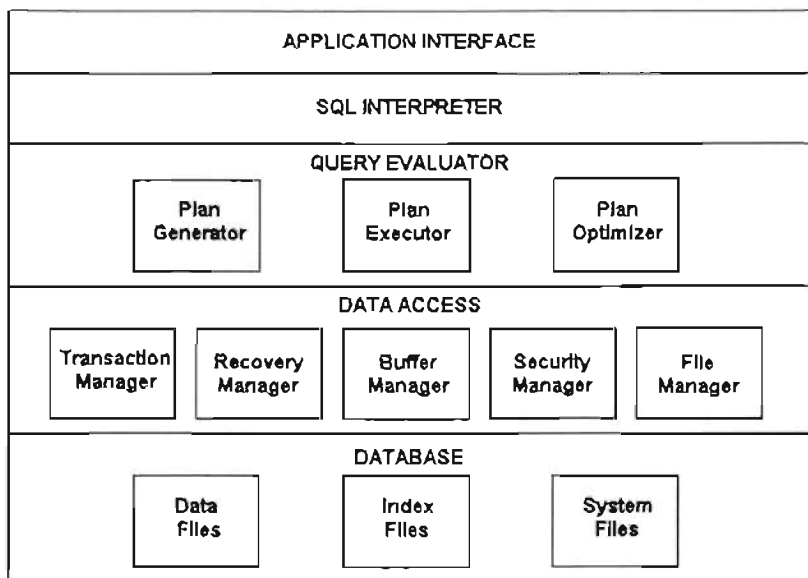


Figura 2-5

Un Manejador de Bases de Datos tiene los siguientes componentes:

**Application Interface:**

La Interfaz de Aplicación son librerías para comunicarse con la base de datos. Muchos de los DBMS tienen una simple línea de comandos.

**SQL Interpreter:**

El Intérprete de SQL revisa la sintaxis de las consultas realizadas y las traslada a un lenguaje interno para su ejecución dentro del DBMS.

**Query Evaluator:**

El Evaluador de las Consultas genera diferentes evaluaciones de las consultas y considerando las características y propiedades de la base de datos traslada las consultas a acciones de bajo nivel para su ejecución.

**Data Access:**

El Acceso a Datos es un módulo que maneja los accesos para el almacenamiento de la información en el disco de la computadora, incluye los manejadores de transacciones, recuperación, buffer, seguridad y archivos.

**Database:**

La Base de Datos son los datos físicos que se almacenan en archivos y que contiene archivos de indexación (ordenamiento) para un acceso rápido a la información.

#### 2.2.4.2 LENGUAJE DE GESTIÓN CON LA BASE DE DATOS

Con la aparición de los primeros manejadores de bases de datos surgió la necesidad de almacenar, organizar y consultar la información de cada una de las bases de datos creadas por distintos productos, para ello se utilizaban lenguajes propios de cada manejador (Lenguaje de gestión con la base de datos), pero esto representaba un problema, ya que para gestionar cada una de las bases de datos se requerían especialistas en cada producto.

En junio de 1970 el Dr. E.F. Cood, investigador de IBM, "...publicó un artículo titulado <Un modelo relacional de datos para grandes bancos de datos compartidos> que perfilaba una teoría matemática de cómo almacenar y manipular datos utilizando una estructura tabular. Las bases de datos y SQL tienen sus orígenes en este artículo...".

SQL es un lenguaje para organizar, gestionar y recuperar datos almacenados en una base de datos. SQL es una abreviación de Structured Query Language (Lenguaje Estructurado de Consultas). SQL funciona con un tipo específico de bases de datos, llamado Base de Datos Relacional.

La importancia de SQL radica básicamente en que después de su aparición en los años 80's se crearon 3 estándares publicados por la American National Standards Institute (ANSI) y la Internacional Standards Organization (ISO); por ello hoy en día la mayoría de los Manejadores de Bases de Datos utilizan SQL como lenguaje de gestión con la base de datos, entre ellos: SQL Server, Oracle, Sybase, Informix, DB2, PostgreSQL y MySQL.

Básicamente SQL está compuesto de 2 partes:

**Data Definition Language:**

Que es un conjunto de comandos SQL para crear y borrar bases de datos, agregar y eliminar tablas, crear índices y modificar cada uno de éstos. Generalmente se utiliza DDL durante la construcción de la base de datos

**Data Manipulation Language:**

Conjunto de comandos SQL para trabajar con una base de datos, que nos permite buscar, insertar y borrar información. DML se utiliza para interactuar con la base de datos durante el uso normal.

## 2.3 INVESTIGACIÓN DEL PROBLEMA

### 2.3.1 DEFINICIÓN DEL PROBLEMA

Actualmente el Gobierno del Distrito Federal no cuenta con un sistema de recepción de solicitudes de servicios que responda a las nuevas necesidades tecnológicas y sociales de los ciudadanos, cada una de las Dependencias que conforman la Administración Pública del Distrito Federal que tienen atribuciones y funciones para la recepción de servicios (mismos que se encuentran marcados por el Manual de Trámites y Servicios al Público) reciben y manejan las solicitudes de muy distintas maneras, que varían desde el procedimiento, forma de orientar al ciudadano, información que se proporciona, forma de recepción y asignación de folios, tipo de seguimiento, número de requisitos que se solicitan, cantidad de copias requeridas, tipo de tecnología utilizada para almacenar y procesar dicha información, éstas son varias de las situaciones que impiden que se pueda dentro del Gobierno reportar de forma segura y confiable el número de servicios que se solicitan y la cantidad de los que se atienden. Cabe aclarar que actualmente hay diversos puntos donde se pueden recibir solicitudes de servicios publicados en el Manual de Trámites y Servicios al Público. Es importante mencionar que el Manual de Trámites es un esfuerzo de la Administración Pública por compilar de forma ordenada y uniforme, una serie de servicios que se prestan en las diversas Dependencias que la conforman; es por ello que el objetivo del presente trabajo es diseñar un sistema en apego a las características de dicho Manual.

Ahora bien, una vez explicado en términos generales la problemática que se desea resolver es importante enumerar los problemas que se tienen derivado de ésta falta de un sistema integral para el registro de solicitudes de servicios, entre ellos tenemos:

Una gran cantidad de programas desarrollados en diversos lenguajes de programación para distintas plataformas.

Diversos procedimientos de recepción de solicitudes.

Dependencias de sistemas operativos para ejecutar las aplicaciones.

El uso de licencias de manejadores de bases de datos, servidores Web, sistemas operativos y clientes de aplicaciones.

Aplicaciones con clientes que requieren una gran cantidad de recursos de hardware para poder operar adecuadamente.

Aplicaciones que se encuentran integradas en sus distintas capas: datos, interfaz y reglas del negocio, lo que dificulta su actualización y mantenimiento.

Dificultad para poder compartir información y para poder generar reportes globales de las solicitudes de servicios.

No hay apego a la Normatividad para la recepción de solicitudes, en éste caso hablamos de apegarse al Manual de Trámites y Servicios apoyándose de los procedimientos establecidos en la Ley de Procedimiento Administrativo del Distrito Federal.

No hay forma de que el ciudadano pueda solicitar vía electrónica cualquier tipo de servicio publicado en el Manual.

No hay un uso eficiente de los recursos de software y hardware, toda vez que se gasta por cada sistema que se diseña para la gestión de servicios.

### 2.3.2 OBJETIVO GENERAL

Se propone un sistema integral para el registro, seguimiento, control y evaluación de la demanda de servicios de la ciudadanía, lo anterior para facilitar la atención de las solicitudes, evaluar la gestión del gobierno y apoyar la toma de decisiones. Se propone un módulo para el registro de las solicitudes de servicios, un módulo para el seguimiento de las solicitudes por parte de las áreas operativas, un módulo de catálogos, un módulo de administración de los usuarios y permisos del sistema, un módulo de reportes que apoye a la toma de decisiones y un módulo de orientación.

Características:

Interfaz amigable

Clasificación de las solicitudes

Generación de las órdenes de trabajo

Generación de reportes

Disponibilidad de la información

Canalización automática a las áreas operativas

Seguimiento detallado de las solicitudes

Información oportuna para la toma de decisiones

Eliminar discrecionalidad

Evitar pérdida y duplicación de la información

Intercambiar información entre distintas áreas de gobierno

Genere un reporte global que evalúe la situación de la gestión en el área donde opere el sistema dando los indicadores de gestión.

### 2.3.3 OBJETIVOS ESPECÍFICOS

Diseñar una aplicación utilizando una arquitectura de 3 capas.

Realizar el análisis y diseño del Programa de Gestión de Servicios.

Utilizar UML para el modelado del sistema.

Diseñar una nueva Base de Datos.

Diseñar y desarrollar una Interfaz amigable y configurable.

La arquitectura de la aplicación será basada en Web.

Utilizar un lenguaje de programación de Aplicaciones Web para el desarrollo del sistema.

Implementar mecanismos que hagan posible el uso del sistema vía Internet e Intranet.

Utilizar Software Libre para el sistema operativo del servidor, servidor Web, base de datos y generador de aplicaciones.

### 2.3.4 ALCANCE Y LIMITES

La Base de Datos se construirá en MySQL.

La aplicación se desarrollará en PHP.

Se utilizará Apache como servidor Web.

El sistema operativo para el servidor será Linux, aunque al utilizar Apache y MySQL la aplicación puede operar en un ambiente Windows.

El sistema incluirá las siguientes características:

Clasificación de las solicitudes por materias, tipo de servicio, área de seguimiento, estado de la solicitud, tipo de captación, ubicación geográfica por colonias y calles y tipo de organización que realiza la solicitud.

Mecanismos de seguridad para el acceso al mismo.

Acceso a cada uno de los módulos y submódulos dependiendo de los permisos.

Sólo abarcará los servicios del Manual de Trámites y Servicios al Público.

Permitirá la captura de catálogos de materias, servicios, geográficos, organizaciones, áreas y tipo de captación.

Permitirá la captura de solicitudes de servicios.

Soportará la captura de solicitudes a través de Internet.

Permitirá la emisión de reportes.

Contendrá una bitácora de accesos.

Contendrá una bitácora de operaciones.

Permitirá el alta de usuarios y la asignación de permisos para cada uno.

En cuanto a los límites del sistema es importante señalar que el seguimiento detallado (turnado y cada una de las acciones) realizado sobre la solicitud no se incluirá en éste proyecto toda vez que ello implica estandarizar mediante mecanismos normativos los procedimientos empleados en cada una de las Dependencias, Órganos Desconcentrados y Unidades Administrativas del Gobierno y a que son distintos para cada una, y es una cuestión de índole normativa que requiere de mecanismos legales dentro del marco del Derecho Administrativo, es por ello que el diseño y desarrollo del sistema se llevará a cabo con los procedimientos ya normados para la clasificación y recepción de solicitudes de servicios, ya que en el Manual de Trámites y Servicios al Público no se detallan los procedimientos para dar seguimiento y atención de manera interna a cada una de los tipos de solicitudes, además de que es importante señalar que dentro de la Administración Pública los Servidores Públicos sólo pueden realizar estrictamente las atribuciones que los Reglamentos y normas les confieren.

### 2.3.5 COSTO DE DESARROLLO

En lo que respecta al costo del desarrollo del Programa de Gestión de Servicios, lo que debemos tomar en cuenta para realizar el cálculo son todos aquellos recursos necesarios para llevar a buen término la construcción del sistema que van desde los recursos de hardware hasta los recursos humanos, para el caso de éste proyecto podemos decir que el costo de hardware y software no se incluye toda vez que el hardware con el que se desarrollo ya se contaba y en el caso de software se utilizó Software Libre<sup>5</sup> gratuito.

Ahora bien en cuanto a los recursos humanos, hora-hombre empleado para la realización del proyecto tenemos que mencionar que se desarrollo por una sola persona

---

<sup>5</sup> El software libre es software que viene con autorización para que cualquiera pueda usarlo, copiarlo y distribuirlo, ya sea literal o con modificaciones, gratis o mediante una gratificación. <http://www.gnu.org>

y que sólo se emplearon 2 horas hombres por semana dentro del horario laborable por un lapso de 30 semanas, calculando que la hora-hombre aproximadamente se paga a 100 pesos, tendríamos un calculo de 6,000 pesos en lo que respecta a los recursos humanos, pero hay que sumar el precio de la capacitación que en este caso fue de 18,000 pesos, por lo que tendríamos un total de 24,000.

### 2.3.6 COSTO DE OPERACIÓN

En lo que se refiere a costos de operación, se deben considerar los recursos necesarios para que el sistema opere, desde recursos materiales hasta humanos, para ello debemos 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 costo del personal necesario para dar soporte al sistema.

Ahora bien, abordando cada uno de los puntos antes mencionados, primeramente tenemos 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 será preciso que exista una red, lo anterior expresado en términos económicos resulta difícil de calcular toda vez que dependerá de cada una de las Dependencias donde se instale el sistema, pero tomando como modelo a las 16 Delegaciones políticas del Distrito Federal, cada una de ellas en sus Centros de Servicio y Atención Ciudadana (CESAC) a través de las Unidades de Atención Ciudadana fueron dotadas de una infraestructura informática que resulta bastante útil para la instalación de la aplicación, ya que cuentan con una red ethernet 10/100, equipos Pentium II a 350 Mhz, 64 MB de RAM con discos duros de 4GB, además de que los servidores proporcionados tienen las siguientes características: Pentium Dual II a 350 Mhz, 128 MB de RAM y 2 discos duros de 9GB en un arreglo RAID.

Y con los datos anteriormente listados podemos decir que para instalar el sistema en lo que respecta a hardware no se realizarán costos adicionales ya que para poder instalar el navegador Web (Netscape o Internet Explorer) en cada uno de los clientes no se requieren grandes recursos y las características del equipo actual (por lo menos en el caso de las Delegaciones) son suficientes; y para poder instalar los servidores de Web y de Bases de Datos (en este caso MySQL), las características del equipo son más que suficientes.

Pasando al tema del costo del software, se requiere básicamente el sistema operativo, el servidor Web, el servidor de bases de datos y un intérprete de lenguaje script para desarrollo en Web, en ese sentido la propuesta de este proyecto se basa en la utilización de Software Libre gratuito y por ello se propone Linux como sistema operativo, Apache como servidor Web, MySQL como servidor 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 dinero para su capacitación ya que el Gobierno del Distrito Federal tiene planeado para el año 2004 impartir una serie cursos de software libre para los servidores públicos, que incluyen los productos antes mencionados, pero si es necesario calcular el costo del sueldo de la persona encargada del sistema y que como mínimo deberá tener un nivel de Enlace y cuyo sueldo oscila alrededor de los 8 mil pesos por mes, aunque esto no debe tomarse como el costo de operación ya que esta persona podrá desempeñar otras actividades adicionales al soporte del sistema.

### 2.3.7 BENEFICIOS

Los beneficios que se esperan del sistema no están relacionados con la reducción del personal en las áreas de atención, de lo que se trata es proporcionar a la ciudadanía un servicio de calidad en tiempo y forma, es decir, que se le atienda la recepción en el menor tiempo posible, que se le de una respuesta expedita y que dicha respuesta sea de calidad, los beneficios del sistema se enlistan a continuación:

Estandarizar la forma de clasificación de los servicios en apego al MTYP.

Estandarizar la forma de orientar al ciudadano en apego al MTYP.

Apego a la Normatividad para la recepción de solicitudes, específicamente al Manual de Trámites y Servicios apoyándose de los procedimientos establecidos en la Ley de Procedimiento Administrativo del Distrito Federal.

Estandarizar la forma de asignación de folios.

Estandarizar el tipo de tecnología utilizada para almacenar y procesar la información.

Orientar de manera precisa a la ciudadanía de los requisitos necesarios para la prestación de servicios.

Evitar discrecionalidad en los requisitos solicitados al ciudadano para la prestación de un servicio.

Ahorro de recursos en el diseño y desarrollo de sistemas propios para la recepción de solicitudes de servicios en cada Dependencia del Gobierno que preste servicios.

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.

Se podrá compartir información entre Dependencias ya que dicha información estará clasificada.

Permitirá la generación de reportes de una manera más sencilla

Clasificación de la información por diversos tipos de rubros como pueden ser: tipo de servicio, materia del servicio, tipo de captación de la solicitud, ubicación geográfica de la solicitud, estado que guarda la solicitud, etc.

Responder a las nuevas necesidades tecnológicas y sociales de la ciudadanía.

Contar con información confiable del número de servicios que se solicitan y la cantidad de los que se atienden.

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 debido a que se desarrollará con Software Libre altamente aceptado en el mundo de la informática.

Permitir que el ciudadano pueda solicitar vía electrónica cualquier tipo de servicio publicado en el Manual.



Uso eficiente de los recursos de software y hardware, toda vez que se evitarán gastos en nuevos sistemas para la gestión de servicios.

Permitirá evaluar la gestión del gobierno y apoyar la toma de decisiones.

Contar con un control de los usuarios que ingresan al sistema y realizan acciones dentro del sistema.

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 desplegadas y botones.

Permitirá de manera fácil la generación de las órdenes de trabajo.

Disponibilidad de la información

Mejorar la atención al ciudadano

Evitar pérdida y duplicación de la información.

Ahorro de papel de las solicitudes vía Internet.

Ofrecer más opciones a los ciudadanos para la solicitud de servicios.

Se contarán con nuevos solicitantes a través de Internet.

## TEMA 3 ANÁLISIS Y DISEÑO

### 3.1 DETERMINACIÓN DE REQUERIMIENTOS

De las necesidades recabadas se obtuvo que básicamente se requiere de un sistema con los siguientes 6 módulos: de registro, de seguimiento, de reportes, de catálogos, de orientación y de administración, y que deberán tener las funciones que se listan en las Tablas 2 y 3.

#### Funciones del Sistema

No	Función
1	Validar acceso de usuario
2	Validar permisos de usuario
3	Salir del sistema
	<b>Funciones del Módulo de Registro</b>
4	Registrar solicitante
5	Modificar datos del solicitante
6	Eliminar solicitante
7	Consultar datos del solicitante
8	Registrar solicitud
9	Modificar datos de la solicitud
10	Eliminar solicitud
11	Consultar datos de la solicitud
12	Generar acuse de respuesta
	<b>Funciones del Módulo de Seguimiento</b>
13	Recepción de solicitud
14	Modificar estado de la solicitud
15	Consultar seguimiento de la solicitud
16	Generar ordenes de trabajo

Tabla 2

## Funciones del Sistema (continuación)

No	Función
	<b>Funciones del Módulo de Reportes</b>
17	Generar reporte
	<b>Funciones del Módulo de Catálogos</b>
18	Agregar un registro a un catálogo
19	Modificar datos de un registro
20	Eliminar un registro
21	Consultar datos de un registro
22	Buscar un registro
23	Imprimir catálogo
	<b>Funciones del Módulo de Administración</b>
24	Configurar el sistema
25	Registrar usuario
26	Modificar datos del usuario
27	Eliminar usuario
28	Consultar datos del usuario
29	Buscar usuario
30	Imprimir catálogo de usuarios
31	Consultar bitácora de acceso
32	Imprimir bitácora de acceso
33	Consultar bitácora de operaciones
34	Imprimir bitácora de operaciones
	<b>Funciones del Módulo de Orientación</b>
35	Consultar cédulas de servicios
36	Imprimir consulta

Tabla 3

### 3.2 SUBSISTEMAS DEL SISTEMA

Módulo de Registro  
Módulo de Seguimiento  
Módulo de Reportes  
Módulo de Catálogos  
Módulo de Administración  
Módulo de Orientación

### 3.3 CASOS DE USO

En los siguientes puntos se presentarán algunos diagramas que forman parte del UML<sup>6</sup>.

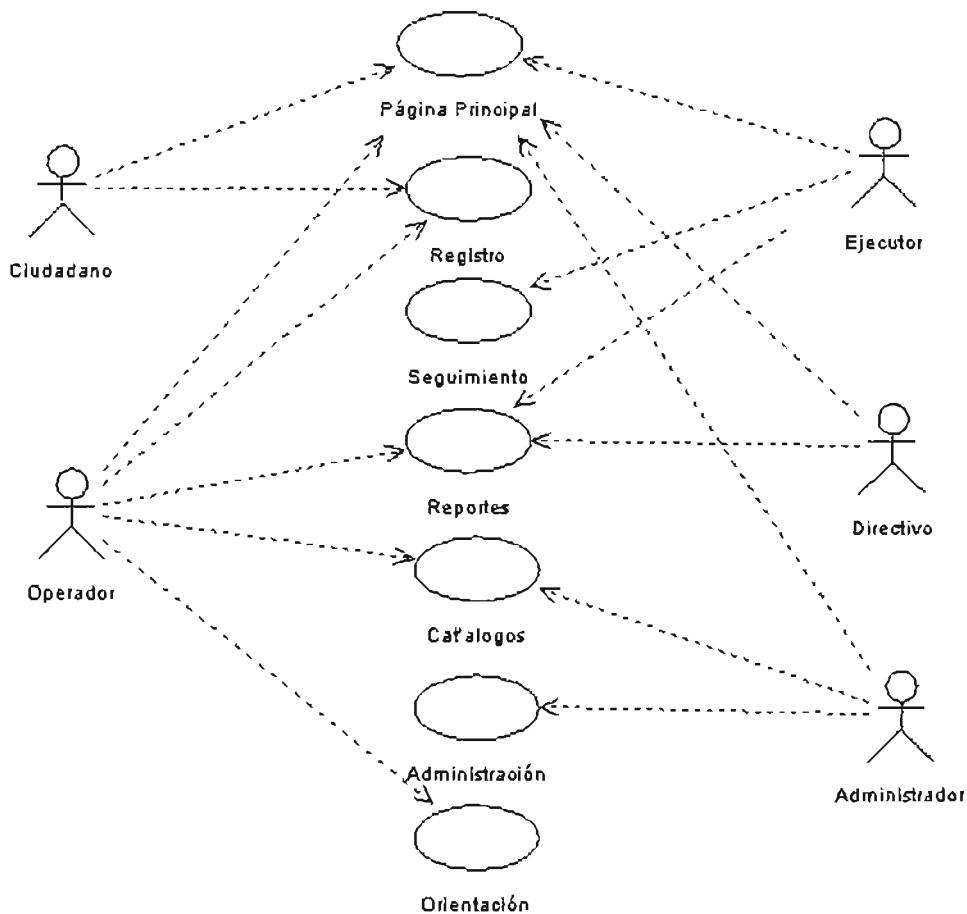


Figura 3-1

<sup>6</sup> El UML es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software a partir de un conjunto de diagramas; No define un proceso de desarrollo específico tan sólo se trata de una notación que permite crear diseños que capturan las ideas de los analistas de sistemas en forma convencional y fácil de comprender para comunicarlas a otras personas.

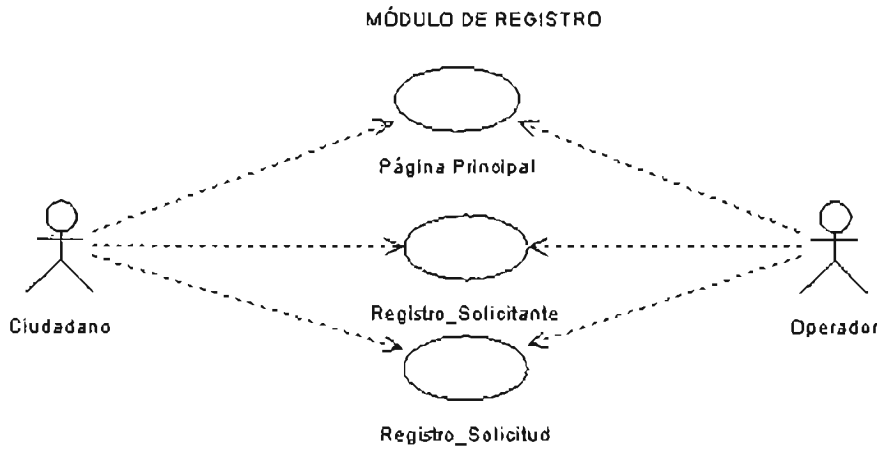


Figura 3-2

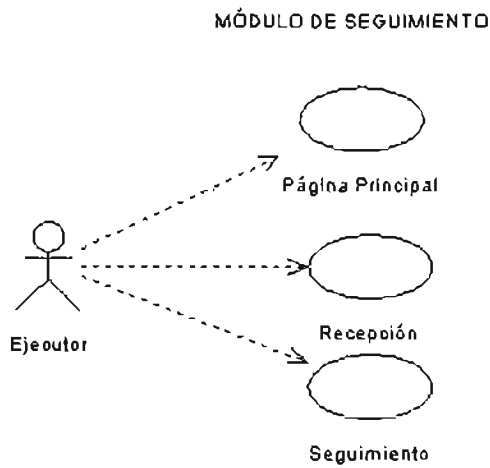


Figura 3-3

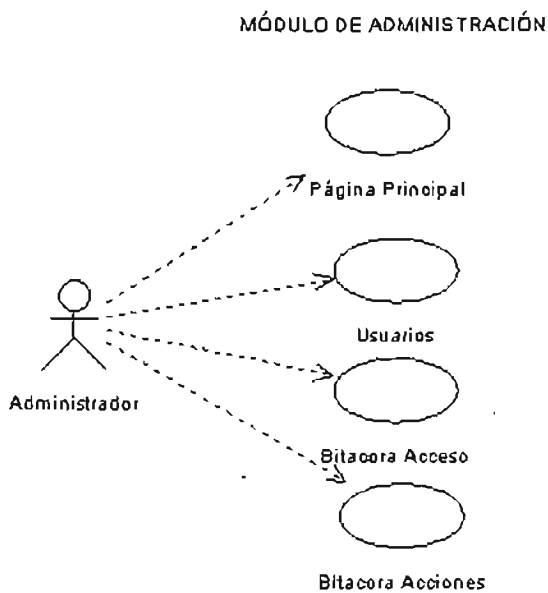


Figura 3-4

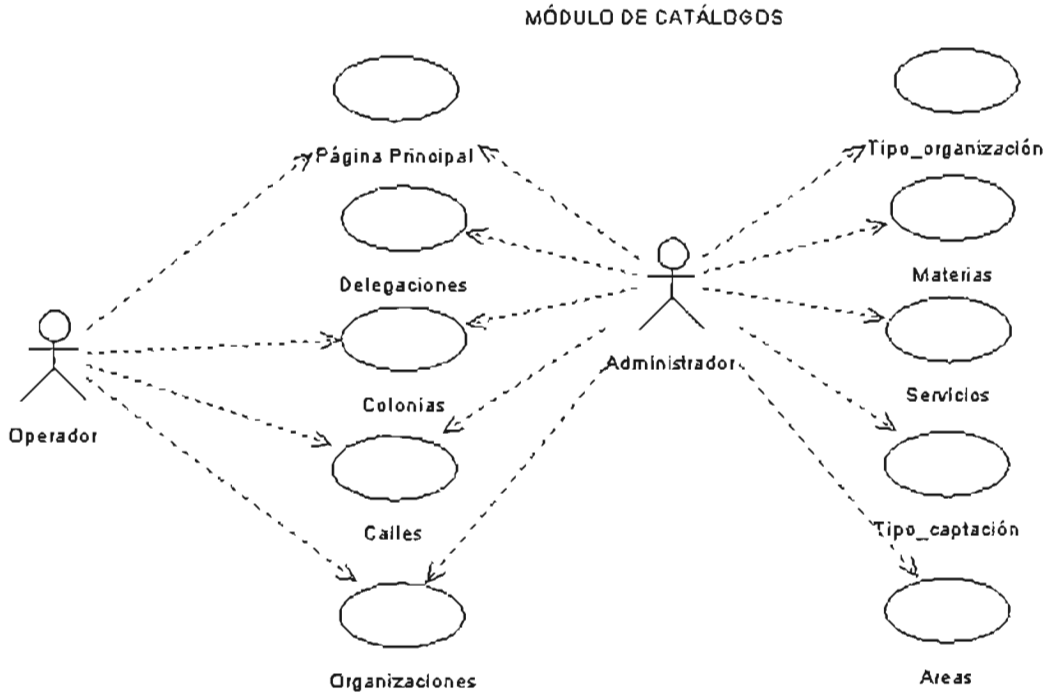


Figura 3-5

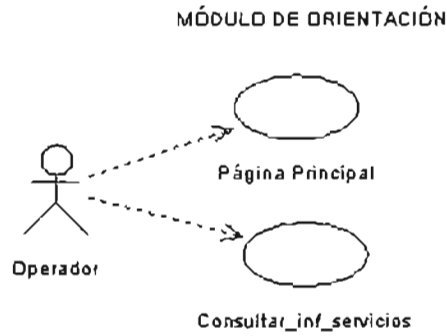


Figura 3-6

<b>Caso de Uso:</b>	Página Principal
<b>Nombre:</b>	Página Principal
<b>Actores:</b>	Ciudadano, Operador, Ejecutor, Directivo y Administrador
<b>Propósito:</b>	Entrar a la aplicación y utilizar las herramientas del sistema.
<b>Tipo:</b>	Primario
<b>Descripción:</b>	Cuando el usuario entra a ésta página podrá ingresar sus datos para acceder al sistema.
<b>Referencias:</b>	1 y 2

**Curso de Eventos:** Página Principal

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario accede a la página principal del sistema	2	Muestra la página
3	El usuario ingresa su usuario y contraseña	4	Ingresa al sistema mostrando las opciones a las que tiene acceso dependiendo de sus permisos

<b>Caso de Uso:</b>	Registro
<b>Nombre:</b>	Registro
<b>Actores:</b>	Ciudadano, Operador,
<b>Propósito:</b>	Registrar solicitantes y solicitudes, y ver estado de una solicitud
<b>Tipo:</b>	Primario
<b>Descripción:</b>	Cuando se entra a la aplicación se validan los permisos del usuario dentro de la aplicación. Podrán registrar solicitantes y solicitudes a través de formularios.
<b>Referencias:</b>	4,5,6,7,8,9,10,11,12

**Curso de Eventos:** Registro

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra página con las opciones: solicitante y solicitud
3	El usuario selecciona la opción solicitante ó solicitud	4	Muestra una ventana con los registros del capturados y las opciones: agregar, eliminar, modificar y consultar
5	El usuario selecciona la opción Agregar	6	Se muestra formulario de captura donde el usuario teclea la información solicitada
7	El usuario captura datos y selecciona el botón guardar	8	Valida información y devuelve un mensaje de confirmación en caso contrario envía mensaje de error
9	El usuario selecciona la opción Eliminar	10	Se pide confirmación y se valida que no existan registros ligados, se elimina enviando mensaje de notificación ó en su caso se notifica causa de improcedencia de la acción.
11	El usuario selecciona la opción Modificar	12	Se muestra formulario cargado con los datos del registro seleccionado
13	El usuario modifica los datos y selecciona el botón guardar	14	Valida información y guarda los datos. En caso de haber algún error envía mensaje al usuario.
15	El usuario selecciona la opción Consultar	16	Muestra los datos del registro seleccionado



<b>Caso de Uso:</b>	Seguimiento
<b>Nombre:</b>	Seguimiento
<b>Actores:</b>	Ejecutor
<b>Propósito:</b>	Recepción y seguimiento de las solicitudes
<b>Tipo:</b>	Primario
<b>Descripción:</b>	Una vez que el usuario ingresa tendrá dos opciones: recibir solicitudes o dar seguimiento a ellas, para la recepción de solicitudes se le mostrará una lista con las solicitudes que le corresponde atender para cual se le pedirá ingrese su clave para recibirlas. Podrá modificar el estado de la solicitud y generar órdenes de trabajo
<b>Referencias:</b>	13,14, 15, 16,

**Curso de Eventos:** Seguimiento

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra página con las opciones: recepción y seguimiento
3	El usuario selecciona la opción recepción	4	Muestra lista con las solicitudes registradas y un campo donde deberá ingresar su clave para confirmar la recepción de la solicitud
5	El usuario ingresa su clave y presiona el botón recibir	6	Valida clave y cambia estado de la solicitud ó envía mensaje en caso de error.
7	El usuario selecciona la opción seguimiento	8	Muestra lista con las solicitudes y las opciones: Seguimiento y consultar.
9	El usuario presiona el botón modificarlo	10	Se muestra formulario cargado con los datos de la solicitud y donde puede cambiar el estado de la solicitud y agregar las acciones realizadas y negrear orden de trabajo
11	El usuario presiona el botón guardar	12	Valida información y pide confirmación, envía mensaje al usuario.
13	El usuario presiona el botón orden de trabajo	14	Genera en una nueva ventana un formato de orden de trabajo con los datos del solicitante y solicitud
15	El usuario selecciona la opción consultar seguimiento	16	Muestra los datos de la solicitud y el seguimiento de la misma.

<b>Caso de Uso:</b>	Reportes
<b>Nombre:</b>	Reportes
<b>Actores:</b>	Operador, Ejecutor, Directivo
<b>Propósito:</b>	Generar reportes de las solicitudes de servicios
<b>Tipo:</b>	Primario
<b>Descripción:</b>	Cuando entra el usuario podrá capturar varios parámetros para generar reportes de las solicitudes de servicios.
<b>Referencias:</b>	17

**Curso de Eventos:** Reportes

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra página con los parámetros para generar su reporte: Tipo de Reporte, Fecha de Recepción, Estado, Tipo de Captación, Ubicación y Organización
3	El usuario captura los parámetros del informe requerido y presiona el botón aceptar	4	Genera el reporte solicitado o envía mensaje en caso de haber error.

<b>Caso de Uso:</b>	Catálogos
<b>Nombre:</b>	Catálogos
<b>Actores:</b>	Operador, Administrador
<b>Propósito:</b>	Dar de alta, modificar o eliminar los catálogos necesarios para el funcionamiento del sistema
<b>Tipo:</b>	Primario
<b>Descripción:</b>	<p>Cuando el usuario entra a ésta parte del sistema tendrá varias opciones de donde deberá seleccionar el catálogo con el cuál desea trabajar: Colonias, calles, organizaciones, servicios, etc.</p> <p>Una vez seleccionada la opción se mostrará una lista con los registros del catálogo los cuáles podrá eliminar, modificar, consultar o agregar según sea el caso. Para la eliminación se validará que no esté relacionado con algún dato del sistema, para la modificación se solicitará confirmación y para agregar se le mostrará un formulario donde deberá llenar los datos solicitados para dar de alta su registro.</p>
<b>Referencias:</b>	18, 19, 20, 21, 22, 23

**Curso de Eventos:** Catálogos

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra página con los catálogos disponibles
3	El usuario selecciona el catálogo	4	Muestra un listado del catálogo con las opciones: agregar, eliminar, modificar, consultar, buscar e imprimir.
5	El usuario selecciona la opción Agregar	6	Se muestra formulario de captura donde el usuario teclea la información solicitada
7	El usuario captura datos y selecciona el botón guardar	8	Valida información y envía mensaje de error en su caso.
9	El usuario selecciona la opción Eliminar	10	Se pide confirmación, valida la operación y elimina enviando mensaje de error en su caso.
11	El usuario selecciona la opción Modificar	12	Se muestra formulario cargado con los datos del catalogo seleccionada
13	El usuario modifica los datos y selecciona el botón guardar	14	Valida información y envía mensaje de error en su caso.
15	El usuario selecciona la opción Consultar	16	Muestra los datos del registro seleccionado
17	El usuario selecciona la opción Buscar	18	Muestra formulario de búsqueda
19	El usuario captura el criterio de búsqueda y presiona el botón guardar	20	Valida información y envía mensaje de error en su caso.
21	El usuario selecciona la opción Imprimir	22	Genera en una ventana nueva el informe para impresión

<b>Caso de Uso:</b>	Administración
<b>Nombre:</b>	Administración
<b>Actores:</b>	Administrador
<b>Propósito:</b>	Personalizar el sistema, administrar los usuarios, permisos y bitácoras
<b>Tipo:</b>	Primario
<b>Descripción:</b>	El usuario podrá seleccionar las opciones: configurar, usuarios, bitácora de acceso y bitácora de operaciones.
<b>Referencias:</b>	24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34

**Curso de Eventos:** Administración

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra página con las opciones: configuración, usuarios y bitácoras de acceso y operaciones
3	El usuario selecciona la opción <i>Usuarios</i>	4	Muestra usuarios y opciones: agregar, eliminar, modificar, buscar e imprimir.
5	El usuario selecciona la opción <i>Agregar</i>	6	Se muestra formulario de captura.
7	El usuario captura datos y selecciona el botón guardar	8	Valida información y envía mensaje de error en su caso.
9	El usuario selecciona la opción <i>Eliminar Usuario</i>	10	Pide confirmación, valida operación, y elimina enviando mensaje.
11	El usuario selecciona la opción <i>Modificar Usuario</i>	12	Se muestra formulario cargado con los datos del <i>Usuario</i> seleccionado
13	El usuario modifica los datos y selecciona el botón guardar	14	Valida y guarda información y envía mensaje de error en su caso.
15	El usuario selecciona la opción <i>Consultar Usuario</i>	16	Muestra los datos de la <i>Usuario</i> seleccionado
17	El usuario selecciona la opción <i>Buscar</i>	18	Muestra formulario de búsqueda
19	Captura el criterio de búsqueda y presiona el botón <i>Aceptar</i>	20	Valida información y envía mensaje de error en su caso.
21	El usuario selecciona la opción <i>Imprimir</i>	22	Genera en una ventana nueva el informe para impresión
23	El usuario selecciona la opción <i>Bitácora de Acceso</i>	24	Muestra un listado de los accesos al sistema y la opción: imprimir
25	El usuario selecciona la opción <i>Imprimir</i>	26	Genera en una ventana nueva el informe para impresión
27	El usuario selecciona la opción <i>Bitácora de Operaciones</i>	28	Muestra un listado de las operaciones en el sistema y la opción: imprimir
29	El usuario selecciona la opción <i>Imprimir</i>	30	Genera en una ventana nueva el informe para impresión
31	El usuario selecciona la opción <i>Configuración</i>	32	Se muestra formulario con los datos de la <i>Dependencia</i> para modificarlos.
33	El usuario modifica los datos y presiona el botón <i>Guardar</i>	34	Valida y guarda la información y envía mensaje de error en su caso.

<b>Caso de Uso:</b>	Orientación
<b>Nombre:</b>	Orientación
<b>Actores:</b>	Operador
<b>Propósito:</b>	Mostrar información de los servicios ofrecidos por el sistema
<b>Tipo:</b>	Primario
<b>Descripción:</b>	Cuando el usuario entra a ésta página se le mostrará un formulario de selección para que escoja el servicio del cual quiere información de: Tiempo de respuesta, requisitos, costo, fundamento jurídico, etc.
<b>Referencias:</b>	35, 36

**Curso de Eventos:** Orientación

No	Acción del Actor	No	Respuesta del Sistema
1	El caso de uso inicia cuando el usuario ingresa a la página	2	Muestra un formulario donde puede seleccionar por materia el servicio del que desea información.
3	El usuario selecciona la materia y posteriormente el servicio y presiona el botón aceptar	4	Genera en una nueva ventana un formato con la información del Servicio en base a los parámetros indicados.

### 3.4 CONTRATOS DE LOS CASOS DE USO

**Contratos para el caso de uso:** [Página Principal](#)

**Contrato:** Valida\_usuarios()

<b>Nombre</b>	Valida_usuarios
<b>Responsabilidades</b>	Validar a los usuarios y sus permisos
<b>Tipo</b>	Sistema
<b>Referencias cruzadas</b>	R1, R2
<b>Notas</b>	Se valida que los usuarios tengan acceso al sistema y así mismo se validan los permisos que tienen para utilizar los módulos del sistema y los permisos para las funciones dentro de cada módulo.
<b>Excepciones</b>	Enviar un mensaje de error si no se localizó al usuario o si la contraseña no es válida y si no se tienen permisos para acceder al sistema
<b>Salida</b>	Envía mensaje de bienvenida y muestra el nombre del usuario y el área a la que está asignada la cuenta de usuario. Activa los módulos y sus funciones a las que tiene permiso el usuario
<b>Pre-condiciones</b>	Debe existir el usuario El usuario debe tener permisos para el sistema, sus módulos y las funciones de cada módulo
<b>Post-condiciones</b>	Los datos personales y permisos del usuario se almacenan en el servidor para utilizarlos mientras esté activa la sesión. Se registra operación en la bitácora de accesos.

**Contratos para el caso de uso: Registro**

**Contrato:** Registra\_datos\_solicitante()

**Nombre** Registra\_solicitante  
**Responsabilidades** Dar de alta un solicitante para registrar solicitudes  
**Tipo** Sistema  
**Referencias cruzadas** R4  
**Notas** Se almacenan los datos en la tabla de solicitantes  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos  
**Salida** Envía mensaje de confirmación del ingreso  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para dar de alta a los solicitantes y deben tener permiso para registrar solicitantes  
**Post-condiciones** Los datos personales del solicitante se almacenan en la tabla de solicitantes  
 Se crea una instancia de Datos Solicitante  
 Se crea una instancia de Datos Solicitud  
 Se registra operación en la bitácora

**Contrato:** Modifica\_datos\_solicitante()

**Nombre** Modifica\_solicitante  
**Responsabilidades** Modifica los datos de un solicitante  
**Tipo** Sistema  
**Referencias cruzadas** R5  
**Notas** Se almacenan los datos en la tabla de solicitantes  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos  
**Salida** Envía mensaje de confirmación de la modificación  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para modificar a los solicitantes y deben tener permiso para ello.  
 Se debe seleccionar el solicitante a modificar  
**Post-condiciones** Los datos personales del solicitante se actualizan en la tabla de solicitantes  
 Se registra operación en la bitácora

**Contrato:** Elimina\_registro\_solicitante()

**Nombre** Elimina\_solicitante  
**Responsabilidades** Elimina registros solicitantes  
**Tipo** Sistema  
**Referencias cruzadas** R6  
**Notas** Se elimina registro de la tabla de solicitantes  
**Excepciones** Enviar un mensaje de error si no se eliminó el registro  
**Salida** Envía mensaje de eliminación de registro  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para eliminar a los solicitantes y deben tener permiso para eliminar solicitantes  
 Se debe seleccionar el solicitante a eliminar.  
 El solicitante no debe tener registros relacionados  
**Post-condiciones** Busca el solicitante a eliminar y actualiza los datos de la tabla solicitantes  
 Se registra operación en la bitácora

**Contrato:** Consulta\_datos\_solicitante()

**Nombre** Consulta\_solicitante  
**Responsabilidades** Muestra los datos del solicitante  
**Tipo** Sistema  
**Referencias cruzadas** R7  
**Notas** Se muestran los datos del solicitante que se encuentran en la base de datos  
**Excepciones**  
**Salida** Muestra los datos del solicitante  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de solicitantes y deben tener permiso para consultar los datos de solicitantes.  
 El solicitante debe estar registrado en la tabla solicitantes  
**Post-condiciones** Se crea una instancia de Datos Solicitante  
 Se hace una consulta en la base de datos de los datos personales del solicitante  
 Se registra operación en la bitácora



**Contrato:** Registra\_datos\_solicitud()

**Nombre** Registra\_solicitud

**Responsabilidades** Dar de alta una solicitud

**Tipo** Sistema

**Referencias cruzadas** R8

**Notas** Se almacenan los datos en la tabla de solicitudes

**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos

**Salida** Envía mensaje de confirmación del ingreso

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para dar de alta a las solicitudes y deben tener permiso para registrar solicitudes

**Post-condiciones** Los datos generales de la solicitud se almacenan en la tabla de solicitudes y solicitudes\_detalle  
Se crea una instancia de Datos Solicitud  
Se registra operación en la bitácora

**Contrato:** Modifica\_datos\_solicitud ()

**Nombre** Modifica\_solicitud

**Responsabilidades** Modifica los datos de una solicitud

**Tipo** Sistema

**Referencias cruzadas** R9

**Notas** Se almacenan los datos en la tabla de solicitudes

**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos

**Salida** Envía mensaje de confirmación de la modificación

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para modificar solicitudes y deben tener permiso para ello  
Se debe seleccionar la solicitud a modificar

**Post-condiciones** Los datos generales de la solicitud se actualizan en la tabla de solicitudes y solicitudes\_detalle  
Se registra operación en la bitácora

**Contrato:** Elimina\_registro\_solicitud()

**Nombre** Elimina\_solicitud  
**Responsabilidades** Elimina registros de solicitudes  
**Tipo** Sistema  
**Referencias cruzadas** R10  
**Notas** Se elimina registro de la tabla de solicitudes  
**Excepciones** Enviar un mensaje de error si no se eliminó el registro  
**Salida** Envía mensaje de eliminación de registro  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para eliminar solicitudes y deben tener permiso para eliminar solicitudes  
 Se debe seleccionar la solicitud a eliminar.  
 Las solicitudes no debe tener registros relacionados  
**Post-condiciones** Busca la solicitud a eliminar y actualiza los datos de la tabla solicitudes y solicitudes\_detalle.  
 Se registra operación en la bitácora

**Contrato:** Consulta\_datos\_solicitud()

**Nombre** Consulta\_solicitud  
**Responsabilidades** Muestra los datos de la solicitud  
**Tipo** Sistema  
**Referencias cruzadas** R12  
**Notas** Se muestran los datos de la solicitud que se encuentran en la base de datos  
**Excepciones**  
**Salida** Muestra los datos de la solicitud  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de solicitudes y deben tener permiso para consultar los datos de solicitudes.  
 La solicitud debe estar registrada en la tabla solicitantes y solicitantes\_detalle  
**Post-condiciones** Se crea una instancia de Datos Solicitud  
 Se hace una consulta en la base de datos de los datos generales de la solicitud  
 Se registra operación en la bitácora

<b>Contrato:</b>	Genera_acuse_recibo()
<b>Nombre</b>	Genera_acuse
<b>Responsabilidades</b>	Mostrar un formato de acuse de recibo
<b>Tipo</b>	Sistema
<b>Referencias cruzadas</b>	R13
<b>Notas</b>	El acuse debe contener los datos generales de la solicitud y su solicitante, así como número de folio
<b>Excepciones</b>	
<b>Salida</b>	Muestra el formato de acuse de recibo
<b>Pre-condiciones</b>	Los usuarios deben de acceder a la aplicación para generar acuses de recibo y deben tener permiso para ello. La solicitud debe estar registrada en la tabla solicitantes y solicitantes_detalle
<b>Post-condiciones</b>	Se crea una instancia de Datos Solicitante Se crea una instancia de Datos Solicitud Se hace una consulta en la base de datos de los datos generales de la solicitud y su solicitante Se registra operación en la bitácora

**Contratos para el caso de uso: Seguimiento**

**Contrato:** Recepción\_solicitud()

**Nombre** Recepción\_solicitud  
**Responsabilidades** Recibir electrónicamente las solicitudes  
**Tipo** Sistema  
**Referencias cruzadas** R14  
**Notas** Se recibe en cada área las solicitudes que les hayan sido turnadas y se modifica el estado de la solicitud en la tabla solicitudes\_detalle

**Excepciones**

**Salida** Envía mensaje de recepción de la solicitud  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para recibir solicitudes y deben tener permiso para ello. La solicitud debe estar registrada en la tabla solicitantes y solicitantes\_detalle y debe ser turnada a una de las áreas registradas en la tabla areas

**Post-condiciones** Se modifica el estado de la solicitud  
 Se registra operación en la bitácora

**Contrato:** Modifica\_estado\_solicitud()

**Nombre** Modifica\_edo\_solicitud  
**Responsabilidades** Modificar el estado de una solicitud  
**Tipo** Sistema  
**Referencias cruzadas** R15  
**Notas** Se modifica el estado de una solicitud y se anexan observaciones sobre la misma

**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos

**Salida** Envía mensaje de modificación de la solicitud  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para modificar el estado de las solicitudes y deben tener permiso para ello. La solicitud debe estar registrada en la tabla solicitantes y solicitantes\_detalle y debe estar turnada a una de las áreas registradas en la tabla areas

**Post-condiciones** Se modifica el estado de la solicitud  
 Se registra operación en la bitácora



**Contrato:** Consulta\_seguimiento\_solicitud()

**Nombre** Consulta\_solicitud

**Responsabilidades** Muestra los datos de la solicitud

**Tipo** Sistema

**Referencias cruzadas** R16

**Notas** Se muestran los datos de la solicitud que se encuentran en la base de datos

**Excepciones**

**Salida** Muestra los datos de la solicitud

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de solicitudes y deben tener permiso para consultar los datos de solicitudes.

La solicitud debe estar registrada en la tabla solicitantes y solicitantes\_detalle

**Post-condiciones** Se crea una instancia de Datos Solicitud

Se hace una consulta en la base de datos de los datos generales de la solicitud

Se registra operación en la bitácora

**Contrato:** Genera\_ordenes\_trabajo()

**Nombre** Genera\_ordenes

**Responsabilidades** Genera un formato de orden de trabajo

**Tipo** Sistema

**Referencias cruzadas** R17

**Notas** Genera las ordenes de trabajo para las cuadrillas de trabajadores con la información de la solicitud y el solicitante

**Excepciones**

**Salida** Muestra el formato de orden de trabajo

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para generar órdenes de trabajo y deben tener permiso para ello.

La solicitud debe estar registrada en la tabla solicitantes y solicitantes\_detalle

**Post-condiciones** Se hace una consulta en la base de datos de los datos generales de la solicitud y su solicitante

Se registra operación en la bitácora

Contratos para el caso de uso:                      Reportes

**Contrato:** Genera\_reportes ()

**Nombre**    Genera\_reporte

**Responsabilidades**                              Generar un reporte de las solicitudes

**Tipo**    Sistema

**Referencias cruzadas**                          R17

**Notas**    Generar un reporte de las solicitudes conforme a los parámetros indicados por el usuario

**Excepciones**                                        Enviar un mensaje de error si no se llenaron correctamente los datos o si no se encontraron registros

**Salida**    Muestra el reporte

**Pre-condiciones**                                    Los usuarios deben de acceder a la aplicación para generar reportes y deben tener permiso para ello.

**Post-condiciones**                                  Se realiza la búsqueda de solicitudes en la base de datos conforme a los parámetros indicados por el usuario

Se registra operación en la bitácora

**Contratos para el caso de uso: Catálogos**

**Contrato:** Registra\_datos\_catálogo()

**Nombre** Registra\_datos\_catálogo  
**Responsabilidades** Dar de alta un registro dentro de un catálogo  
**Tipo** Sistema  
**Referencias cruzadas** R18  
**Notas** Se almacenan los datos en la tabla correspondiente  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos  
**Salida** Envía mensaje de confirmación del ingreso  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para dar de alta un registro y deben tener permiso para ello  
**Post-condiciones** Los datos generales del registro se almacenan en la tabla correspondiente  
 Se crea una instancia de Datos Catálogo  
 Se registra operación en la bitácora

**Contrato:** Modifica\_datos\_catálogo ()

**Nombre** Modifica\_datos\_catálogo  
**Responsabilidades** Modifica los datos de un registro de un catálogo  
**Tipo** Sistema  
**Referencias cruzadas** R19  
**Notas** Se almacenan los datos en la tabla correspondiente  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos  
**Salida** Envía mensaje de confirmación de la modificación  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para modificar el registro de un catálogo y deben tener permiso para ello  
 Se debe seleccionar el registro a modificar  
**Post-condiciones** Los datos generales del registro se actualizan en la tabla correspondiente  
 Se registra operación en la bitácora

**Contrato:** Elimina\_registro\_catálogo()

**Nombre** Elimina\_registro\_catálogo  
**Responsabilidades** Elimina registros de un catálogo  
**Tipo** Sistema  
**Referencias cruzadas** R20  
**Notas** Se elimina el registro de la tabla correspondiente al catálogo  
**Excepciones** Enviar un mensaje de error si no se eliminó el registro  
**Salida** Envía mensaje de eliminación de registro  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para eliminar registros del catálogo y deben tener permiso para ello  
 Se debe seleccionar la delegación a eliminar.  
 El registro no debe estar referenciado por otras tablas  
**Post-condiciones** Busca el registro a eliminar y actualiza los datos de la tabla correspondiente  
 Se registra operación en la bitácora

**Contrato:** Consulta\_datos\_catálogo()

**Nombre** Consulta\_datos\_catálogo  
**Responsabilidades** Muestra los datos del registro de un catálogo  
**Tipo** Sistema  
**Referencias cruzadas** R21  
**Notas** Se muestran los datos del registro que se encuentra en la base de datos  
**Excepciones**  
**Salida** Muestra los datos del registro  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos y deben tener permiso para ello.  
 El registro debe estar en la tabla correspondiente  
**Post-condiciones** Se crea una instancia de Datos  
 Se hace una consulta en la base de datos de los datos generales del registro  
 Se registra operación en la bitácora



**Contrato:** Buscar\_registro\_catálogo ()

**Nombre** Buscar\_registro\_catálogo  
**Responsabilidades** Buscar registros dentro de un catálogo  
**Tipo** Sistema  
**Referencias cruzadas** R22  
**Notas** Se realiza una búsqueda en la tabla correspondiente  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos de búsqueda  
**Salida** Envía registros encontrados en la tabla  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para realizar búsquedas en el catálogo correspondiente y deben tener permiso para ello  
**Post-condiciones** Se realiza búsqueda de registros en la tabla correspondiente.  
 Se crea una instancia de Datos  
 Se registra operación en la bitácora

**Contrato:** Imprimir\_catálogo()

**Nombre** Imprimir\_catálogo  
**Responsabilidades** Imprimir reporte del catálogo correspondiente  
**Tipo** Sistema  
**Referencias cruzadas** R23  
**Notas** Se imprime un reporte con los registros que contiene el catálogo correspondiente  
**Excepciones** Enviar mensaje cuando no existan datos en el catálogo correspondiente  
**Salida** Muestra el reporte con la información de los registros contenidos en la tabla correspondiente  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para imprimir el catálogo y deben tener permiso para ello. El catálogo debe tener información para poder generar el reporte  
**Post-condiciones** Se hace una consulta en la base de datos de los datos al catálogo correspondiente.  
 Se registra operación en la bitácora

**Contratos para el caso de uso: Administración**

**Contrato:** Registra\_datos\_usuario()

<b>Nombre</b>	Registra_usuario
<b>Responsabilidades</b>	Dar de alta un usuario
<b>Tipo</b>	Sistema
<b>Referencias cruzadas</b>	R25
<b>Notas</b>	Se almacenan los datos en la tabla usuarios
<b>Excepciones</b>	Enviar un mensaje de error si no se llenaron correctamente los datos
<b>Salida</b>	Envía mensaje de confirmación del ingreso
<b>Pre-condiciones</b>	Los usuarios deben de acceder a la aplicación para dar de alta a los usuarios y deben tener permiso para ello
<b>Post-condiciones</b>	Los datos generales del usuario se almacenan en la tabla usuarios Se crea una instancia de Datos Usuario Se registra operación en la bitácora

**Contrato:** Modifica\_datos\_usuario()

<b>Nombre</b>	Modifica_usuario
<b>Responsabilidades</b>	Modifica los datos de un usuario
<b>Tipo</b>	Sistema
<b>Referencias cruzadas</b>	R26
<b>Notas</b>	Se almacenan los datos en la tabla de usuarios
<b>Excepciones</b>	Enviar un mensaje de error si no se llenaron correctamente los datos
<b>Salida</b>	Envía mensaje de confirmación de la modificación
<b>Pre-condiciones</b>	Los usuarios deben de acceder a la aplicación para modificar a los usuarios y deben tener permiso para ello Se debe seleccionar el usuario a modificar
<b>Post-condiciones</b>	Los datos personales del usuario se actualizan en la tabla de usuarios Se registra operación en la bitácora

**Contrato:** Elimina\_registro\_usuario()

**Nombre** Elimina\_usuario

**Responsabilidades** Elimina registros de usuarios

**Tipo** Sistema

**Referencias cruzadas** R27

**Notas** Se elimina registro de la tabla usuarios

**Excepciones** Enviar un mensaje de error si no se eliminó el registro

**Salida** Envía mensaje de eliminación de registro

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para eliminar a los usuarios y deben tener permiso para ello  
Se debe seleccionar el usuario a eliminar.

El usuario no debe tener registros relacionados

**Post-condiciones** Busca el usuario a eliminar y actualiza los datos de la tabla usuarios

Se registra operación en la bitácora

**Contrato:** Consulta\_datos\_usuario()

**Nombre** Consulta\_usuario

**Responsabilidades** Muestra los datos del usuario

**Tipo** Sistema

**Referencias cruzadas** R28

**Notas** Se muestran los datos del usuario que se encuentran en la base de datos

**Excepciones**

**Salida** Muestra los datos del usuario

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de usuarios y deben tener permiso para ello.

El usuario debe estar registrado en la tabla usuarios

**Post-condiciones** Se crea una instancia de Datos Usuario

Se hace una consulta en la base de datos de los datos generales del usuario

Se registra operación en la bitácora

**Contrato:** Buscar\_usuario ()

**Nombre** Buscar\_usuario  
**Responsabilidades** Buscar registros dentro de la tabla de usuarios  
**Tipo** Sistema  
**Referencias cruzadas** R29  
**Notas** Se realiza una búsqueda en la tabla usuarios  
**Excepciones** Enviar un mensaje de error si no se llenaron correctamente los datos de búsqueda  
**Salida** Envía registros encontrados en la usuario  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para realizar búsquedas en la tabla usuario y deben tener permiso para ello  
**Post-condiciones** Se realiza búsqueda de registros en la tabla usuarios.  
 Se crea una instancia de Datos Usuario  
 Se registra operación en la bitácora

**Contrato:** Imprimir\_usuarios()

**Nombre** Imprimir\_usuarios  
**Responsabilidades** Imprimir reporte del los usuarios del sistema  
**Tipo** Sistema  
**Referencias cruzadas** R30  
**Notas** Se imprime un reporte con los usuarios registrados en el sistema  
**Excepciones** Enviar mensaje cuando no existan datos en la tabla de usuarios  
**Salida** Muestra el reporte de usuarios del sistema  
**Pre-condiciones** Los usuarios deben de acceder a la aplicación para imprimir el reporte de usuarios y deben tener permiso para ello.  
 La tabla usuarios debe tener información para poder generar el reporte  
**Post-condiciones** Se hace una consulta en la base de datos de los datos a la tabla usuarios.  
 Se registra operación en la bitácora

**Contrato:** Consulta\_bitacora\_acceso()

**Nombre** Consulta\_bitacora\_acceso  
**Responsabilidades** Muestra los datos de la bitácora de accesos  
**Tipo** Sistema  
**Referencias cruzadas** R31  
**Notas** Se muestran los datos de la bitácora de accesos que se encuentran en la base de datos

**Excepciones**

**Salida** Muestra los datos de la bitácora de accesos

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de la bitácora de accesos y deben tener permiso para ello.  
 Deben existir datos en la tabla bitacora\_acceso

**Post-condiciones** Se crea una instancia de Datos Bitácora Acceso  
 Se hace una consulta en la base de datos de los datos contenidos en la tabla bitacora\_acceso

**Contrato:** Imprimir\_bitacora\_acceso ()

**Nombre** Imprimir\_bitacora\_acceso  
**Responsabilidades** Imprimir reporte de la bitácora de accesos al sistema

**Tipo** Sistema  
**Referencias cruzadas** R32  
**Notas** Se imprime un reporte con los accesos hechos y registrados en el sistema

**Excepciones** Enviar mensaje cuando no existan datos en la tabla bitacora\_acceso

**Salida** Muestra el reporte de accesos al sistema

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para imprimir la bitácora de accesos y deben tener permiso para ello.  
 La tabla bitacora\_acceso debe tener información para poder generar el reporte

**Post-condiciones** Se hace una consulta en la base de datos de los datos a la tabla la bitácora\_acceso.

**Contrato:** Consulta\_bitacora\_operaciones()

**Nombre** Consulta\_bitacora\_operaciones  
**Responsabilidades** Muestra los datos de la bitácora de operaciones  
**Tipo** Sistema  
**Referencias cruzadas** R33  
**Notas** Se muestran los datos de la bitácora de operaciones que se encuentran en la base de datos

**Excepciones**

**Salida** Muestra los datos de la bitácora de operaciones

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de la bitácora de operaciones y deben tener permiso para ello.  
 Deben existir datos en la tabla bitacora\_operacion

**Post-condiciones** Se crea una instancia de Datos Bitácora Operacion  
 Se hace una consulta en la base de datos de los datos contenidos en la tabla bitacora\_operacion

**Contrato:** Imprimir\_bitacora\_operaciones ()

**Nombre** Imprimir\_bitacora\_operaciones  
**Responsabilidades** Imprimir reporte de la bitácora de operaciones en el sistema  
**Tipo** Sistema  
**Referencias cruzadas** R34  
**Notas** Se imprime un reporte con las operaciones hechas y registradas en el sistema

**Excepciones** Enviar mensaje cuando no existan datos en la tabla bitacora\_operacion

**Salida** Muestra el reporte de operaciones realizadas en el sistema

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para imprimir la bitácora de operaciones y deben tener permiso para ello.  
 La tabla bitacora\_operacion debe tener información para poder generar el reporte

**Post-condiciones** Se hace una consulta en la base de datos de los datos a la tabla la bitacora\_operacion

**Contratos para el caso de uso: Orientación**

**Contrato:** Consulta\_datos\_orientacion()

**Nombre** Consulta\_orientacion

**Responsabilidades** Mostrar la información detallada de los servicios

**Tipo** Sistema

**Referencias cruzadas** R35

**Notas** Se muestra la información detallada de los servicios (según los datos contenidos en el MTYSP del D.F..) que se encuentran en la base de datos, según las especificaciones del usuario

**Excepciones**

**Salida** Muestra la información detallada de los servicios

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para consultar los datos de los servicios y deben tener permiso para ello.

**Post-condiciones** El servicio debe estar registrado en la tabla servicios

Se crea una instancia de Datos Servicio

Se hace una consulta en la base de datos de los datos generales del servicio

**Contrato:** Imprime\_datos\_orientacion()

**Nombre** Imprime\_orientacion

**Responsabilidades** Mostrar un formato semejante a las cédulas del Manual de Trámites y Servicios al Público del D.F.

**Tipo** Sistema

**Referencias cruzadas** R36

**Notas** El formato debe contener la información detallada del servicio

**Excepciones**

**Salida** Muestra el formato de cédula de servicio

**Pre-condiciones** Los usuarios deben de acceder a la aplicación para imprimir las cédulas de servicio y deben tener permiso.

**Post-condiciones** El servicio debe estar registrado en la tabla servicios

Se crea una instancia de Datos Servicio

Se hace una consulta en la base de datos de los datos generales de la solicitud y su solicitante

### 3.5 DIAGRAMAS DE SECUENCIA

La forma en que interactúan los usuarios del sistema con éste se muestra en la Figura 3-7 y Figura 3-8.

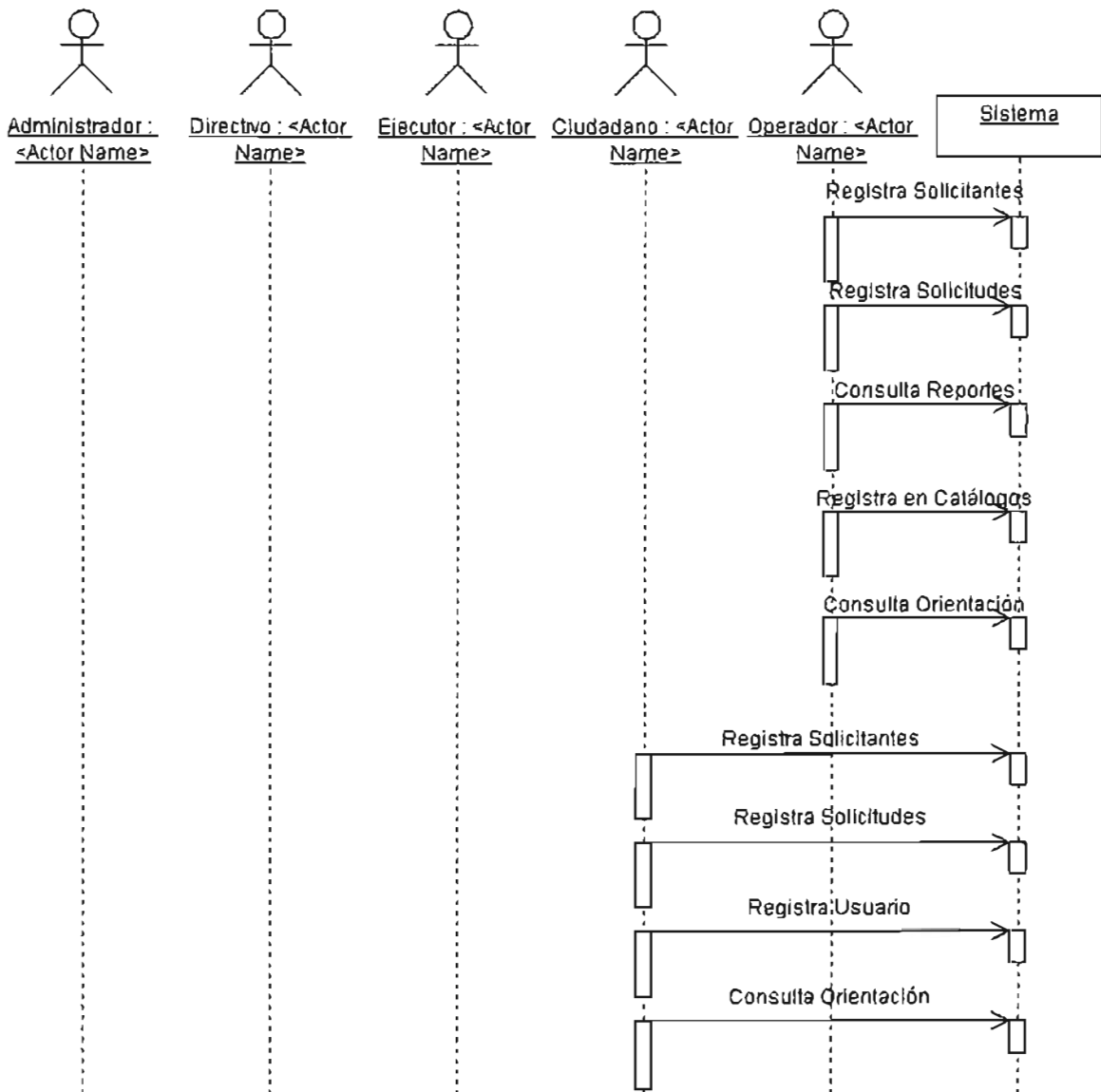


Figura 3-7



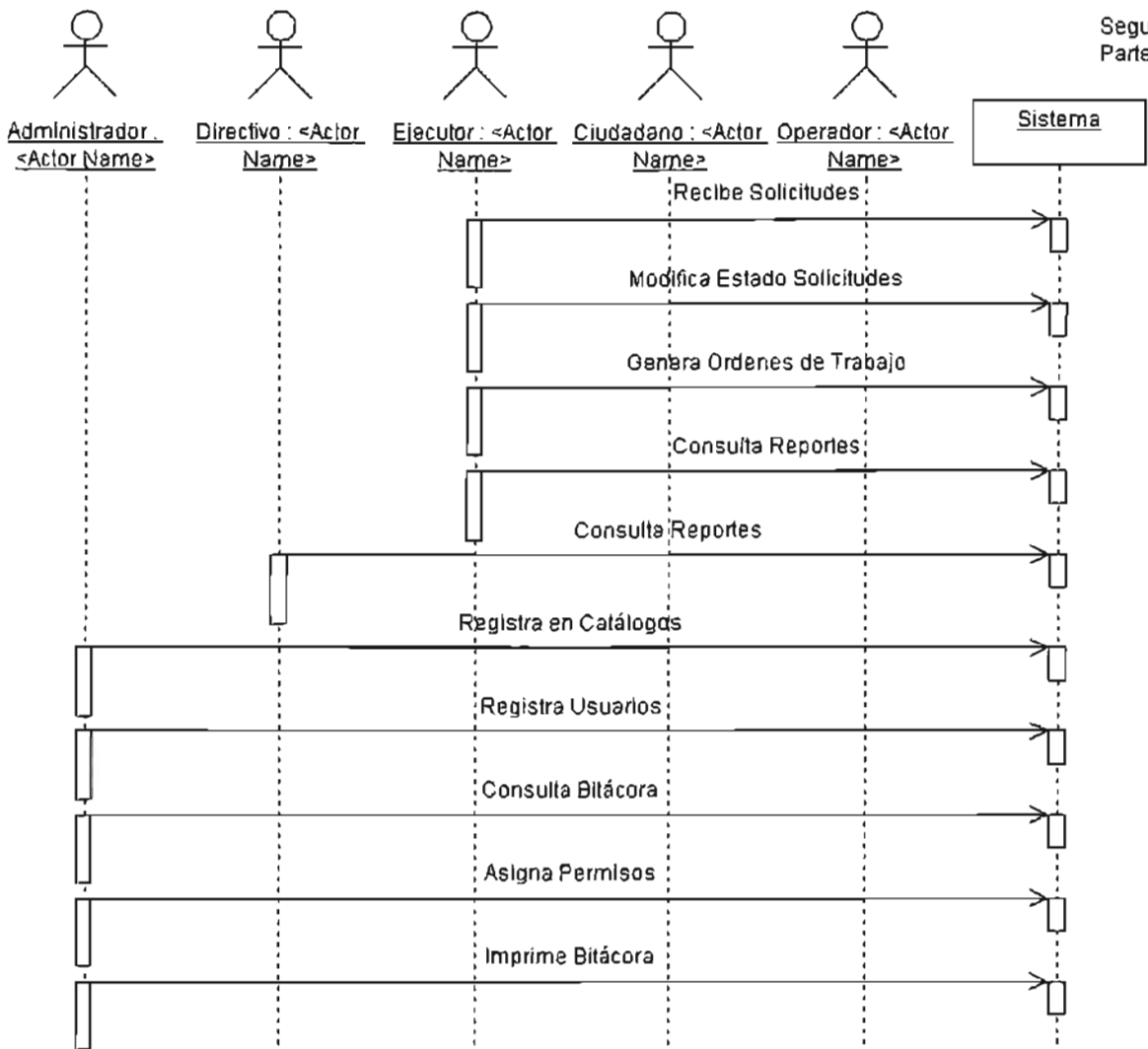


Figura 3-8

## TEMA 4 DISEÑO DE INTERFAZ

### 4.1 INTRODUCCIÓN

Una vez definido el problema en el Tema 2, determinado los requerimientos en el tema 3.1 y modelado el sistema en UML en los numerales 3.4 y 3.5, procederemos a describir la propuesta de Interfaz Gráfica de la aplicación para ello se describirán los diagramas de navegación y la ventanas de cada una de las partes del sistema.

### 4.2 SUBSISTEMAS

Programa de Gestión de Servicios - Diagrama de Navegación

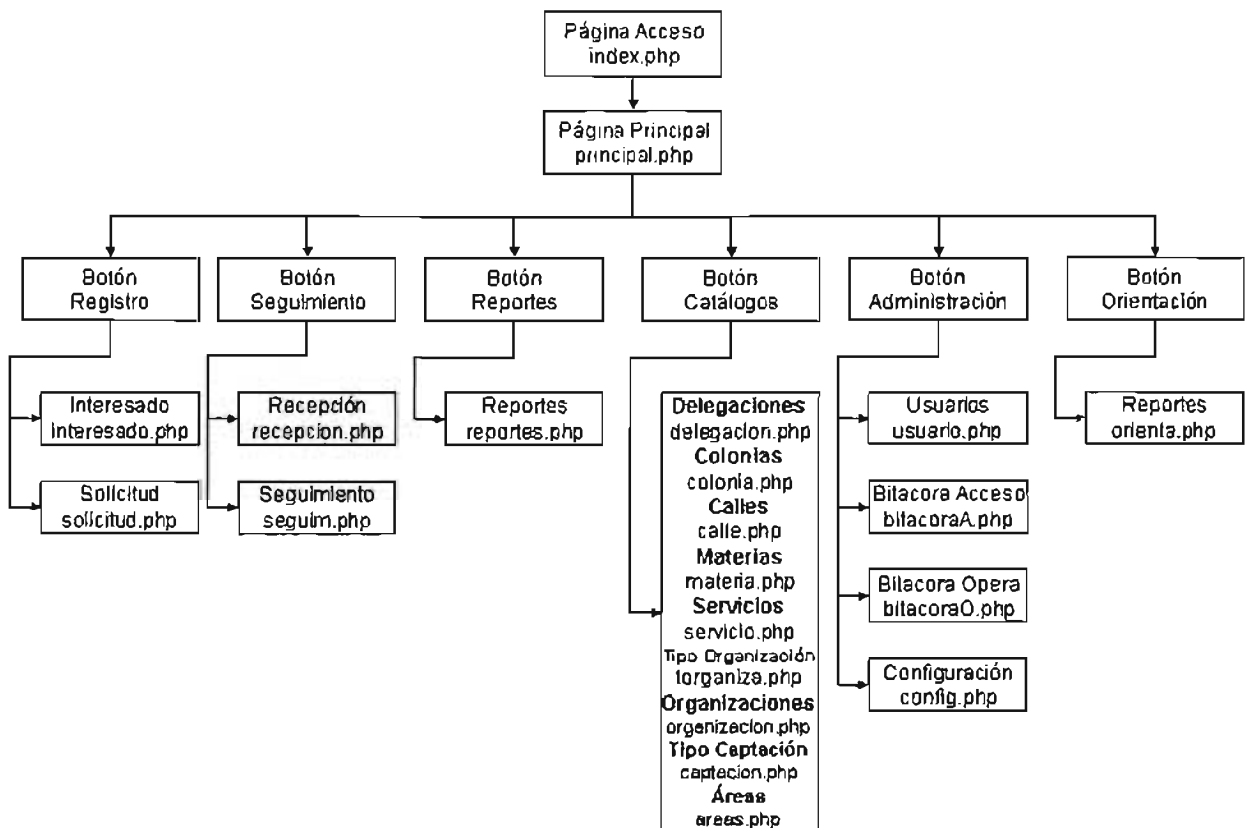


Figura 4-1

En la Figura 4-1 se puede observar el Diagrama de Navegación principal del sistema, que cuenta con 6 módulos cuyos Diagramas de Navegación e interfaces se irán mostrando y explicando en las siguientes páginas.

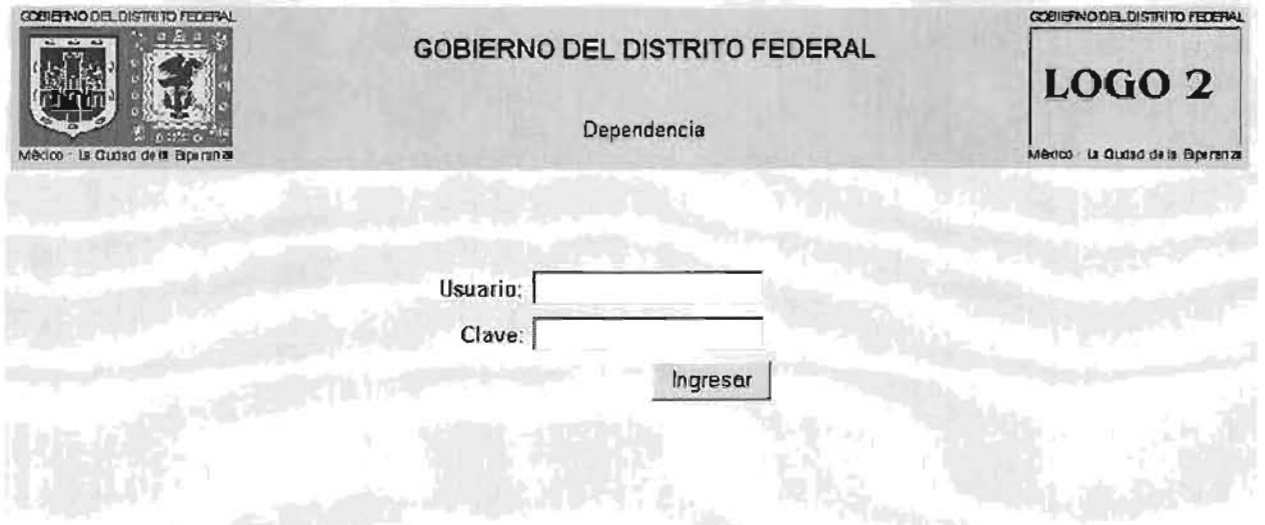


Figura 4-2 index.php

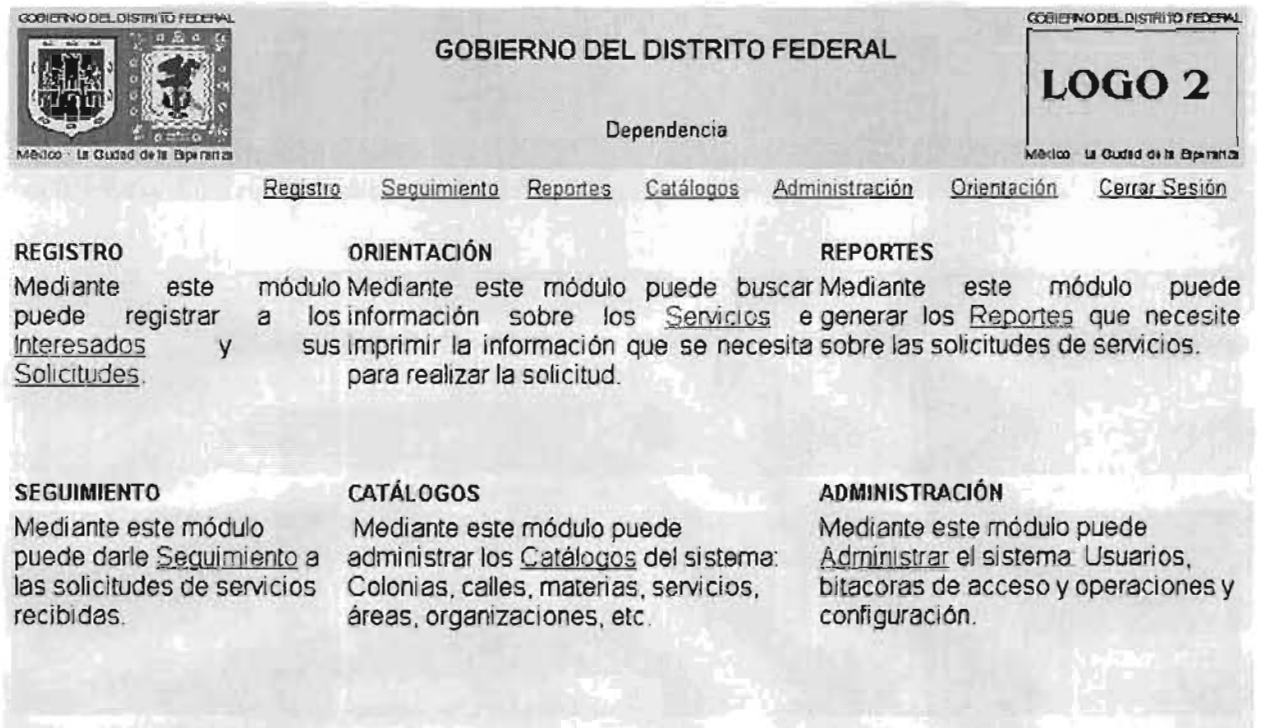


Figura 4-3 principal.php

4.2.1 MÓDULO DE REGISTRO

Subsistema Registro - Interesados

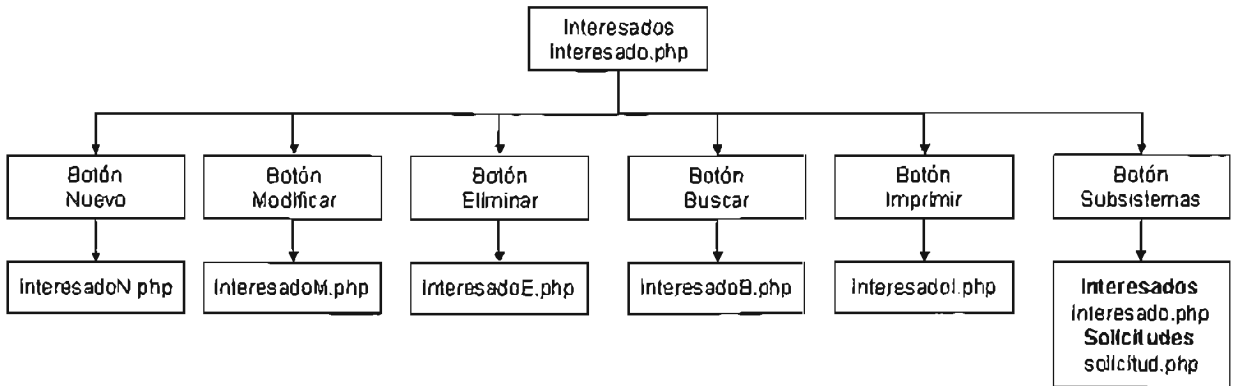


Figura 4-4

ID	Paterno	Materno	Nombre	RFC	Teléfono	
1	Arreola		Juan José	ARJJ650201	55 55 99 99	
8	Bernal	Almeda	Ricardo	BEAR620609	52 22 21 23	
9	Bernal	García	Ricardo	BEGR420523	52 21 00 01	
3	Fuentes	Márquez	Carlos	MAFC300605	54 53 52 11	
7	Guzmán	Wolfer	Ricardo	WOGR720130	55 95 96 99	
5	Luna	Méndez	Carolina	MELC720902	52 50 48 46	
10	Merguen	Alba	Erika	MEAE600811	55 06 05 08	
2	Páramo	Rulfo	Pedro	RUPP450308	55 12 12 11	
6	Paz	Paz	Octavio	PAP300922	51 49 47 20	
4	Rodríguez	Luna	Juan José	LURJ601210	55 22 21 20	

Página 1/1

Figura 4-5 interesado.php

[Solicitudes](#)
[Interesados](#)
[Regresar](#)

### REGISTRO DE INTERESADO

<b>Nombre</b>	<b>Apellido Paterno</b>	<b>Apellido Materno</b>	<b>RFC</b>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>	<b>Num Interior</b>
<input type="text" value="Alvaro Obregón"/>	<input type="text" value="Colonia1"/>	<input type="text" value="Calle1"/>	<input type="text"/>
<b>Num Ext</b>	<b>Teléfono</b>	<b>Fax</b>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>Organización</b>		<b>Puesto</b>	
<input type="text"/>		<input type="text"/>	

Figura 4-6 interesadoN.php

[Solicitudes](#)
[Interesados](#)
[Regresar](#)

### DETALLE DE INTERESADO

<b>Nombre</b>	<b>Apellido Paterno</b>	<b>Apellido Materno</b>	<b>RFC</b>
Nombre	Apellido Paterno	Apellido Materno	RFC
<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>	<b>Num Interior</b>
Delegación	Colonia	Calle	Número Interior
<b>Num Ext</b>	<b>Teléfono</b>	<b>Fax</b>	
Número Exterior	Teléfono	Fax	
<b>Organización</b>		<b>Puesto</b>	
Organización		Puesto	

[Regresar](#)

Figura 4-7 interesadoD.php

[Solicitudes](#)
[Interesados](#)
[Regresar](#)

### MODIFICACIÓN DE INTERESADO

<b>Nombre</b> <input type="text"/>	<b>Apellido Paterno</b> <input type="text"/>	<b>Apellido Materno</b> <input type="text"/>	<b>RFC</b> <input type="text"/>
<b>Delegación</b> <input type="text" value="Alvaro Obregón"/>	<b>Colonia</b> <input type="text" value="Colonia1"/>	<b>Calle</b> <input type="text" value="Calle1"/>	<b>Num Interior</b> <input type="text"/>
<b>Num Ext</b> <input type="text"/>	<b>Teléfono</b> <input type="text"/>	<b>Fax</b> <input type="text"/>	
<b>Organización</b> <input type="text" value="Partido x"/>	<b>Puesto</b> <input type="text"/>		

Figura 4-8 interesadoM.php

[Solicitudes](#)
[Interesados](#)
[Regresar](#)

### ELIMINACIÓN DE INTERESADO

<b>Nombre</b> Nombre	<b>Apellido Paterno</b> Apellido Paterno	<b>Apellido Materno</b> Apellido Materno	<b>RFC</b> RFC
<b>Delegación</b> Delegación	<b>Colonia</b> Colonia	<b>Calle</b> Calle	<b>Num Interior</b> Número Interior
<b>Num Ext</b> Número Exterior	<b>Teléfono</b> Teléfono	<b>Fax</b> Fax	
<b>Organización</b> Organización	<b>Puesto</b> Puesto		

Figura 4-9 interesadoE.php

[Solitudes](#) [Interesados](#) [Regresar](#)

### BÚSQUEDA DE INTERESADO

<b>Nombre</b>	<b>Apellido Paterno</b>	<b>Apellido Materno</b>	<b>RFC</b>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>	<b>Num Interior</b>
<input type="text" value="Alvaro Obregón"/>	<input type="text" value="Colonia"/>	<input type="text" value="Calle1"/>	<input type="text"/>
<b>Num Ext</b>	<b>Teléfono</b>	<b>Fax</b>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
<b>Organización</b>		<b>Puesto</b>	
<input type="text"/>		<input type="text"/>	

Figura 4-10 interesadoB.php

Subsistema Registro - Solicitudes

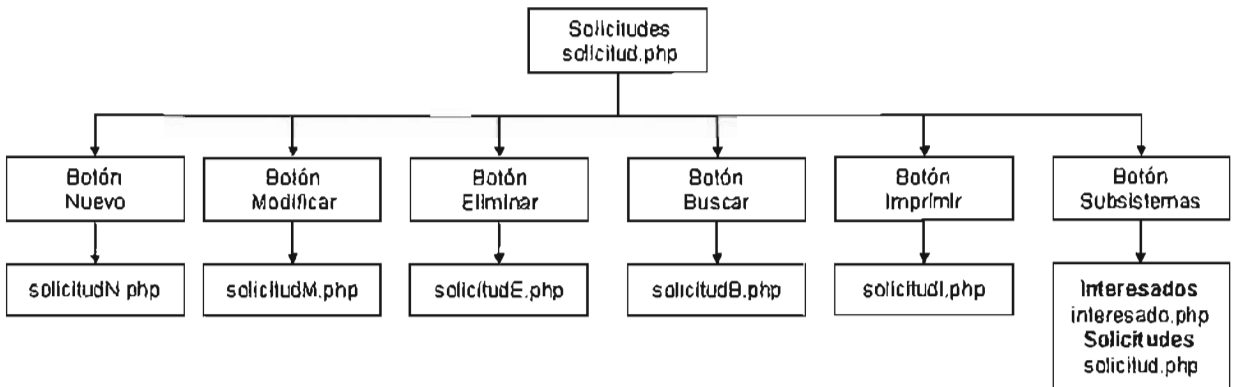


Figura 4-11

Año	Folio	Solicitud	Interesado	Fecha	Estado	
2004	1	Atención a falta de Aguas	Juan José	Sun 13, Jun 2004	Ingresado	
2004	10	Atención a falta de Aguas	Erika	Sun 13, Jun 2004	Ingresado	
2004	8	Limpieza de calles	Ricardo	Sun 13, Jun 2004	Proceso	
2004	2	Atención a falta de Aguas	Pedro	Sun 13, Jun 2004	Ingresado	
2004	5	Reparación de luminarias	Carolina	Sun 13, Jun 2004	Proceso	
2004	3	Bacheo	Carlos	Sat 12, Jun 2004	Atendido	
2004	4	Bacheo	Juan José	Sun 13, Jun 2004	Ingresado	
2004	6	Asesoría Jurídica	Octavio	Sun 13, Jun 2004	Proceso	
2004	9	Limpieza de calles	Ricardo	Sun 13, Jun 2004	Ingresado	
2004	7	Asesoría Jurídica	Ricardo	Sun 13, Jun 2004	Proceso	

Página 1/1

Figura 4-12 solicitud.php



[Solicitudes](#) [Interesados](#) [Regresar](#)

### REGISTRO DE SOLICITUD

Interesado ID <input type="text" value="0001"/>	Materia <input type="text" value="Agraria"/>	Servicio <input type="text" value="Atención a falta de agua"/>	
Tipo de Captación <input type="text" value="Personal"/>	Folio Externo <input type="text"/>		

Observaciones

### DETALLES

Delegación <input type="text" value="Cuauhtémoc"/>	Colonia <input type="text" value="Centro"/>	Calle <input type="text" value="Izazaga"/>	
Número Interior <input type="text"/>	Número Exterior <input type="text"/>	Entre Calle <input type="text" value="5 de Febrero"/>	Y Calle <input type="text" value="Isabel la Católica"/>

Figura 4-13 solicitudN.php

[Solicitudes](#) [Interesados](#) [Regresar](#)

### DETALLE DE SOLICITUD

Interesado ID Interesado ID	Materia Materia	Servicio Servicio	
Tipo de Captación Tipo de Captación	Folio Externo Folio Externo		

Observaciones  
Observaciones

### DETALLES

Delegación Delegación	Colonia Colonia	Calle Calle	
Número Interior Número Interior	Número Exterior Número Exterior	Entre Calle Entre Calle	Y Calle Y Calle

[Regresar](#)

Figura 4-14 solicitudD.php

Solicitudes Interesados Requisar

### MODIFICACIÓN DE SOLICITUD

<b>Interesado ID</b> 0001	<b>Materia</b> Agraria	<b>Servicio</b> Atención a falta de agua
<b>Tipo de Captación</b> Personal	<b>Folio Externo</b> 	
<b>Observaciones</b>  		

### DETALLES

<b>Delegación</b> Cauhtémoc	<b>Colonia</b> Centro	<b>Calle</b> Izazaga	
<b>Número Interior</b> 	<b>Número Exterior</b> 	<b>Entre Calle</b> 5 de Febrero	<b>Y Calle</b> Isabel la Católica

Figura 4-15 sollicitudM.php

Solicitudes Interesados Requisar

### ELIMINACIÓN DE SOLICITUD

<b>Interesado ID</b> Interesado ID	<b>Materia</b> Materia	<b>Servicio</b> Servicio
<b>Tipo de Captación</b> Tipo de Captación	<b>Folio Externo</b> Folio Externo	
<b>Observaciones</b> Observaciones		

### DETALLES

<b>Delegación</b> Delegación	<b>Colonia</b> Colonia	<b>Calle</b> Calle	
<b>Número Interior</b> Número Interior	<b>Número Exterior</b> Número Exterior	<b>Entre Calle</b> Entre Calle	<b>Y Calle</b> Y Calle

Figura 4-16 solicitudE.php

[Solicitudes](#) [Interesados](#) [Regresar](#)

### BÚSQUEDA DE SOLICITUD

Interesado ID	Materia	Servicio
<input type="text" value="0001"/>	<input type="text" value="Agraria"/>	<input type="text" value="Atención a falta de agua"/>
Tipo de Captación	Folio Externo	
<input type="text" value="Personal"/>	<input type="text"/>	
Observaciones		
<input type="text"/>		

### DETALLES

Delegación	Colonia	Calle	
<input type="text" value="Cuauhtémoc"/>	<input type="text" value="Centro"/>	<input type="text" value="Izazaga"/>	
Número Interior	Número Exterior	Entre Calle	Y Calle
<input type="text"/>	<input type="text"/>	<input type="text" value="5 de Febrero"/>	<input type="text" value="Isabel la Católica"/>

Figura 4-17 solicitudB.php

4.2.2 MÓDULO DE SEGUIMIENTO

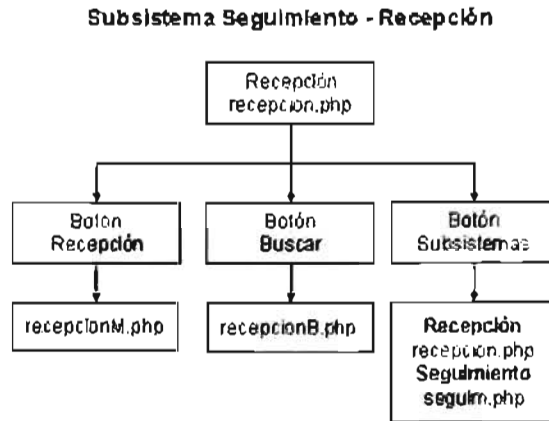


Figura 4-18

Recepción Seguimiento  Buscar

**RECEPCIÓN DE SOLICITUDES**

Año	Folio	Solicitud	Interesado	Fecha	
2004	1	Atención a falta de Aguas	Juan José	Sun 13, Jun 2004	
2004	2	Atención a falta de Aguas	Pedro	Sun 13, Jun 2004	
2004	3	Bacheo	Carlos	Sat 12, Jun 2004	
2004	4	Bacheo	Juan José	Sun 13, Jun 2004	
2004	5	Reparación de luminarias	Carolina	Sun 13, Jun 2004	
2004	6	Asesoría Jurídica	Octavio	Sun 13, Jun 2004	
2004	7	Asesoría Jurídica	Ricardo	Sun 13, Jun 2004	
2004	8	Limpieza de calles	Ricardo	Sun 13, Jun 2004	
2004	9	Limpieza de calles	Ricardo	Sun 13, Jun 2004	
2004	10	Atención a falta de Aguas	Enka	Sun 13, Jun 2004	

Página 1/1

Figura 4-19 recepcion.php

Recepción Seguimiento [Regresar](#)

### RECEPCIÓN DE SOLICITUD

<b>Interesado ID</b>	<b>Materia</b>	<b>Servicio</b>
Interesado ID	Materia	Servicio

<b>Tipo de Captación</b>	<b>Folio Externo</b>
Tipo de Captación	Folio Externo

**Observaciones**  
Observaciones

### DETALLES

<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>
Delegación	Colonia	Calle

<b>Número Interior</b>	<b>Número Exterior</b>	<b>Entre Calle</b>	<b>Y Calle</b>
Número Interior	Número Exterior	Entre Calle	Y Calle

Figura 4-20 recepcionM.php

Recepción Seguimiento [Regresar](#)

### BÚSQUEDA DE SOLICITUD EN RECEPCIÓN

<b>Interesado ID</b>	<b>Materia</b>	<b>Servicio</b>
<input type="text" value="0001"/>	<input type="text" value="Agraria"/>	<input type="text" value="Atención a falta de agua"/>

<b>Tipo de Captación</b>	<b>Folio Externo</b>
<input type="text" value="Personal"/>	<input type="text"/>

**Observaciones**

### DETALLES

<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>
<input type="text" value="Cueuhtémoc"/>	<input type="text" value="Centro"/>	<input type="text" value="Izazaga"/>

<b>Número Interior</b>	<b>Número Exterior</b>	<b>Entre Calle</b>	<b>Y Calle</b>
<input type="text"/>	<input type="text"/>	<input type="text" value="5 de Febrero"/>	<input type="text" value="Isabel la Católica"/>

Figura 4-21 recepcionB.php

**Subsistema Seguimiento - Seguimiento**

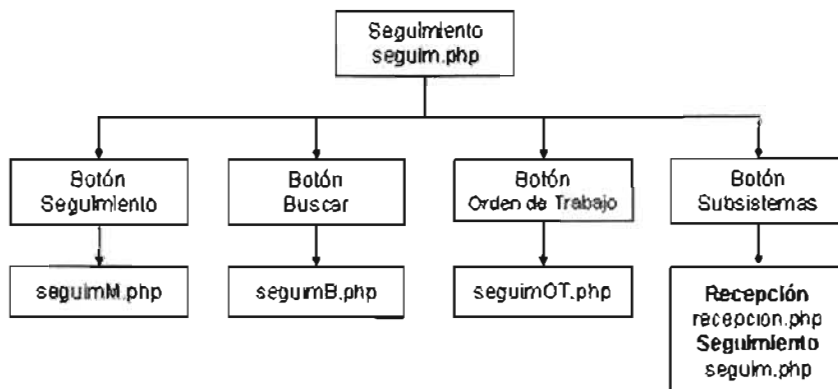


Figura 4-22

Año	Folio	Solicitud	Interesado	Fecha	
2004	1	Atención a falta de Aguas	Juan José	Sun 13, Jun 2004	
2004	2	Atención a falta de Aguas	Pedro	Sun 13, Jun 2004	
2004	3	Bacheo	Carlos	Sat 12, Jun 2004	
2004	4	Bacheo	Juan José	Sun 13, Jun 2004	
2004	5	Reparación de luminarias	Carolina	Sun 13, Jun 2004	
2004	6	Asesoría Jurídica	Octavio	Sun 13, Jun 2004	
2004	7	Asesoría Jurídica	Ricardo	Sun 13, Jun 2004	
2004	8	Limpieza de calles	Ricardo	Sun 13, Jun 2004	
2004	9	Limpieza de calles	Ricardo	Sun 13, Jun 2004	
2004	10	Atención a falta de Aguas	Enka	Sun 13, Jun 2004	

Página 1/1

Figura 4-23 seguim.php

Recepción Seguimiento [Regresar](#)

### SEGUIMIENTO DE SOLICITUD

<b>Interesado ID</b>	<b>Materia</b>	<b>Servicio</b>
Interesado ID	Materia	Servicio
<b>Tipo de Captación</b>	<b>Folio Externo</b>	<b>Estado de la Solicitud</b>
Tipo de Captación	Folio Externo	Proceso ▾

**Observaciones**

### DETALLES

<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>	
Delegación	Colonia	Calle	
<b>Número Interior</b>	<b>Número Exterior</b>	<b>Entre Calle</b>	<b>Y Calle</b>
Número Interior	Número Exterior	Entre Calle	Y Calle

Figura 4-24 seguimM.php

Recepción Seguimiento [Regresar](#)

### BÚSQUEDA DE SOLICITUD EN SEGUIMIENTO

<b>Interesado ID</b>	<b>Materia</b>	<b>Servicio</b>
0001 ▾	Agraria ▾	Atención a falta de agua ▾
<b>Tipo de Captación</b>	<b>Folio Externo</b>	
Personal ▾		

### DETALLES

<b>Delegación</b>	<b>Colonia</b>	<b>Calle</b>	
Cuauhtémoc ▾	Centro ▾	Izazaga ▾	
<b>Número Interior</b>	<b>Número Exterior</b>	<b>Entre Calle</b>	<b>Y Calle</b>
		5 de Febrero ▾	Isabel la Católica ▾

Figura 4-25 seguimB.php

4.2.3 MÓDULO DE REPORTES

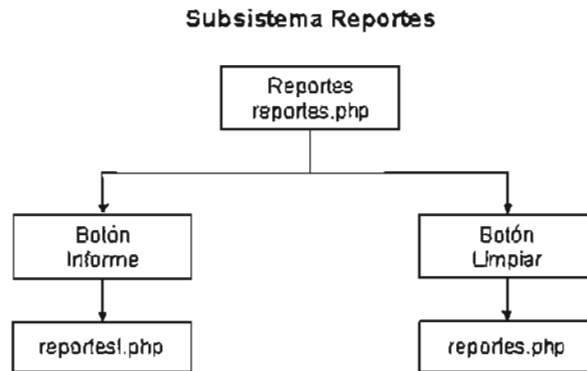


Figura 4-26

Informe Limpia

**Tipo de Reporte**  
 Área  Materias  Servicios  Global

**Fecha de Recepción**  
Del 01/01/04 al 01/01/04

**Estado**  
 Ingresado  Recibido  Seguimiento  Atendido

**Tipo de Captación**  
 Todas  Personal  Telefónica  Carta  email  Recorridos

**Ubicación**  
Colonia   
Calle

**Organización**  
Tipo   
Organización

Figura 4-27 reportes.php



4.2.4 MÓDULO DE CATÁLOGOS

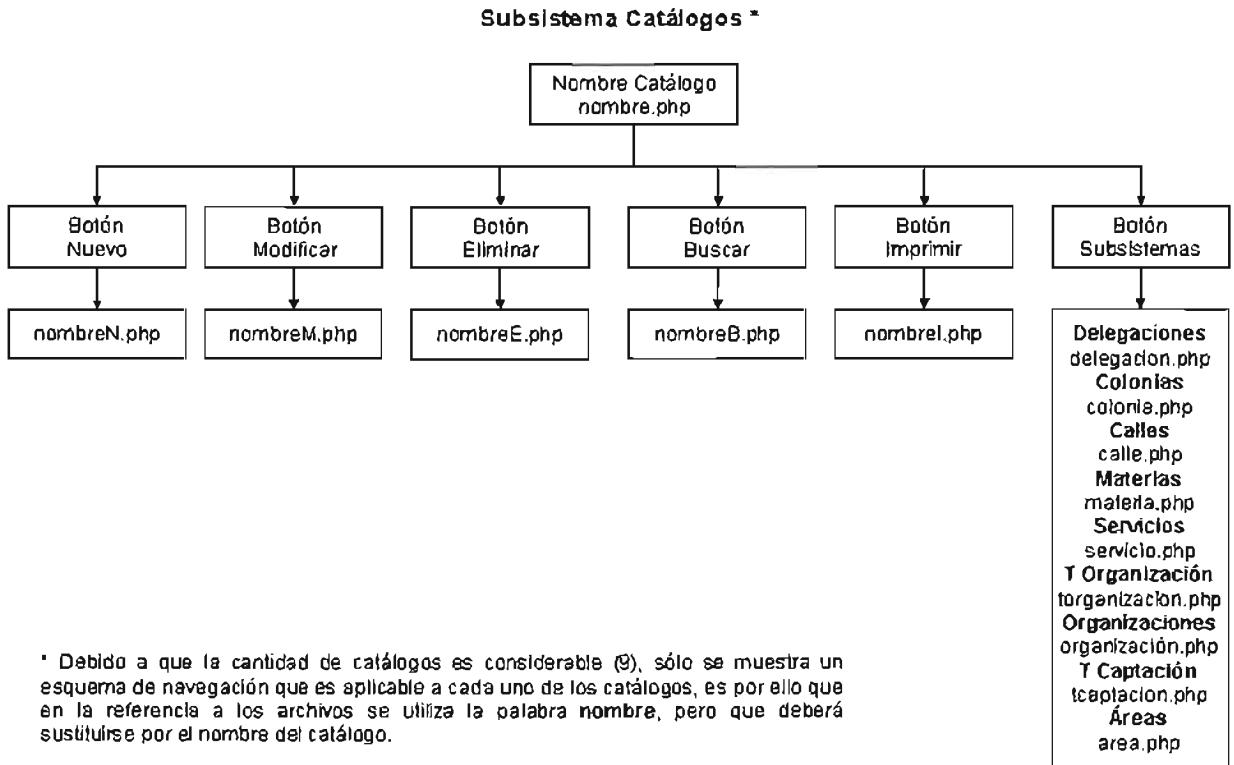


Figura 4-28



Figura 4-29 delegacion.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### REGISTRO DE DELEGACIÓN

Nombre de la Delegación

Figura 4-30 delegacionN.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### DETALLE DE DELEGACIÓN

ID	1
Nombre	Alvaro Obregón.

[Regresar](#)

Figura 4-31 delegacionD.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### MODIFICACIÓN DE DELEGACIÓN

ID	1
Nombre	<input type="text" value="Alvaro Obregón."/>

Figura 4-32 delegacionM.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### ELIMINACIÓN DE DELEGACIÓN































ID	1
Nombre	Alvaro Obregón.

Figura 4-33 delegacionE.php

Del Col Ca Mat Ser Torg Org Cap Ar Nuevo Buscar Imprimir

### COLONIAS

|< << 1 2 >> >|

ID	Colonia	CP	Delegación	
19	16 de Septiembre		Cuauhtémoc	  
15	2 de Febrero		Azcapotzalco	  
3	Barranca Alegre	01080	Alvaro Obregón.	  
14	Barranca Roja		Alvaro Obregón.	  
21	Barrio Santiago		Xochimilco	  
4	Centro	06080	Cuauhtémoc	  
20	Constitucion 1917		Iztapalapa	  
18	Jalapa		Cuajimalpa de Morelos	  
5	Pantitlán	08200	Iztacalco	  
16	Parque Escondido		Benito Juárez	  

Página 1/2

Figura 4-34 colonia.php

Del Col Ca Mat Ser Torg Org Cap Ar [Regresar](#)

### REGISTRO DE COLONIA

Delegación

Nombre de la Colonia

C.P.

Figura 4-35 coloniaN.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**DETALLE DE LA COLONIA**

ID	19
Nombre	16 de Septiembre
C.P.	
Delegación	Cuauhtémoc

[Regresar](#)

Figura 4-36 coloniaD.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**MODIFICACIÓN DE LA COLONIA**

ID	19
Nombre	<input type="text" value="16 de Septiembre"/>
C.P.	<input type="text"/>
Delegación	<input type="text" value="Cuauhtémoc"/>

[Guardar](#)

Figura 4-37 coloniaM.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**ELIMINACIÓN DE LA COLONIA**

ID	5
Nombre	Panitlán
C.P.	08200
Delegación	Iztacalco

[Eliminar](#)





















Figura 4-38 coloniaE.php

ESTA TESIS NO SALE  
DE LA BIBLIOTECA

Del Col Ca Mat Ser Torg Org Cap Ar Nuevo Buscar Imprimir

### CALLES

1 < << 1 2 >> >1

ID	Calle	Colonia	Delegación	
6	Calle 1	Barranca Alegre	Alvaro Obregón.	 
8	Calle 1	Pantitlán	Iztacalco	 
5	Calle 2	Barranca Alegre	Alvaro Obregón.	 
9	Calle 2	Pantitlán	Iztacalco	 
7	Calle 20	Barranca Alegre	Alvaro Obregón.	 
4	Calle 3	Barranca Alegre	Alvaro Obregón.	 
10	Calle 3	Pantitlán	Iztacalco	 
3	Calle 4	Barranca Alegre	Alvaro Obregón.	 
11	Calle 4	Pantitlán	Iztacalco	 
2	Calle 5	Barranca Alegre	Alvaro Obregón.	 

Página 1/2

Figura 4-39 calle.php

Del Col Ca Mat Ser Torg Org Cap Ar Registrar

### REGISTRO DE CALLE

Delegación

Colonia

Nombre de la Calle

Figura 4-40 calleN.php

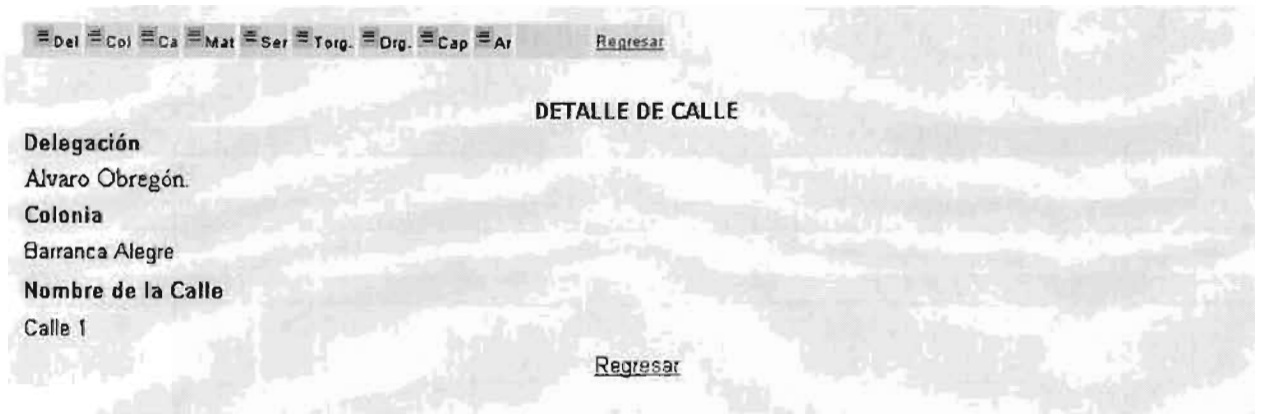


Figura 4-41 calleD.php

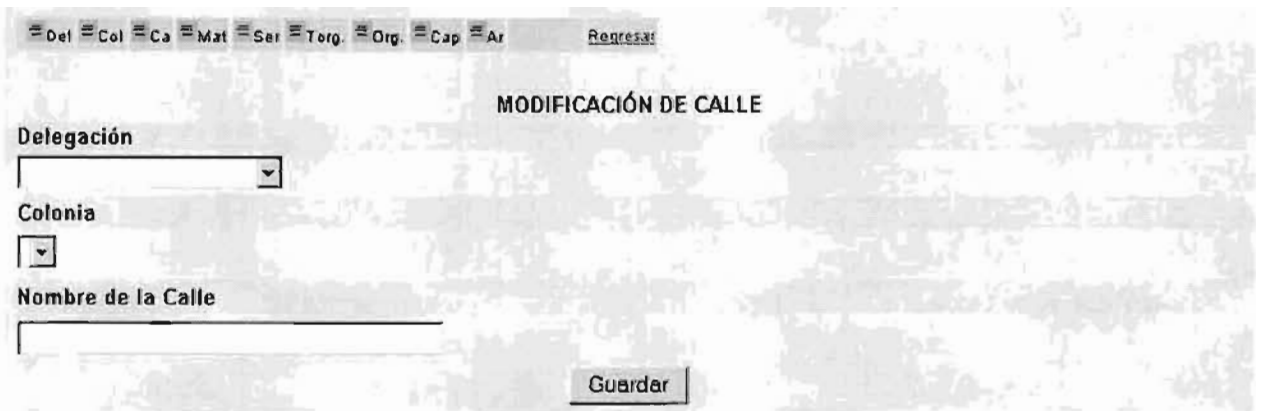


Figura 4-42 calleM.php

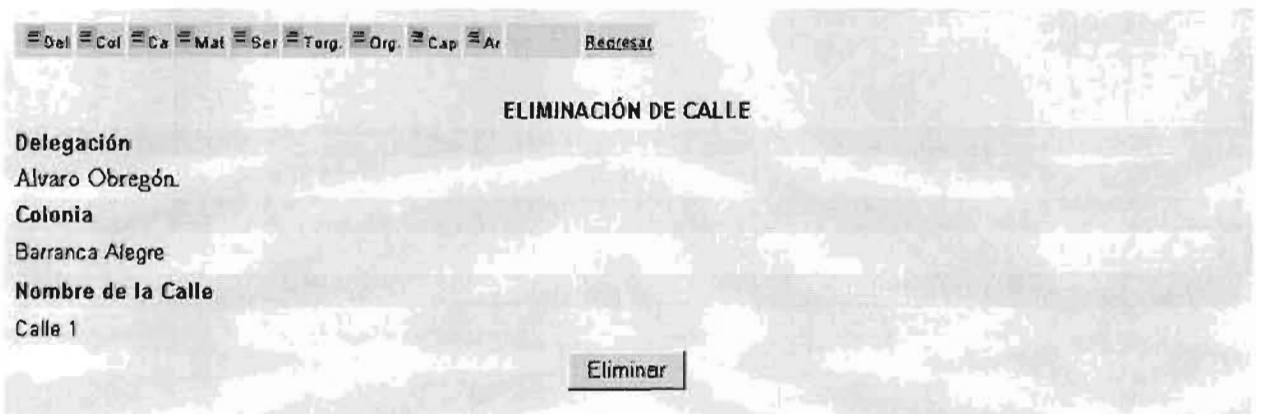


Figura 4-43 calleE.php

ID	Materia	
3	Agraria 1	
1	Agua y Servicios Hidráulicos	
4	Asesoría Telefónica	
5	Asesoría y Asistencia Social	
6	Industria	
7	Medio Ambiente	
8	Registro Público	
9	Salud	
10	Servicios Legales y Archivo de Notarias	
2	Servicios Urbanos	

Página 1/2

Figura 4-44 materia.php

Nombre de la Materia

Guardar

Figura 4-45 materiaN.php

ID	3
Nombre	Agraria 1
Desactivar	

Regresar

Figura 4-46 materiaD.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### MODIFICACIÓN DE LA MATERIA

**ID** 1  
**Nombre**   
**Desactivar**

[Guardar](#)

Figura 4-47 materiaM.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

### ELIMINACIÓN DE LA MATERIA













**ID** 4  
**Nombre** Asesoría Telefónica  
**Desactivar**

[Eliminar](#)

Figura 4-48 materiaE.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Nuevo](#) [Buscar](#) [Imprimir](#)

### SERVICIOS

ID	Servicio	Materia	
3	a	Agraria1	 
6	Asesoría Jurídica	Asesoría y Asistencia Social	 
2	Atención a falta de Aguas	Agua y Servicios Hidráulicos	 
4	Bacheo	Servicios Urbanos	 
7	Limpieza de calles	Servicios Urbanos	 
5	Reparación de luminarias	Servicios Urbanos	 

Página 1/1

Figura 4-49 servicio.php



Def Col Ca Mat Ser Torg Org Cap Ar [Regresar](#)

### REGISTRO DE SERVICIO

Materia  Área que Atiende el Servicio

Servicio

Descripción

Usuarios

Tiempo de Respuesta

Requisitos

Beneficios

Fundamento Jurídico

Observaciones

Oficinas









Costo

Área de pago

Figura 4-50 servicioN.php

Del Col Ca Mat Ser Torg. Org. Cap Ar Nuevo Buscar Imprimir

### TIPO DE ORGANIZACIONES

ID	Tipo de Organización	
7	Asociaciones Civiles	 
11	Diputados Federales	 
9	Diputados Locales	 
5	Partidos Politicos	 

Página 1/1

Figura 4-51 torganizacion.php

Del Col Ca Mat Ser Torg. Org. Cap Ar [Regresar](#)

### REGISTRO DE TIPO DE ORGANIZACIÓN

Nombre del Tipo de Organización

Figura 4-52 torganizacionN.php

Del Col Ca Mat Ser Torg. Org. Cap Ar [Regresar](#)

### DETALLE DEL TIPO DE ORGANIZACIÓN

ID	7
Nombre	Asociaciones Civiles

[Regresar](#)

Figura 4-53 torganizacionD.php

≡ Del ≡ Col ≡ Ca ≡ Mat ≡ Ser ≡ Torg. ≡ Org. ≡ Cap ≡ Ar [Regresar](#)

**MODIFICACIÓN DEL TIPO DE ORGANIZACIÓN**

ID            7

Nombre     

Figura 4-54 torganizacionM.php

≡ Del ≡ Col ≡ Ca ≡ Mat ≡ Ser ≡ Torg. ≡ Org. ≡ Cap ≡ Ar [Regresar](#)

**ELIMINACIÓN DEL TIPO DE ORGANIZACIÓN**

ID            7

Nombre      Asociaciones Civiles

Figura 4-55 torganizacionE.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg](#) [Org](#) [Cap](#) [Ar](#)

[Nuevo](#) [Buscar](#) [Imprimir](#)

### ORGANIZACIONES

ID	Organización	Tipo de Organización	
7	Asociación de Colonos Cuauhtémoc	Asociaciones Civiles	
6	Oficina Diputado XL	Diputados Federales	
5	Partido Comunista	Partidos Políticos	
8	Partido de la revolución Democrática	Partidos Políticos	
4	ProVida	Asociaciones Civiles	<a href="#">Eliminar</a>

Página 1/1

Figura 4-56 organizacion.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg](#) [Org](#) [Cap](#) [Ar](#)

[Regresar](#)

### REGISTRO DE ORGANIZACIÓN

Tipo de Organización

Nombre de la Organización

[Guardar](#)

Figura 4-57 organizacionN.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg](#) [Org](#) [Cap](#) [Ar](#)

[Regresar](#)

### DETALLE DE LA ORGANIZACIÓN

ID	7
Nombre	Asociación de Colonos Cuauhtémoc
Tipo	Asociaciones Civiles

[Regresar](#)

Figura 4-58 organizacionD.php

The screenshot shows a web interface for modifying an organization. At the top, there is a navigation menu with links: Del, Col, Ca, Mat, Ser, Torg., Org., Cap, Ar, and a 'Regresar' button. The main heading is 'MODIFICACIÓN DE LA ORGANIZACIÓN'. Below this, there are three input fields: 'ID' with the value '7', 'Nombre' with the value 'Asociación de Colonos Cuauhtémoc', and 'Tipo' with a dropdown menu showing 'Asociaciones Cíviles'. A 'Guardar' button is located at the bottom right of the form.

Figura 4-59 organizacionM.php

The screenshot shows a web interface for deleting an organization. At the top, there is a navigation menu with links: Del, Col, Ca, Mat, Ser, Torg., Org., Cap, Ar, and a 'Regresar' button. The main heading is 'ELIMINACIÓN DE LA ORGANIZACIÓN'. Below this, there are three input fields: 'ID' with the value '7', 'Nombre' with the value 'Asociación de Colonos Cuauhtémoc', and 'Tipo' with the value 'Asociaciones Cíviles'. An 'Eliminar' button is located at the bottom right of the form.

Figura 4-60 organizacionE.php

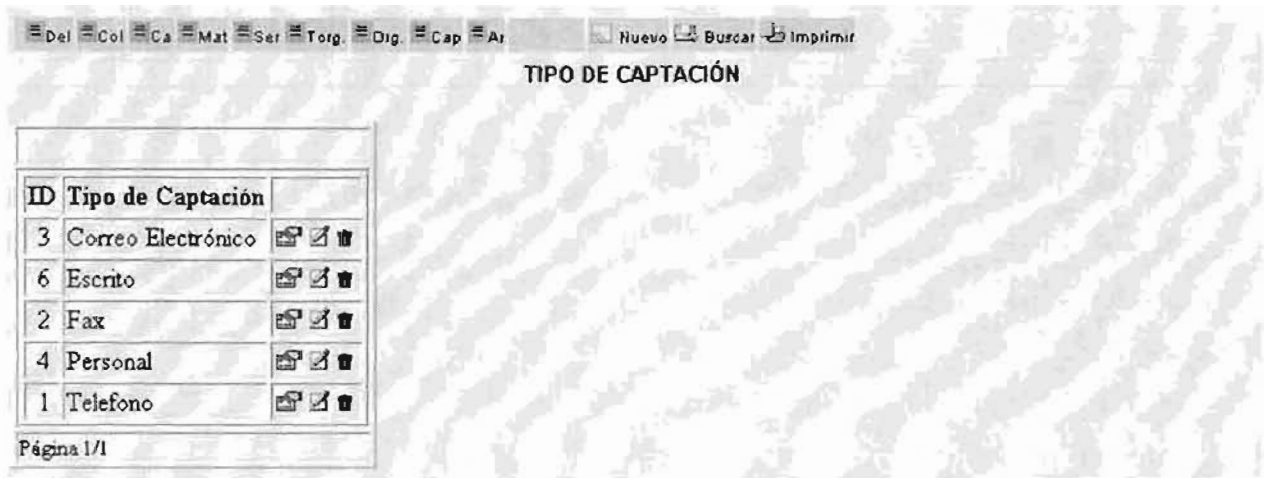


Figura 4-61 captacion.php



Figura 4-62 captacionN.php



Figura 4-63 captacionD.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**MODIFICACIÓN DEL TIPO DE CAPTACIÓN**

ID 3

Nombre

Figura 4-64 captacionM.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**ELIMINACIÓN DEL TIPO DE CAPTACIÓN**

ID 3

Nombre Correo Electrónico

Figura 4-65 captacionE.php

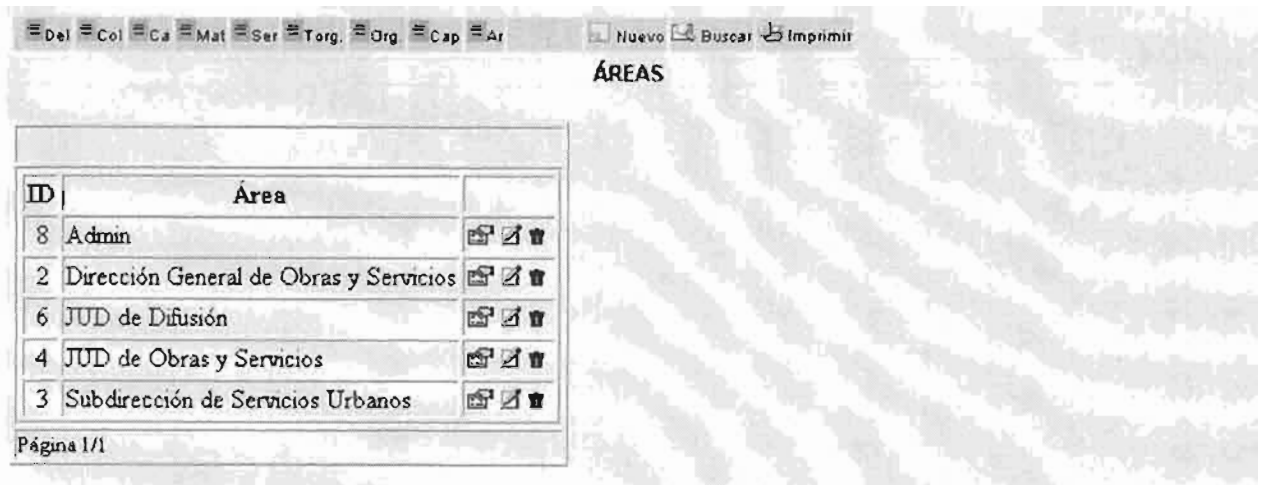


Figura 4-66 area.php



Figura 4-67 areaN.php



Figura 4-68 areaD.php



[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**MODIFICACIÓN DEL ÁREA**

ID 8

Nombre

Figura 4-69 areaM.php

[Del](#) [Col](#) [Ca](#) [Mat](#) [Ser](#) [Torg.](#) [Org.](#) [Cap](#) [Ar](#) [Regresar](#)

**ELIMINACIÓN DEL ÁREA**

ID 2

Nombre Dirección General de Obras y Servicios

Figura 4-70 areaE.php

4.2.5 MÓDULO DE ADMINISTRACIÓN

Subsistema Administración - Usuarios

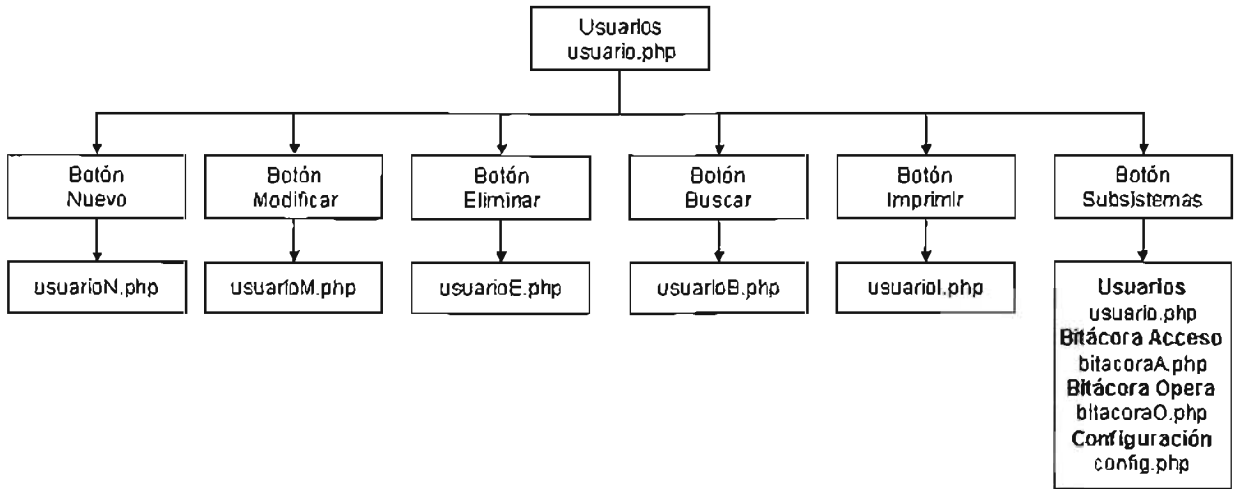


Figura 4-71

ID	Usuario	Nombre	Paterno	Materno	Área	
1	admin	Admin	-	-	Admin	
7	adiaz	Alfonso	Díaz	Farías	JUD de Obras y Servicios	
3	erios	Edgar	Ríos	Linares	Dirección General de Obras y Servicios	
4	fgarcia	Filiberto	García	Domínguez	JUD de Obras y Servicios	
11	hpotter	Harry	Potter		JUD de Difusión	
8	hparedes	Herctor	Paredes	Guerrero	JUD de Obras y Servicios	
5	jromero	José	Romero	Pérez	Dirección General de Obras y Servicios	
9	arellano	José María	Arellano	Mora	Subdirección de Servicios Urbanos	
6	pmoreno	Pablo	Moreno	López	Subdirección de Servicios Urbanos	
2	usuario	Usuario	-	-	JUD de Difusión	

Página 1/2

Figura 4-72 usuario.php

[Usuarios](#) | [Bitacora A](#) | [Bitacora O](#) | [Config](#) | [Regresar](#)

### REGISTRO DE USUARIO

Nombre	Apellido Paterno	Apellido Materno
<input type="text"/>	<input type="text"/>	<input type="text"/>
Usuario	Contraseña	Área
<input type="text"/>	<input type="text"/>	<input type="text"/>
		Activo <input checked="" type="checkbox"/>
<input type="button" value="Guardar"/>		

Figura 4-73 usuarioN.php

[Usuarios](#) | [Bitacora A](#) | [Bitacora O](#) | [Config](#) | [Regresar](#)

### DETALLE DE USUARIO

ID	Nombre	Apellido Paterno	Apellido Materno
7	Alfonso	Diaz	Fariás
Usuario	Contraseña	Área	Activo
adiaz	adiaz	JUD de Obras y Servicios	<input checked="" type="checkbox"/>
<a href="#">Regresar</a>			

Figura 4-74 usuarioD.php

[Usuarios](#) | [Bitacora A](#) | [Bitacora O](#) | [Config](#) | [Regresar](#)

### MODIFICACIÓN DE USUARIO

ID	Nombre	Apellido Paterno	Apellido Materno
1	<input type="text" value="Admin"/>	<input type="text"/>	<input type="text"/>
Usuario	Contraseña	Área	Activo
admin	admin	<input type="text" value="Admin"/>	<input checked="" type="checkbox"/>
<input type="button" value="Guardar"/>			

Figura 4-75 usuarioM.php

[Usuarios](#) | [Bitacora A](#) | [Bitacora O](#) | [Config](#) | [Regresar](#)

### ELIMINACIÓN DE USUARIO

ID	Nombre	Apellido Paterno	Apellido Materno
9	José María	Arellano	Mora
Usuario	Contraseña	Área	Activo
arellano	arellano	Subdirección de Servicios Urbanos	<input checked="" type="checkbox"/>
<input type="button" value="Eliminar"/>			

Figura 4-76 usuarioE.php

Subsistema Administración – Bitácora de Accesos

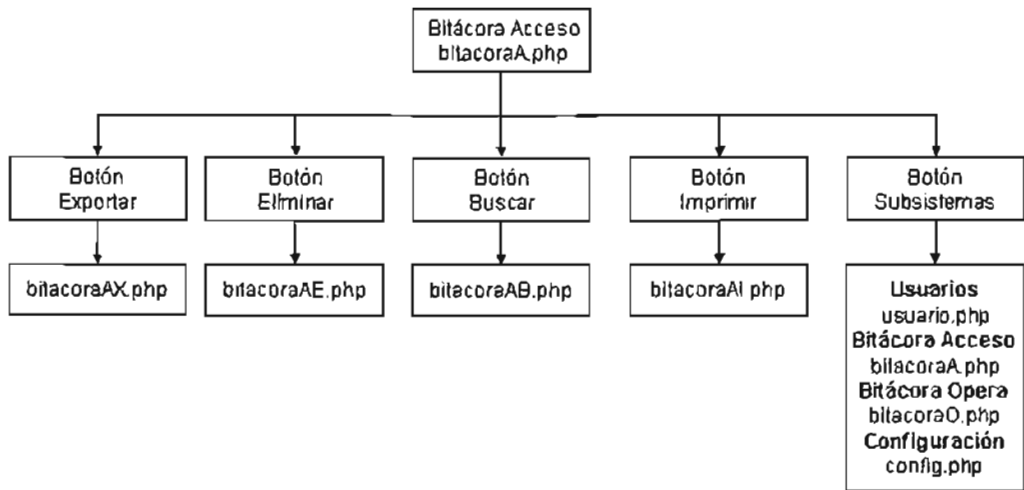


Figura 4-77

ID	Usuario	Fecha Entrada	Hora Entrada	Fecha Salida	Hora Salida	
1	admin	Sun 13, Jun 2004	12:17:00	Sun 13, Jun 2004	12:20:00	
2	usuario	Sun 13, Jun 2004	12:17:00	Sun 13, Jun 2004	13:10:00	
3	erios	Sun 13, Jun 2004	13:17:00	Sun 13, Jun 2004	15:11:00	
4	fgarcia	Sun 13, Jun 2004	14:00:00	Sun 13, Jun 2004	14:10:00	
5	jromero	Sun 13, Jun 2004	19:00:00	Sun 13, Jun 2004	19:09:00	

Página 1/1

Figura 4-78 bitacoraA.php

Subsistema Administración – Bitácora de Operaciones

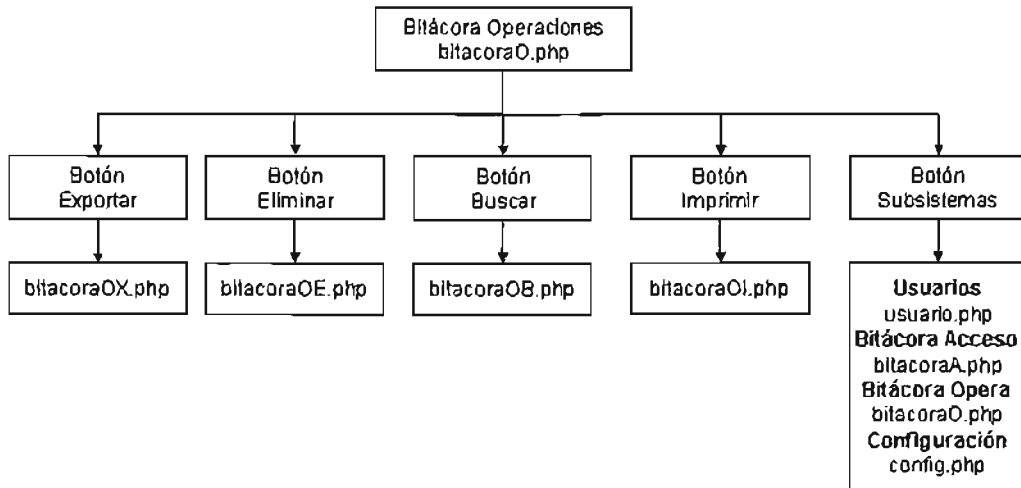


Figura 4-79

ID	Usuario	Operación	Fecha	Hora	Folio	Tabla	
1	admin	Insertar	Sat 12, Jun 2004	19:00:00	1	delegacion	
3	erios	Insertar	Sat 12, Jun 2004	19:59:00	2	calle	
2	usuario	Consultar	Sat 12, Jun 2004	12:00:00	2	colonia	
5	romero	Consultar	Sat 12, Jun 2004	20:01:00	2	captacion	
4	fgarcia	Modificar	Sat 12, Jun 2004	20:00:00	2	captacion	

Página 1/1

Figura 4-80 bitacoraO.php

Subsistema Administración – Configuración

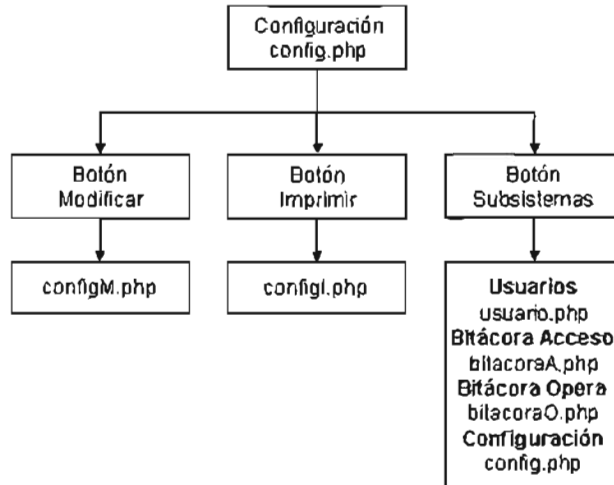


Figura 4-81

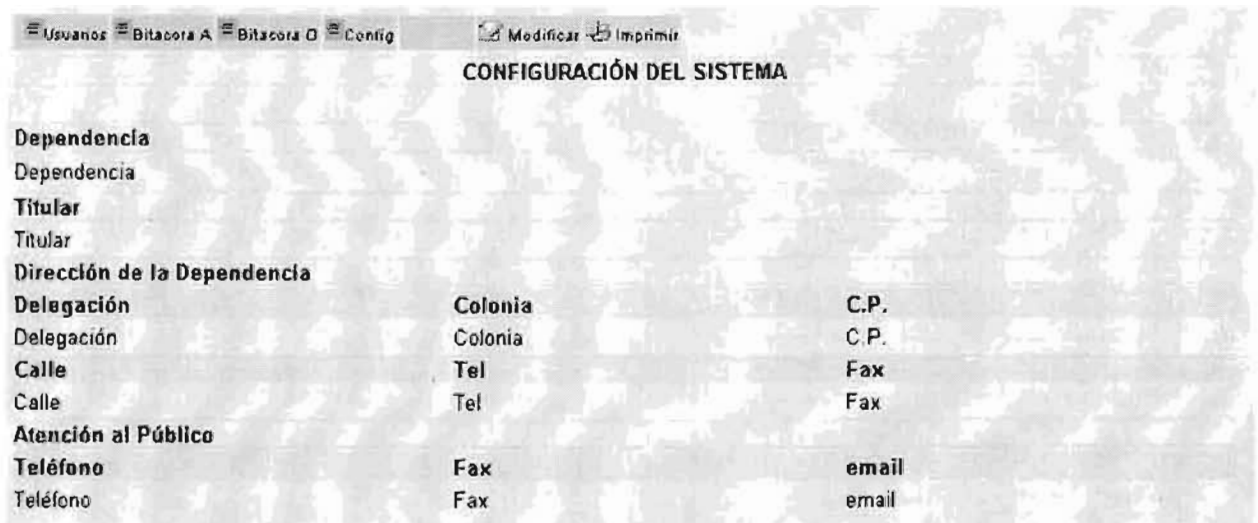


Figura 4-82 config.php

[Usuarios](#) [Bitacora A](#) [Bitacora D](#) [Config](#) [Regresar](#)

### MODIFICACIÓN DE LA CONFIGURACIÓN DEL SISTEMA

**Dependencia**

**Titular**

**Dirección de la Dependencia**

<b>Delegación</b> <input type="text"/>	<b>Colonia</b> <input type="text"/>	<b>C.P.</b> <input type="text"/>
<b>Calle</b> <input type="text"/>	<b>Tel</b> <input type="text"/>	<b>Fax</b> <input type="text"/>
<b>Atención al Público</b>	<b>Fax</b>	<b>email</b>
<b>Teléfono</b> <input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 4-83 configM.php

#### 4.2.6 MÓDULO DE ORIENTACIÓN

Tablas:

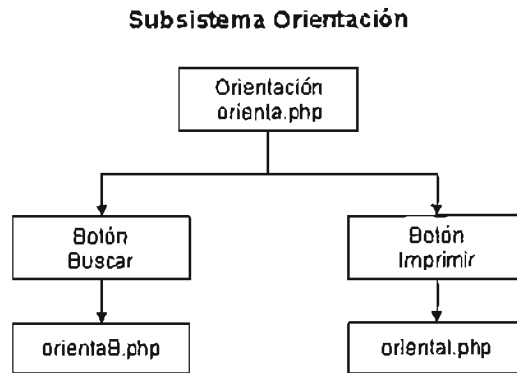


Figura 4-84



Figura 4-85 orientat.php



## TEMA 5 DISEÑO DE LA BASE DE DATOS

### 5.1 DISEÑO DE LA BASE DE DATOS

La propuesta de la base de datos a partir de los requerimientos obtenidos se muestra en la Figura 5-1.

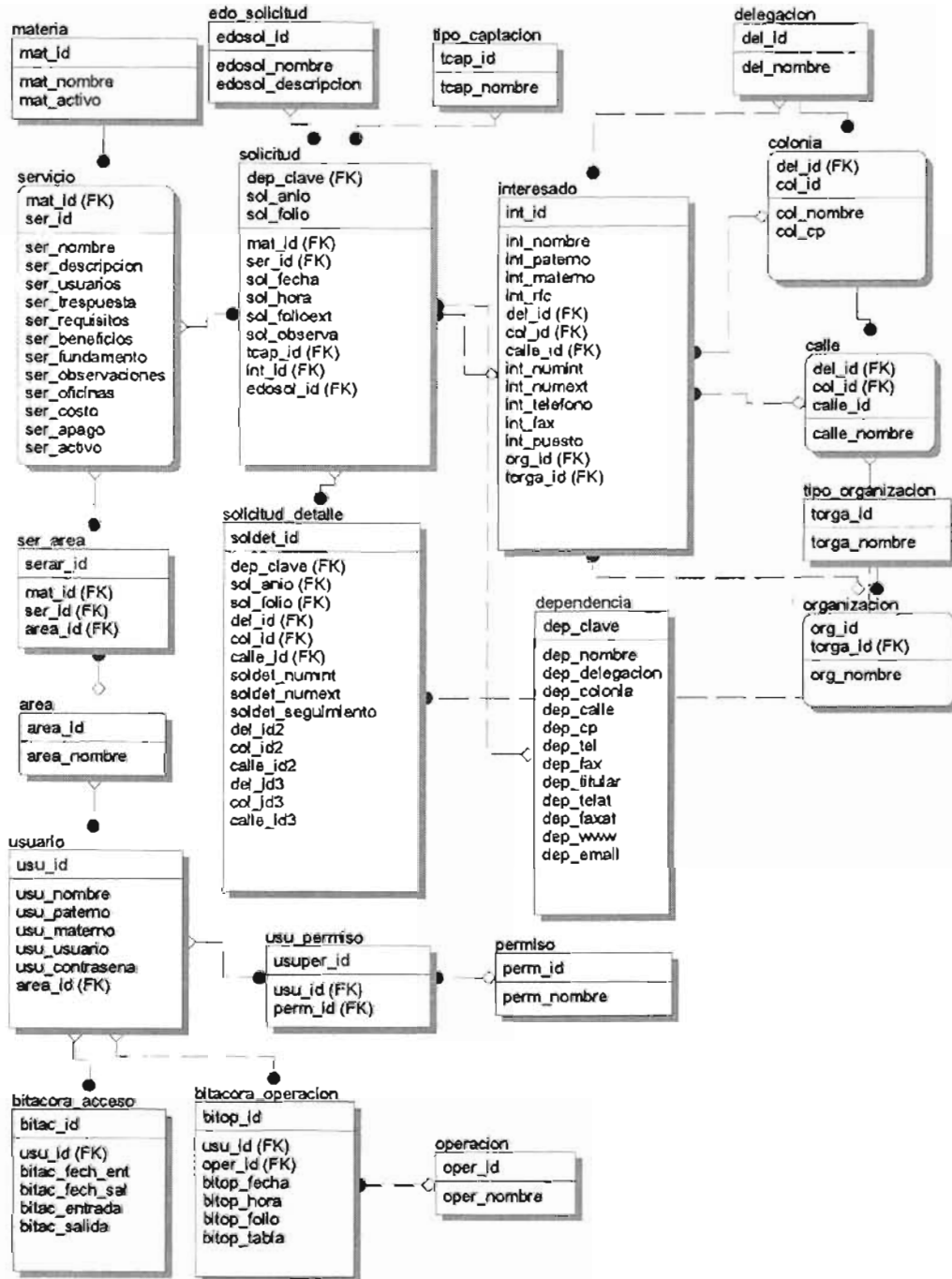


Figura 5-1

5.2 DICCIONARIO DE DATOS

<b>Tabla:</b>	<b>delegacion</b>				
<b>Descripción:</b>	Catálogo de las Delegaciones del Distrito Federal, ejemplo: Coyoacán, Cuauhtémoc, Iztacalco, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
del_id	Clave de la Delegación	INT		Auto increment	Requerido
del_nombre	Nombre de la Delegación	CHAR	25		Requerido

<b>Tabla:</b>	<b>colonia</b>				
<b>Descripción:</b>	Catálogo de Colonias por Delegación del D.F.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
del_id	Clave de la Delegación	INT			Requerido
col_id	Clave de la Colonia	INT		Auto increment	Requerido
col_nombre	Nombre de la Colonia	CHAR	40		Requerido
col_cp	Código Postal	CHAR	5		Requerido
<b>Tablas Relacionadas</b>					
delegacion					

<b>Tabla:</b>	<b>calle</b>				
<b>Descripción:</b>	Catálogo de Calles por Colonia y Delegación del D.F.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
del_id	Clave de la Delegación	INT			Requerido
col_id	Clave de la Colonia	INT			Requerido
calle_id	Clave de la Calle	INT		Auto increment	Requerido
calle_nombre	Nombre de la calle	CHAR	40		Requerido
<b>Tablas Relacionadas</b>					
delegacion colonia					

<b>Tabla:</b>	<b>tipo_organizacion</b>				
<b>Descripción:</b>	Catálogo de los tipos de organizaciones, ejemplo: Partidos Políticos, Senadores, Asociaciones Civiles, Asamblea D.F., etc				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
torga_id	Clave del tipo de organización	INT		Auto increment	Requerido
torga_nombre	Nombre del tipo de organización	CHAR	80		Requerido

<b>Tabla:</b>	<b>organizacion</b>				
<b>Descripción:</b>	Catálogo de las organizaciones a las que puede pertenecer el interesado, ejemplo: Ciudadano, PRD, PRI, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
torga_id	Clave del tipo de organización	INT			Requerido
org_id	Clave de la Organización	INT		Auto increment	Requerido
org_nombre	Nombre de la Organización	CHAR	80		Requerido
<b>Tablas Relacionadas</b>					
tipo_organizacion					

<b>Tabla:</b>	<b>tipo_captacion</b>				
<b>Descripción:</b>	Catálogo de las distintas formas en que se puede recibir una solicitud: correo, email, teléfono, personal, fax, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
tcap_id	Clave del tipo de Captación	INT		Auto increment	Requerido
tcap_nombre	Nombre del tipo de Captación	CHAR	25		Requerido

<b>Tabla:</b>	<b>edo_solicitud</b>				
<b>Descripción:</b>	Catálogo de los distintos estados que puede tener una solicitud, ejemplo: registrada, seguimiento, atendida, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
edosol_id	Clave del Estado	INT		Auto increment	Requerido
edosol_nombre	Nombre del Estado	VARCHAR	15		Requerido
edosol_descripcion	Descripción del Estado	VARCHAR	255		

<b>Tabla:</b>	<b>area</b>				
<b>Descripción:</b>	Catálogo de las áreas que atenderán cada uno de las solicitudes, ejemplo: Obras, Servicios Urbanos, Desasolve, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
area_id	Clave del Área	INT		Auto increment	Requerido
area_nombre	Nombre del Área	VARCHAR	40		Requerido

<b>Tabla:</b>	<b>materia</b>				
<b>Descripción:</b>	Catálogos de la materias en que agrupan los distintos servicios, ejemplo: Agua, Salud, Servicios Urbano, Asistencia Social, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
mat_id	Clave de la materia	INT		Auto increment	Requerido
mat_nombre	Nombre de la materia	CHAR	40		Requerido
mat_activo	Estado de la materia, activado-desactivado	BOOLEAN			

<b>Tabla:</b>	<b>servicio</b>				
<b>Descripción:</b>	Catálogos de los Servicios por materias, ejemplo: Atención a faltas de agua, bacheo, desasolve, etc.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
mat_id	Clave de la materia	INT			Requerido
ser_id	Clave del servicio	INT		Auto increment	Requerido
ser_nombre	Nombre del servicio	CHAR	50		Requerido
ser_descripcion	Descripción del servicio	CHAR	80		
ser_usuarios	Usuarios del servicio	CHAR	255		
ser_trespuesta	Tiempo de respuesta del servicio	CHAR	18		
ser_requisitos	Requisitos del servicio	TEXT			
ser_beneficios	Beneficios del servicio	TEXT			
ser_fundamento	Fundamento jurídico del servicio	TEXT			
ser_observaciones	Observaciones del servicio	TEXT			
ser_oficinas	Oficinas de gestión del servicio	TEXT			
ser_costo	Costo del servicio	CHAR	255		
ser_apago	Área de pago del servicio	CHAR	255		
ser_activo	Estado del servicio, activado-desactivado	BOOLEAN			
<b>Tablas Relacionadas</b>					
materia					

<b>Tabla:</b>	<b>ser_area</b>				
<b>Descripción:</b>	Tabla que relaciona los servicios con cada una de las áreas que será la responsable de darle seguimiento, ejemplo: El servicio de Desasolve lo atiende el área Servicios Urbanos.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
serar_id	Clave del servicio – área	INT		Auto increment	Requerido
mat_id	Clave de la materia	INT			Requerido
ser_id	Clave del servicio	INT			Requerido
area_id	Clave del Área	INT			Requerido
<b>Tablas Relacionadas</b>					
materia    servicio    area					

<b>Tabla:</b>	<b>usuario</b>				
<b>Descripción:</b>	Catálogos de los usuarios del sistema				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
usu_id	Clave del Usuario	INT		Auto increment	Requerido
usu_nombre	Nombre del Usuario	VARCHAR	30		Requerido
usu_paterno	Apellido paterno del Usuario	VARCHAR	30		Requerido
usu_materno	Apellido materno del Usuario	VARCHAR	30		Requerido
usu_usuario	Alias de acceso del Usuario	CHAR	8		Requerido
usu_contrasena	Clave de acceso del Usuario	CHAR	8	Password	Requerido
area_id	Clave del Área a la que pertenece	INT			
<b>Tablas Relacionadas</b>					
area					

<b>Tabla:</b>	<b>operacion</b>				
<b>Descripción:</b>	Catálogo de los tipos de operaciones que se efectúan en el sistema, ejemplo: Registro, modificación, eliminación, consulta, impresión.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
oper_id	Clave de la operación	INT		Auto increment	Requerido
oper_nombre	Nombre del Usuario	VARCHAR	10		Requerido

<b>Tabla:</b>	<b>bitacora_acceso</b>				
<b>Descripción:</b>	Bitácora de acceso al sistema de cada uno de los usuario				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
bitac_id	Clave del Registro de Acceso al Sistema	INT		Auto increment	Requerido
usu_id	Clave del usuario	INT			Requerido
bitac_fech_ent	Fecha de Entrada	DATE			Requerido
bitac_fech_sal	Fecha de Salida	DATE			Requerido
bitac_entrada	Hora de Entrada	TIME			Requerido
bitac_salida	Hora de Salida	TIME			Requerido
<b>Tablas Relacionadas</b>					
usuario					

<b>Tabla:</b>	<b>bitacora_operacion</b>				
<b>Descripción:</b>	Bitácora de las acciones realizadas en el sistema de cada uno de los usuario				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
bitop_id	Clave de la Operación en el Sistema	INT		Auto increment	Requerido
usu_id	Clave del usuario	INT			Requerido
oper_id	Clave de la operación	INT			Requerido
bitop_fecha	Fecha de la operación	DATE			Requerido
bitop_hora	Hora de la operación	TIME			Requerido
bitop_folio	Folio en donde se realizó la operación	CHAR	12		Requerido
bitop_tabla	Tabla de la operación	CHAR	30		Requerido
<b>Tablas Relacionadas</b>					
usuario operacion					

<b>Tabla:</b>	<b>permiso</b>				
<b>Descripción:</b>	Catálogo de los permisos para cada módulo del sistema, así como de los submódulos y de las acciones permitidas para cada uno de ellos, ejemplo: 2000 – Permiso para el módulo de Seguimiento				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
perm_id	Clave del permiso	INT			Requerido
perm_nombre	Nombre del permiso dentro del Sistema	VARCHAR	80		Requerido

<b>Tabla:</b>	<b>usu_permiso</b>				
<b>Descripción:</b>	Tabla que relaciona los usuarios con los permisos que tienen cada uno de ellos dentro del sistema, ejemplo: El usuario Ejecutor tiene el Permiso para acceder al módulo de Seguimiento.				
<b>Campos</b>					
Nombre	Descripción	Tipo	Tamaño	Formato	Validación
usuper_id	Clave del permiso para usuario	INT		Auto increment	Requerido
usu_id	Clave del usuario	INT			Requerido
perm_id	Clave del permiso	INT			Requerido
<b>Tablas Relacionadas</b>					
permiso usuario					

<b>Tabla:</b>	<b>solicitud</b>				
<b>Descripción:</b>	Tabla principal de las solicitudes presentadas por los ciudadanos, por folio				
<b>Campos</b>					
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Formato</b>	<b>Validación</b>
dep_clave	Clave de la Dependencia	INT			Requerido
sol_anio	Año de la solicitud	CHAR	4		Requerido
sol_folio	Folio de la solicitud	CHAR	6		Requerido
mat_id	Clave de la materia	INT			Requerido
ser_id	Clave del servicio	INT			Requerido
sol_fecha	Fecha de la solicitud	DATE			Requerido
sol_hora	Hora de la solicitud	TIME			Requerido
sol_folioext	Folio externo de la solicitud	CHAR	18		
sol_observa	Observaciones de la solicitud	TEXT			
tcap_id	Clave del tipo de Captación	INT			Requerido
int_id	Clave del Interesado	INT			Requerido
edosol_id	Clave del Estado	INT			Requerido
<b>Tablas Relacionadas</b>					
dependencia tipo_captacion interesado edo_solicitud					



<b>Tabla:</b>	<b>interesado</b>				
<b>Descripción:</b>	Catálogo de los interesados que presentan solicitudes, ejemplo: Pedro Páramo Rulfo				
<b>Campos</b>					
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Formato</b>	<b>Validación</b>
int_id	Clave del Interesado	INT		Auto increment	Requerido
int_nombre	Nombre del Interesado	VARCHAR	30		Requerido
int_paterno	Apellido paterno del Interesado	VARCHAR	30		Requerido
int_materno	Apellido materno del Interesado	VARCHAR	30		Requerido
int_rfc	RFC del Interesado	VARCHAR	15		
del_id	Clave de la Delegación Dirección	INT			Requerido
col_id	Clave de la Colonia Dirección	INT			Requerido
calle_id	Clave de la Calle Dirección	INT			Requerido
int_numint	Número Interior del Interesado	VARCHAR	10		
int_numext	Número Exterior del Interesado	VARCHAR	10		
int_telefono	Teléfono del Interesado	VARCHAR	15		
int_fax	Fax del Interesado	VARCHAR	15		
int_puesto	Puesto del Interesado	VARCHAR	30		
torga_id	Clave del tipo de organización	INT			Requerido
org_id	Clave de la Organización	INT			Requerido
<b>Tablas Relacionadas</b>					
delegacion colonia calle tipo_organizacion organizacion					

<b>Tabla:</b>	<b>solicitud_detalle</b>				
<b>Descripción:</b>	Tabla que contiene los detalles de cada una de las solicitudes, lugar de ubicación de la solicitud de servicio				
<b>Campos</b>					
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Formato</b>	<b>Validación</b>
soldet_id	Clave del detalle de la solicitud	INT		Auto increment	Requerido
dep_clave	Clave de la Dependencia	INT			Requerido
sol_anio	Año de la solicitud	CHAR	4		Requerido
sol_folio	Folio de la solicitud	CHAR	6		Requerido
del_id	Clave de la Delegación Dirección	INT			Requerido
col_id	Clave de la Colonia Dirección	INT			Requerido
calle_id	Clave de la Calle Dirección	INT			Requerido
soldet_numext	Número Exterior de la Solicitud	VARCHAR	10		
soldet_numint	Número Interior de la Solicitud	VARCHAR	10		
del_id	Clave de la Delegación (entre calle)	INT			
col_id	Clave de la Colonia (entre calle)	INT			
calle_id	Clave de la Calle (entre calle)	INT			
del_id	Clave de la Delegación (y calle)	INT			
col_id	Clave de la Colonia (y calle)	INT			
calle_id	Clave de la Calle (y calle)	INT			
soldet_seguimiento	Seguimiento de la Solicitud (acciones)	TEXT			
<b>Tablas Relacionadas</b>					
solicitud delegacion colonia calle					

<b>Tabla:</b>	<b>dependencia</b>				
<b>Descripción:</b>	Tabla que contiene la configuración (los datos generales) de la Dependencia donde operará el sistema, ejemplo: clave, nombre, titular, dirección, etc, éstos datos se reflejarán en los reportes.				
<b>Campos</b>					
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Formato</b>	<b>Validación</b>
dep_clave	Clave de la Dependencia	INT		Auto increment	Requerido
dep_nombre	Nombre de la Dependencia	VARCHAR	50		Requerido
dep_delegacion	Delegación de la Dependencia	VARCHAR	50		Requerido
dep_colonia	Colonia de la Dependencia	VARCHAR	50		Requerido
dep_calle	Calle de la Dependencia	VARCHAR	50		Requerido
dep_cp	CP de la Dependencia	CHAR	5		Requerido
dep_tel	Teléfono de la Dependencia	VARCHAR	30		
dep_fax	Fax de la Dependencia	VARCHAR	20		
dep_titular	Titular de la Dependencia	VARCHAR	50		
dep_telat	Tel Atención de la Dependencia	VARCHAR	30		
dep_faxat	Fax Atención de la Dependencia	VARCHAR	20		
dep_www	Sitio Web de la Dependencia	VARCHAR	40		
dep_email	email Atención de la Dependencia	VARCHAR	25		



Estructura de la tabla `calle`

```
CREATE TABLE `calle` (  
  `del_id` tinyint(2) NOT NULL default '0',  
  `col_id` tinyint(6) NOT NULL default '0',  
  `calle_id` tinyint(7) NOT NULL auto_increment,  
  `calle_nombre` char(40) NOT NULL default "",  
  PRIMARY KEY (`calle_id`, `col_id`, `del_id`),  
  KEY `calle_nombre` (`calle_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Calles';
```

Estructura de la tabla `colonia`

```
CREATE TABLE `colonia` (  
  `del_id` tinyint(2) NOT NULL default '0',  
  `col_id` tinyint(6) NOT NULL auto_increment,  
  `col_nombre` char(40) NOT NULL default "",  
  `col_cp` char(5) NOT NULL default "",  
  PRIMARY KEY (`col_id`),  
  KEY `col_nombre` (`col_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Colonias';
```

Estructura de la tabla `delegacion`

```
CREATE TABLE `delegacion` (  
  `del_id` tinyint(2) NOT NULL auto_increment,  
  `del_nombre` char(25) NOT NULL default "",  
  PRIMARY KEY (`del_id`),  
  KEY `del_nombre` (`del_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Delegaciones';
```

Estructura de la tabla `edo\_solicitud`

```
CREATE TABLE `edo_solicitud` (  
  `edosol_id` tinyint(2) NOT NULL auto_increment,  
  `edosol_nombre` varchar(15) NOT NULL default "",  
  `edosol_descripcion` varchar(255) default NULL,  
  PRIMARY KEY (`edosol_id`),  
  KEY `edosol_nombre` (`edosol_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Estados';
```

Estructura de la tabla `dependencia`

```
CREATE TABLE `dependencia` (
  `dep_clave` tinyint(1) NOT NULL default '0',
  `dep_nombre` varchar(50) NOT NULL default "",
  `dep_delegacion` varchar(50) NOT NULL default "",
  `dep_colonia` varchar(50) NOT NULL default "",
  `dep_calle` varchar(50) NOT NULL default "",
  `dep_cp` varchar(5) NOT NULL default "",
  `dep_tel` varchar(30) default NULL,
  `dep_fax` varchar(20) default NULL,
  `dep_titular` varchar(50) default NULL,
  `dep_telat` varchar(30) default NULL,
  `dep_faxat` varchar(20) default NULL,
  `dep_www` varchar(40) default NULL,
  `dep_email` varchar(25) default NULL,
  PRIMARY KEY (`dep_clave`)
) TYPE=MyISAM COMMENT='Configuración de la Dependencia';
```

Estructura de la tabla `interesado`

```
CREATE TABLE `interesado` (
  `int_id` tinyint(7) NOT NULL auto_increment,
  `int_nombre` varchar(30) NOT NULL default "",
  `int_paterno` varchar(30) NOT NULL default "",
  `int_materno` varchar(30) NOT NULL default "",
  `int_rfc` varchar(15) default NULL,
  `del_id` tinyint(2) NOT NULL default '0',
  `col_id` tinyint(6) NOT NULL default '0',
  `calle_id` tinyint(7) NOT NULL default '0',
  `int_numint` varchar(10) default NULL,
  `int_numext` varchar(10) default NULL,
  `int_telefono` varchar(15) default NULL,
  `int_fax` varchar(15) default NULL,
  `int_puesto` varchar(30) default NULL,
  `torga_id` tinyint(2) NOT NULL default '0',
  `org_id` tinyint(3) NOT NULL default '0',
  PRIMARY KEY (`int_id`),
  KEY `int_nombre` (`int_nombre`,`int_paterno`,`int_rfc`)
) TYPE=MyISAM COMMENT='Catálogo de Interesados';
```

Estructura de la tabla `usu\_permiso`

```
CREATE TABLE `usu_permiso` (
  `usuper_id` tinyint(4) NOT NULL auto_increment,
  `usu_id` tinyint(3) NOT NULL default '0',
  `perm_id` tinyint(6) NOT NULL default '0',
  PRIMARY KEY (`usuper_id`)
) TYPE=MyISAM COMMENT='Relación de Usuarios y Permisos';
```

Estructura de la tabla `materia`

```
CREATE TABLE `materia` (  
  `mat_id` tinyint(2) NOT NULL auto_increment,  
  `mat_nombre` char(40) NOT NULL default "",  
  `mat_activo` char(1) default NULL,  
  PRIMARY KEY (`mat_id`),  
  KEY `mat_nombre` (`mat_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Materias';
```

Estructura de la tabla `tipo\_organizacion`

```
CREATE TABLE `tipo_organizacion` (  
  `torga_id` tinyint(2) NOT NULL auto_increment,  
  `torga_nombre` char(80) NOT NULL default "",  
  PRIMARY KEY (`torga_id`),  
  KEY `torga_nombre` (`torga_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Tipos de Organizaciones';
```

Estructura de la tabla `operacion`

```
CREATE TABLE `operacion` (  
  `oper_id` tinyint(1) NOT NULL auto_increment,  
  `oper_nombre` varchar(10) NOT NULL default "",  
  PRIMARY KEY (`oper_id`),  
  KEY `oper_nombre` (`oper_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Operaciones en el Sistema';
```

Estructura de la tabla `organizacion`

```
CREATE TABLE `organizacion` (  
  `torga_id` tinyint(2) NOT NULL default '0',  
  `org_id` tinyint(3) NOT NULL auto_increment,  
  `org_nombre` char(80) NOT NULL default "",  
  PRIMARY KEY (`org_id`),  
  KEY `org_nombre` (`org_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Organizaciones';
```

Estructura de la tabla `permiso`

```
CREATE TABLE `permiso` (  
  `perm_id` tinyint(6) NOT NULL default '0',  
  `perm_nombre` varchar(30) NOT NULL default "",  
  PRIMARY KEY (`perm_id`)  
) TYPE=MyISAM COMMENT='Catálogo de Permisos del Sistema';
```

Estructura de la tabla `ser\_area`

```
CREATE TABLE `ser_area` (  
  `serar_id` tinyint(4) NOT NULL auto_increment,  
  `mat_id` tinyint(2) NOT NULL default '0',  
  `ser_id` tinyint(4) NOT NULL default '0',  
  `area_id` tinyint(2) NOT NULL default '0',  
  PRIMARY KEY (`serar_id`)  
) TYPE=MyISAM COMMENT='Relación de Servicios y Áreas';
```

Estructura de la tabla `servicio`

```
CREATE TABLE `servicio` (  
  `mat_id` tinyint(2) NOT NULL default '0',  
  `ser_id` tinyint(4) NOT NULL auto_increment,  
  `ser_nombre` varchar(255) NOT NULL default "",  
  `ser_descripcion` varchar(255) default NULL,  
  `ser_usuarios` varchar(255) default NULL,  
  `ser_trespuesta` varchar(255) default NULL,  
  `ser_requisitos` text,  
  `ser_beneficios` text,  
  `ser_fundamento` text,  
  `ser_observaciones` text,  
  `ser_oficinas` text,  
  `ser_costo` varchar(255) default NULL,  
  `ser_apago` varchar(255) default NULL,  
  `ser_activo` char(1) default NULL,  
  PRIMARY KEY (`ser_id`),  
  KEY `ser_nombre` (`ser_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Servicios';
```

Estructura de la tabla `solicitud`

```
CREATE TABLE `solicitud` (  
  `dep_clave` tinyint(1) NOT NULL default '0',  
  `sol_anio` varchar(4) NOT NULL default "",  
  `sol_folio` varchar(6) NOT NULL default "",  
  `mat_id` tinyint(2) NOT NULL default '0',  
  `ser_id` tinyint(4) NOT NULL default '0',  
  `sol_fecha` date NOT NULL default '0000-00-00',  
  `sol_hora` time NOT NULL default '00:00:00',  
  `sol_folioext` varchar(30) default NULL,  
  `sol_observa` text,  
  `tcap_id` tinyint(2) NOT NULL default '0',  
  `int_id` tinyint(7) NOT NULL default '0',  
  `edosol_id` tinyint(2) NOT NULL default '0',  
  PRIMARY KEY (`dep_clave`,`sol_anio`,`sol_folio`)  
) TYPE=MyISAM COMMENT='Solicitudes de servicios';
```



Estructura de la tabla `tipo\_captacion`

```
CREATE TABLE `tipo_captacion` (  
  `tcap_id` tinyint(2) NOT NULL auto_increment,  
  `tcap_nombre` char(25) NOT NULL default "",  
  PRIMARY KEY (`tcap_id`),  
  KEY `tcap_nombre` (`tcap_nombre`)  
) TYPE=MyISAM COMMENT='Catálogo de Tipos de Captación';
```

Estructura de la tabla `solicitud\_detalle`

```
CREATE TABLE `solicitud_detalle` (  
  `soldet_id` tinyint(6) NOT NULL auto_increment,  
  `dep_clave` tinyint(1) NOT NULL default '0',  
  `sol_anio` varchar(4) NOT NULL default "",  
  `sol_folio` varchar(6) NOT NULL default "",  
  `del_id` tinyint(2) NOT NULL default '0',  
  `col_id` tinyint(6) NOT NULL default '0',  
  `calle_id` tinyint(7) NOT NULL default '0',  
  `soldet_numext` varchar(10) NOT NULL default "",  
  `soldet_numint` varchar(10) NOT NULL default "",  
  `del_id1` tinyint(2) NOT NULL default '0',  
  `col_id1` tinyint(6) NOT NULL default '0',  
  `calle_id1` tinyint(7) NOT NULL default '0',  
  `del_id2` tinyint(2) NOT NULL default '0',  
  `col_id2` tinyint(6) NOT NULL default '0',  
  `calle_id2` tinyint(7) NOT NULL default '0',  
  `soldet_seguimiento` text NOT NULL,  
  PRIMARY KEY (`soldet_id`)  
) TYPE=MyISAM COMMENT='Detalle de las Solicitudes';
```

Estructura de la tabla `usuario`

```
CREATE TABLE `usuario` (  
  `usu_id` tinyint(3) NOT NULL auto_increment,  
  `usu_nombre` varchar(30) NOT NULL default "",  
  `usu_paterno` varchar(30) NOT NULL default "",  
  `usu_materno` varchar(30) default NULL,  
  `usu_usuario` varchar(8) NOT NULL default "",  
  `usu_contrasena` varchar(8) NOT NULL default "",  
  `area_id` tinyint(2) NOT NULL default '0',  
  PRIMARY KEY (`usu_id`),  
  KEY `usu_nombre` (`usu_nombre`,`usu_usuario`)  
) TYPE=MyISAM COMMENT='Usuarios del Sistema';
```

#### 5.4 SCRIPTS DE CONSULTA Y ELIMINACIÓN

Script de Consulta	Finalidad
SELECT area_id "ID", area_nombre "Área" FROM area ORDER BY 2	Mostrar catálogo de áreas
SELECT area_id, area_nombre FROM area WHERE area_id=valor	Mostrar un registro específico de área
SELECT b.bitac_id "ID", u.usu_usuario "Usuario", b.bitac_fech_ent "Fecha Entrada", b.bitac_entrada "Hora Entrada" b.bitac_fech_sal "Fecha Salida", b.bitac_salida "Hora Salida" FROM bitacora_acceso b, usuario u WHERE b.usu_id=u.usu_id	Mostrar la Bitácora de Accesos al sistema
SELECT b.bitop_id "ID", u.usu_usuario "Usuario", o.oper_nombre "Operación",b.bitop_fecha "Fecha", 'b.bitop_hora "Hora", b.bitop_folio "Folio", b.bitop_tabla "Tabla" 'FROM bitacora_operacion b, usuario u, operacion o WHERE b.usu_id=u.usu_id AND b.oper_id=o.oper_id	Mostrar la Bitácora de Operaciones dentro del sistema
SELECT calle_id "ID", calle_nombre "Calle", n.col_nombre "Colonia", d.del_nombre "Delegación" FROM calle c, colonia n, delegacion d WHERE c.col_id=n.col_id AND c.del_id=d.del_id ORDER BY c.calle_nombre	Mostrar catálogo de calles
SELECT calle_id, calle_nombre, n.col_nombre, d.del_nombre FROM calle c, colonia n, delegacion d WHERE c.calle_id= valor AND c.col_id=n.col_id AND c.del_id=d.del_id	Mostrar un registro específico de calle
SELECT tcap_id "ID", tcap_nombre "Tipo de Captación" FROM tipo_captacion ORDER BY 2	Mostrar catálogo tipo de captación
SELECT tcap_id, tcap_nombre FROM tipo_captacion WHERE tcap_id= valor	Mostrar un registro específico de calle
SELECT col_id "ID", col_nombre "Colonia", col_cp "CP", d.del_nombre "Delegación" FROM colonia c, delegacion d WHERE c.del_id=d.del_id ORDER BY c.col_nombre	Mostrar catálogo colonias
SELECT col_id, col_nombre, col_cp, d.del_nombre FROM colonia c, delegacion d WHERE col_id= valor AND c.del_id=d.del_id	Mostrar un registro específico de colonia
SELECT usu_usuario , usu_contrasena FROM usuario WHERE usu_usuario= valor1 AND usu_contrasena= valor2	Buscar un usuario del sistema
SELECT del_id "ID", del_nombre "Delegación" FROM delegacion ORDER BY 2	Mostrar catálogo de delegaciones
SELECT del_id, del_nombre FROM delegacion WHERE del_id= valor	Mostrar un registro de delegación
SELECT int_id "ID", int_paterno "Paterno", int_materno "Materno", int_nombre "Nombre", int_rfc "RFC", int_telefono "Teléfono" FROM interesado ORDER BY 2	Mostrar catálogo de interesados
SELECT mat_id "ID", mat_nombre "Materia" FROM materia ORDER BY 2	Mostrar catálogo de materias
SELECT mat_id, mat_nombre mat_activo FROM materia WHERE mat_id=valor	Mostrar un registro específico de materia
SELECT org_id "ID", org_nombre "Organización", t.torga_nombre "Tipo de Organización" FROM organizacion o, tipo_organizacion t WHERE o.torga_id=t.torga_id ORDER BY o.org_nombre	Mostrar catálogo de organizaciones
SELECT org_id, org_nombre, t.torga_nombre FROM organizacion o, tipo_organizacion t WHERE org_id= valor AND o.torga_id=t.torga_id ORDER BY o.org_nombre	Mostrar un registro específico de organizacion

Script de Consulta	Finalidad
SELECT sol_anio "Año", sol_folio "Folio", t.ser_nombre "Solicitud", i.int_nombre "Interesado", sol_fecha "Fecha" FROM servicio t, solicitud s, interesado i WHERE s.ser_id=t.ser_id AND s.int_id=i.int_id ORDER BY s.sol_folio AND s.sol_anio	Mostrar solicitudes para recepción
SELECT sol_anio "Año", sol_folio "Folio", t.ser_nombre "Solicitud", i.int_nombre "Interesado", sol_fecha "Fecha" FROM servicio t, solicitud s, interesado i WHERE s.ser_id=t.ser_id AND s.int_id=i.int_id ORDER BY s.sol_folio AND s.sol_anio	Mostrar solicitudes para seguimiento
SELECT ser_id "ID", ser_nombre "Servicio", m.mat_nombre "Materia" FROM servicio s, materia m WHERE s.mat_id=m.mat_id ORDER BY s.ser_nombre	Mostrar catálogo de servicios
SELECT ser_id, ser_nombre, ser_descripcion, ser_usuarios, ser_trespuesta, ser_requisitos, ser_beneficios, ser_fundamento, ser_observaciones, ser_oficinas, ser_costo, ser_apago FROM servicio WHERE ser_id= valor	Mostrar un registro específico de servicio
SELECT sol_anio "Año", sol_folio "Folio", t.ser_nombre "Solicitud", i.int_nombre "Interesado", sol_fecha "Fecha", e.edosol_nombre "Estado" FROM servicio t, solicitud s, interesado i, edo_solicitud e WHERE s.ser_id=t.ser_id AND s.int_id=i.int_id AND s.edosol_id=e.edosol_id ORDER BY s.sol_folio AND s.sol_anio	Mostrar solicitudes ingresadas
SELECT torga_id "ID", torga_nombre "Tipo de Organización" FROM tipo_organizacion ORDER BY 2	Mostrar catálogo de Tipos de organizaciones
SELECT torga_id, torga_nombre FROM tipo_organizacion WHERE torga_id=valor	Mostrar un Tipos de organización específico
SELECT usu_id "ID", usu_usuario "Usuario", usu_nombre "Nombre", usu_paterno "Paterno", usu_materno "Materno", a.area_nombre "Área" FROM usuario u, area a WHERE u.area_id=a.area_id ORDER BY u.usu_nombre	Mostrar catálogo de Usuarios
SELECT usu_id, usu_nombre, usu_paterno, usu_materno, usu_usuario, usu_contrasena, a.area_nombre FROM usuario u, area a WHERE usu_id= valor AND u.area_id=a.area_id	Mostrar un usuario específico

Script de Eliminación	Finalidad
DELETE FROM area WHERE area_id= valor	Eliminar un registro de área
DELETE FROM tipo_captacion WHERE tcap_id= valor	Eliminar un registro de tipo de captación
DELETE FROM colonia WHERE col_id= valor	Eliminar un registro de colonia
DELETE FROM delegacion WHERE del_id= valor	Eliminar un registro de delegacion
DELETE FROM materia WHERE mat_id= valor	Eliminar un registro de materia
DELETE FROM organizacion WHERE org_id= valor	Eliminar un registro de organizacion
DELETE FROM tipo_organizacion WHERE torga_id= valor	Eliminar un registro de tipo de organizacion

## TEMA 6 DISEÑO DE LAS REGLAS DE NEGOCIO

Las reglas de negocio son todos los procesos, algoritmos, procedimientos, reglas, lógica, validaciones, cálculos y restricciones aplicables que requiere el sistema de información para tener una funcionalidad específica.

### 6.1 REGLAS DEL MODELO DE DATOS

Reglas en las tablas dentro de la base de datos.

Tabla	Reglas
<b>delegacion</b>	del_id = autonumérico y <> Nulo del_nombre = char 25 <> Nulo
<b>colonia</b>	col_id = autonumérico y <> Nulo col_nombre = char 40 <> Nulo col_cp = char 5 <> Nulo del_id = int <> Nulo
<b>calle</b>	calle_id = autonumérico y <> Nulo calle_nombre = char 40 y <> Nulo del_id = int <> Nulo col_id = int <> Nulo
<b>tipo_organizacion</b>	torga_id = autonumérico y <> Nulo torga_nombre = char 80
<b>organización</b>	org_id = autonumérico y <> Nulo org_nombre = char 80 y <> Nulo torga_id = int <> Nulo
<b>tipo_captacion</b>	tcap_id = autonumérico y <> Nulo tcap_nombre = char 25 <> Nulo
<b>edo_solicitud</b>	edosol_id = autonumérico y <> Nulo edosol_nombre = varchar 25 <> Nulo edosol_descripcion = varchar 255
<b>area</b>	area_id = autonumérico y <> Nulo area_nombre = varchar 40
<b>materia</b>	mat_id = autonumérico y <> Nulo mat_nombre = char 40 <> Nulo mat_activo = booleano
<b>ser_area</b>	serar_id = autonumérico y <> Nulo mat_id = int <> Nulo ser_id = int <> Nulo area_id = int <> Nulo
<b>usuario</b>	usu_id = autonumérico y <> Nulo usu_nombre = varchar 30 y <> Nulo usu_paterno = varchar 30 y <> Nulo usu_materno = varchar 30 y <> Nulo usu_usuario = char 8 y <> Nulo usu_contrasena = char 8 y <> Nulo area_id = int

Tabla	Reglas
<b>servicio</b>	ser_id = autonumérico y <> Nulo ser_nombre = char 50 y <> Nulo ser_descripcion = char 80   ser_usuarios = char 255 ser_trespuesta= char 18   ser_requisitos = text   ser_beneficios = text ser_fundamento = text   ser_observaciones= text ser_oficinas= text ser_costo= char 255   ser_apago = char 255   ser_activo = booleano mat_id = int <> Nulo
<b>operacion</b>	oper_id = autonumérico y <> Nulo oper_nombre = varchar 10 requerido
<b>bitacora_acceso</b>	bitac_id = autonumérico y <> Nulo usu_id = int y <> Nulo bitac_fech_ent = fecha y <> Nulo bitac_fech_sal = fecha y <> Nulo bitac_entrada = hora y <> Nulo bitac_salida = hora y <> Nulo
<b>bitacora_operacion</b>	bitop_id = autonumérico y <> Nulo usu_id = int y <> Nulo bitopc_fecha = fecha y <> Nulo bitop_hora = hora y <> Nulo bitop_folio = char 12 y <> Nulo bitop_tabla = char 30 y <> Nulo
<b>permiso</b>	perm_id = int y <> Nulo perm_nombre = varchar 80 y <> Nulo
<b>usu_permiso</b>	usuper_id = autonumérico y <> Nulo usu_id = int y <> Nulo perm_id = int y <> Nulo
<b>solicitud</b>	dep_clave = int y <> Nulo sol_anio = char 4 y <> Nulo sol_folio = char 6 y <> Nulo mat_id = int y <> Nulo ser_id = int y <> Nulo sol_fecha = fecha y <> Nulo   sol_hora = hora y <> Nulo sol_folioext = char 18   sol_observa = text tcap_id = int y <> Nulo   int_id = int y <> Nulo edosol_id = int y <> Nulo
<b>interesado</b>	int_id = autonumérico y <> Nulo int_nombre = varchar 30 y <> Nulo int_paterno = varchar 30 y <> Nulo int_materno = varchar 30 y <> Nulo int_rfc = varchar 15   del_id = int y <> Nulo col_id = int y <> Nulo   calle_id = int y <> Nulo int_numint = varchar 10   int_numext = varchar 10 int_telefono = varchar 15   int_fax = varchar 15 int_puesto = varchar 30   torga_id = int y <> Nulo org_id = int y <> Nulo
<b>dependencia</b>	dep_clave = autonumérico y <> Nulo dep_nombre = varchar 50 y <> Nulo dep_delegacion = varchar 50 y <> Nulo dep_colonia = varchar 50 y <> Nulo   dep_calle = varchar 50 y <> Nulo dep_cp = char 5 y <> Nulo   dep_tel = varchar 30 dep_fax = varchar 20   dep_titular = varchar 50 dep_telat = varchar 30   dep_faxat = varchar 20 dep_www = varchar 40   dep_email = varchar 25

## 6.2 REGLAS DE RELACIÓN

Reglas entre las tablas de la base de datos

Reglas	Tablas
Toda colonia debe pertenecer a una delegación	colonia delegacion
Toda calle debe pertenecer a una colonia	colonia calle
Toda organización debe pertenecer a un tipo de organización	tipo_organizacion organizacion
Todo servicio debe pertenecer a una materia	materia servicio
Todo servicio debe ser atendido por una área	area servicio ser_area
Todo usuario debe pertenecer a una área	area servicio
Todo usuario debe tener permisos	permiso usuario usu_permiso
Toda solicitud debe realizarse de un servicio del catálogo	solicitud servicio
Toda solicitud debe tener el dato de la Dependencia para la cual fue configurado el sistema	solicitud dependencia
Toda solicitud debe ser recibida por un tipo de captación	solicitud tipo_captación
Toda solicitud de servicio debe ser hecha por un interesado	solicitud interesado
Toda solicitud de servicio debe tener un estado de la situación que guarda	edo_solicitud solicitud
Toda solicitud debe tener un detalle	solicitud solicitud_detalle
Todo detalle de solicitud debe tener como referencia de ubicación una delegación, colonia y calle	solicitud_detalle delegacion colonia calle
Toda operación registrada en la Bitácora de operaciones debe indicar el tipo de operación realizada	bitacora_operacion operacion
Toda operación registrada en la Bitácora de operaciones debe indicar el usuario que realizó dicha operación	bitacora_operacion usuario
Toda acceso al sistema registrado en la Bitácora de acceso debe indicar el usuario que realizó dicho acceso	bitacora_acceso usuario

### 6.3 REGLAS DE RESTRICCIÓN

Información que puede tener el sistema

Reglas	Tablas
Las fechas que se manejen en el sistema deben tener el siguiente formato: dd/mm/aaaa	bitacora_acceso bitacora_operaciones solicitud
Las horas que se manejen en el sistema deben tener el siguiente formato: hh:mm	bitacora_acceso bitacora_operaciones solicitud

### 6.4 REGLAS DE FLUJO

Determinar y limitar como fluye la información a través del sistema

Reglas
Sólo pueden acceder al sistema los usuarios con cuenta en el sistema, clave de acceso, que se encuentren como activos y tengan permisos para cada módulo del sistema.
Los usuarios al ingresar al sistema sólo podrán ver y acceder a los módulos a los cuales tienen permiso.
Los usuarios sólo podrán ejecutar las operaciones a las que tienen permisos dentro de cada módulo del sistema.
Cuando un usuario ingrese al sistema deberá registrarse dicho acceso en la bitácora de accesos al sistema.
Para registrar un interesado se deberá capturar como mínimo la siguiente información: nombre, apellido paterno, domicilio (delegación, colonia, calle, num. ext, organización)
Para poder modificar los datos de un interesado deberá contener como mínimo la siguiente información: nombre, apellido paterno, domicilio (delegación, colonia, calle, num. ext, ) y organización
Para poder eliminar un interesado éste no deberá tener solicitudes relacionadas.
Las operaciones de registro, modificación, eliminación y búsqueda se registrarán en la bitácora de operaciones en el sistema.
Para registrar una solicitud se deberá capturar como mínimo la siguiente información: interesado, materia, servicio, tipo de captación, observaciones, domicilio donde se prestará el servicio (delegación, colonia, calle).
Para modificar los datos de una solicitud se deberá capturar como mínimo la siguiente información: interesado, materia, servicio, tipo de captación, observaciones, domicilio donde se prestará el servicio (delegación, colonia, calle).
Para poder eliminar una solicitud esta deberá estar en el estado ingresado y no estar en seguimiento en las áreas operativas.

<b>Reglas</b>
Para poder imprimir acuses de recibo se deberá contar la información mínima del interesado y de la solicitud (mismas que se menciona en reglas antes descritas).
El usuario sólo podrá recibir mediante el sistema aquellas solicitudes de servicios que sean de la competencia de su área.
Al registrar una solicitud ésta debe pasar al estado "ingresada"
Al recibir una solicitud ésta debe pasar al estado "en proceso"
Al atender una solicitud ésta debe pasar al estado "atendida"
No se podrá pasar del estado "ingresada" al estado "atendida" sin antes pasar por el estado "en proceso".
El usuario sólo podrá dar seguimiento a aquellas solicitudes de servicios que sean de la competencia de su área.
Para generar la orden de trabajo de una solicitud ésta deberá encontrarse en el estado: "en proceso" y el campo "seguimiento" deberá contener información (que serán las instrucciones que aparecerán en la orden en mención.
Para dar de alta registros de cada uno de los catálogos que contiene el sistema se deberá ingresar toda la información solicitada en cada uno de los catálogos.
Para dar modificar los registros de cada uno de los catálogos que contiene el sistema se deberá ingresar toda la información solicitada en cada uno de los catálogos y se deberá informar al usuario de las implicaciones de la acción.
Para eliminar registros de cada uno de los catálogos que contiene el sistema se deberá validar que estos registros no se encuentren relacionados con otros registros, en caso de ser así la operación será rechazada.
Para dar de alta usuarios del sistema éstos deberán contener como mínimo la siguiente información: nombre, apellido paterno, usuario, contraseña, área a la que pertenece.
Para modificar la información de los usuarios del sistema éstos deberán contener como mínimo la siguiente información: nombre, apellido paterno, usuario, contraseña, área a la que pertenece.
Para eliminar a un usuario del sistema se deberá validar que estos usuarios no contengan registros relacionados (bitácora de accesos).
Para la generación de reportes y la configuración del sistema, se tomarán los datos de la información guardada en la entidad destinada para ello (tabla de la base de datos).
Validar que la captura de información donde se solicite datos con fechas, éstos lleven el siguiente formato: dd/mm/aaaa.
Validar que la captura de información donde se solicite datos con la hora, éstos lleven el siguiente formato: hh/mm.
En cada uno de los programas que conforman la aplicación se validará que el usuario tenga una sesión válida en el sistema



## 6.5 VALIDACIONES

A continuación se muestra el código de la aplicación y la forma en que se deberá validar la información, sólo para los casos generales de inserciones, modificaciones y eliminaciones en cada una de las tablas.

Programa / Código	Función / Tabla
<pre> <b>areaNr.php</b> # Conexión a la Base de Datos .... \$db = NewADOConnection('mysql'); \$db-&gt;Connect("localhost","tessol5","tessol50","tessol5"); \$sql = "SELECT * FROM area WHERE area_id=-1"; \$rs = \$db-&gt;Execute(\$sql); # Arreglo con dato a insertar \$UnaArea = array(); # Construir la sentencia de inserción \$sql_Inserta = \$db-&gt;GetInsertSQL(\$rs,\$UnaArea); <b># Validar la inserción</b> <b>validaInsercion ((\$rs,\$UnaArea)</b> #Insertar el registro \$db-&gt;Execute(\$sql_Inserta); </pre>	<p>Insertar una área en la tabla <b>area</b></p>
<pre> <b>areaMr.php</b> # Conexión a la Base de Datos \$db = NewADOConnection('mysql'); \$db-&gt;Connect("localhost","tessol5","tessol50","tessol5"); \$sql = "SELECT * FROM area WHERE area_id=".\$HTTP_POST_VARS['area_id']; \$rs = \$db-&gt;Execute(\$sql); # Arreglo con dato a modificar \$UnaArea = array(); \$UnaArea["area_nombre"]=\$HTTP_POST_VARS['area_nombre']; # Construir la sentencia de actualización \$sql_Actualiza = \$db-&gt;GetUpdateSQL(\$rs,\$UnaArea); <b># Validar la modificación</b> <b>validaModificacion (\$rs,\$UnaArea)</b> #Actualizar el registro \$db-&gt;Execute(\$sql_Actualiza); </pre>	<p>Modificar una área en la tabla <b>area</b></p>
<pre> <b>areaEr.php</b> # Conexión a la Base de Datos \$db = NewADOConnection('mysql'); \$db-&gt;Connect("localhost","tessol5","tessol50","tessol5"); # Construir la sentencia de eliminación \$sql = "DELETE FROM area WHERE area_id=".\$HTTP_POST_VARS['area_id']; <b># Validar la eliminación</b> <b>validaEliminacion (\$sql)</b> #Eliminar el registro \$rs = \$db-&gt;Execute(\$sql); </pre>	<p>Eliminar una área en la tabla <b>area</b></p>

## TEMA 7 IMPLEMENTACIÓN

### 7.1 PLAN DE IMPLEMENTACIÓN

El objetivo principal de la fase de implementación es la entrega y aprobación del sistema en su totalidad, por lo que es necesario llevar a cabo una serie de actividades encaminadas para poner en producción al sistema, en la siguiente tabla enlistamos las actividades necesarias para poder implementar el sistema.

No	Actividades	Productos
<b>1</b>	<b>Establecimiento del Plan de Implementación</b>	
1.1	Definición del Plan de Implementación	Plan de implementación
1.2	Especificaciones del equipo de implementación	Equipo de implementación
<b>2</b>	<b>Capacitación Necesaria para la Implementación</b>	
2.1	Preparación de la capacitación del equipo de implementación	Programa de Capacitación ( Forma, materiales y calendario de la Capacitación)
2.2	Capacitación del equipo de implementación	Programa de Capacitación - Registro de Asistencia
2.3	Preparación de la capacitación de los usuarios finales	Programa de capacitación usuarios ( Forma, materiales y calendario de la Capacitación)
2.4	Seguimiento de la capacitación de los usuarios finales	Programa de capacitación usuarios - Registro de Asistencia
<b>3</b>	<b>Incorporación del Sistema al Entorno de Operación</b>	
3.1	Preparación de la Instalación	Incidentes de la preparación
3.2	Realización de la Instalación	- Producto Software (instalado) - Código de los componentes de migración y carga de datos - Migración y carga de datos
<b>4</b>	<b>Carga de Datos al Entorno de Operación</b>	
4.1	Migración y carga de datos inicial	- Base de datos /archivos de carga
<b>5</b>	<b>Pruebas de Implementación</b>	
5.1	Preparación de las Pruebas de Implementación	Plan de Pruebas
5.2	Realización de las Pruebas de Implementación	Resultado de las Pruebas de Implementación
5.3	Evaluación del resultado de las Pruebas de Implementación	Evaluación del resultado de las Pruebas de Implementación
<b>6</b>	<b>Pruebas de Aceptación del Sistema</b>	
6.1	Preparación de las Pruebas de Aceptación	Plan de pruebas
6.2	Realización de las Pruebas de Aceptación	Resultado de las Pruebas de Aceptación
6.3	Evaluación del resultado de las Pruebas de Aceptación	Evaluación del Resultado de las Pruebas de Aceptación
<b>7</b>	<b>Preparación del Mantenimiento del Sistema</b>	
7.1	Establecimiento de infraestructura para el mantenimiento	Plan de mantenimiento
7.2	Formalización del plan de mantenimiento	Plan de mantenimiento
<b>8</b>	<b>Presentación y Aprobación del Sistema</b>	
8.1	Convocatoria para la presentación del sistema	Plan de presentación del sistema
8.2	Aprobación del sistema	Aprobación del sistema
<b>9</b>	<b>Instalación en Producción</b>	
9.1	Preparación del entorno de producción	Incidentes de la instalación
9.2	Activación del Sistema en Producción	Sistema en producción

Tabla 4

A continuación se muestra el Plan de Implementación:

Id	Nombre de tarea	Duración	Comienzo	Fin
1	<b>Implementación del Programa de Gestión de Servicios</b>	37 días	lun 02/08/04	mar 21/09/04
2	Definición de Equipo de Implementación	1 día	lun 02/08/04	lun 02/08/04
3	Elaboración de Programa de Capacitación del Equipo de Implementación	1 día	mar 03/08/04	mar 03/08/04
4	Capacitación del Equipo de Implementación	3 días	mié 04/08/04	vie 06/08/04
5	Elaboración de Programa de Capacitación de Usuarios Finales	1 día	lun 09/08/04	lun 09/08/04
6	Capacitación de Usuarios Finales	5 días	mar 10/08/04	lun 16/08/04
7	Preparación de la Instalación	1 día	jue 19/08/04	jue 19/08/04
8	Instalación del Sistema	2 días	vie 20/08/04	lun 23/08/04
9	Migración y Carga Inicial de Datos	2 días	mar 24/08/04	mié 25/08/04
10	Preparación de la Pruebas de Implementación	1 día	jue 26/08/04	jue 26/08/04
11	Realización de las Pruebas de Implementación	5 días	vie 27/08/04	jue 02/09/04
12	Evaluación del resultado de las Pruebas de Implementación	1 día	vie 03/09/04	vie 03/09/04
13	Preparación de la Pruebas de Aceptación	1 día	lun 06/09/04	lun 06/09/04
14	Realización de las Pruebas de Aceptación	3 días	mar 07/09/04	jue 09/09/04
15	Evaluación del resultado de las Pruebas de Aceptación	1 día	vie 10/09/04	vie 10/09/04
16	Establecimiento de infraestructura para el mantenimiento	1 día	lun 13/09/04	lun 13/09/04
17	Formalización del plan de mantenimiento	1 día	mar 14/09/04	mar 14/09/04
18	Convocatoria para la presentación del sistema	1 día	mié 15/09/04	mié 15/09/04
19	Aprobación del sistema	1 día	jue 16/09/04	jue 16/09/04
20	Preparación del entorno de producción	2 días	vie 17/09/04	lun 20/09/04
21	Activación del Sistema en Producción	1 día	mar 21/09/04	mar 21/09/04

Tabla 5

Equipo de Implementación:

Id	Nombre
1	Jefe de Proyecto
2	Responsable de Implementación
3	Responsable de Sistemas
4	Directores de los Usuarios

Tabla 6

<b>Programa de Capacitación del Equipo de Implementación</b>	
	<b>Curso para el Responsable de Sistemas</b>
	<b>Objetivo:</b> Mostrar al Responsable de Sistemas el uso de la tecnología base necesaria para el funcionamiento del nuevo Sistema.
	<b>Materiales:</b> Área con 2 computadoras y Manual de Administración del Sistema
	<b>Temario:</b> Instalación, configuración y administración de Servidores - Servidor Web Apache - Servidor de Base de Datos-MySQL. Instalación y configuración de PHP y módulo para Apache

Tabla 7

<b>Programa de Capacitación a Usuario Finales</b>	
---	--

	<b>Curso para Operadores de Atención Ciudadana</b>
	<b>Objetivo:</b> Mostrar a los operadores que atienden a los ciudadanos la forma de utilizar el Sistema de tal forma que sean capaces de ingresar solicitudes.
	<b>Materiales:</b> Sala de Cómputo con 5 computadoras, un proyector y Manual de Usuario del Sistema (Operador)
	<b>Temario:</b> Introducción a Internet/Intranet Manejo de navegador Web Generalidades del Programa de Gestión de Servicios Módulo de Registro Solicitantes: Registro, modificación, eliminación y búsqueda Solicitudes: Registro, modificación, eliminación y búsqueda Módulo de Reportes

	<b>Curso para Áreas Operativas</b>
	<b>Objetivo:</b> Mostrar a los usuarios de las áreas ejecutoras que atienden las solicitudes de servicios la forma de darle seguimiento a las mismas mediante el nuevo Sistema.
	<b>Materiales:</b> Sala de Cómputo con 5 computadoras, un proyector y Manual de Usuario del Sistema (Ejecutor).
	<b>Temario:</b> Introducción a Internet/Intranet Manejo de navegador Web Generalidades del Programa de Gestión de Servicios Módulo de Seguimiento Recepción: Consulta y recepción. Seguimiento: Seguimiento, búsqueda y ordenes de trabajo. Módulo de Reportes

Tabla 8



**Reporte de Incidentes de la Preparación de la Instalación:**

Descripción del Incidente	Tipo Hardware o Software			
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S
		H		S

Tabla 11

**PLAN DE PRUEBAS**

No	Prueba	Descripción
1	Pruebas funcionales	Verificar que se realizan correctamente todas las funciones.
2	Pruebas de comunicaciones	Comprobar que las interfaces entre los componentes del sistema funcionan bien.
3	Pruebas de rendimiento	Determinar que los tiempos de respuesta estén dentro de las especificaciones establecidas.
4	Pruebas de volúmen	Examinar el comportamiento del sistema con grandes volúmenes de datos.
5	Pruebas de sobrecarga	Probar el funcionamiento del sistema someténdolo a cargas masivas de trabajo.
6	Pruebas de disponibilidad de datos	Probar que el sistema pueda recuperar ante fallos del equipo equipo y del software.
7	Pruebas de facilidad de uso	Revisar que el sistema se adapta a las necesidades de los usuarios finales.
8	Pruebas de operación	Comprobar que funcionan los procedimientos de operación, arranque y reinicio del sistema.
9	Pruebas de entorno	Verificar la correcta interacción del sistemas con otros sistemas.
10	Pruebas de seguridad	Probar que los mecanismos de control de acceso al sistema funcionan adecuadamente.

Tabla 12

A continuación se muestra el Plan de Pruebas

<b>Id</b>	<b>Nombre de tarea</b>	<b>Duración</b>	<b>Comienzo</b>	<b>Fin</b>
1	<b>Plan de Pruebas</b>	5 días	27/08/04	02/09/04
2	Pruebas funcionales	1 día	27/08/04	27/08/04
3	Pruebas de comunicaciones	1 día	27/08/04	27/08/04
4	Pruebas de rendimiento	1 día	30/08/04	30/08/04
5	Pruebas de volúmen	1 día	30/08/04	30/08/04
6	Pruebas de sobrecarga	1 día	30/08/04	30/08/04
7	Pruebas de disponibilidad de datos	1 día	31/08/04	31/08/04
8	Pruebas de facilidad de uso	5 días	27/08/04	02/09/04
9	Pruebas de operación	3 días	27/08/04	31/08/04
10	Pruebas de entorno	1 día	01/09/04	01/09/04
11	Pruebas de seguridad	1 día	02/09/04	02/09/04

## 7.2 ESTÁNDARES DE PROGRAMACIÓN

Nombre del Programa	Función	Programas Requeridos	Tablas
area.php	Mostrar los datos del catálogo de áreas	seguridad.php adodb.inc.php adodb-ABC.inc.php	areas
areaD.php	Mostrar los detalles de una área	seguridad.php adodb.inc.php	areas
areaE.php	Mostrar los datos de una área antes de eliminarla	seguridad.php adodb.inc.php	areas
areaEr.php	Eliminar una área	seguridad.php adodb.inc.php areaE.php	areas
areaM.php	Mostrar los datos de una área para modificarla	seguridad.php adodb.inc.php	areas
areaMr.php	Modificar los datos de una área	seguridad.php adodb.inc.php areaM.php	areas
areaN.php	Capturar los datos de una área	seguridad.php	
areaNr.php	Guardar los datos de un área nueva	seguridad.php adodb.inc.php areaN.php	areas
bitacoraA.php	Mostrar la Bitácora de Accesos al Sistema	seguridad.php adodb.inc.php adodb-ABC.inc.php	bitacora_acceso usuario
bitacoraO.php	Mostrar la Bitácora de Operaciones en el Sistema	seguridad.php adodb.inc.php adodb-ABC.inc.php	bitacora_acceso usuario operacion
calle.php	Mostrar los datos del catálogo de calles	seguridad.php adodb.inc.php adodb-ABC.inc.php	calle colonia delegacion
calleD.php	Mostrar los detalles de una calle	seguridad.php adodb.inc.php	calle colonia delegacion
calleE.php	Mostrar los datos de una calle antes de eliminarla	seguridad.php adodb.inc.php	calle colonia delegacion
calleM.php	Mostrar los datos de una calle para modificarla	seguridad.php adodb.inc.php	calle colonia delegacion
calleN.php	Capturar los datos de una calle	seguridad.php	delegacion colonia



Nombre del Programa	Función	Programas Requeridos	Tablas
captacion.php	Mostrar los datos del catálogo de captaciones	seguridad.php adodb.inc.php adodb-ABC.inc.php	captacion
captacionD.php	Mostrar los detalles de una captación	seguridad.php adodb.inc.php	captación
captacionE.php	Mostrar datos de captación antes de eliminarla	seguridad.php adodb.inc.php	captación
captacionEr.php	Eliminar una captación	seguridad.php adodb.inc.php	captación
captacionM.php	Mostrar los datos de una captación para modificarla	seguridad.php adodb.inc.php	captación
captacionMr.php	Modificar los datos de una captación	seguridad.php adodb.inc.php	captación
captacionN.php	Capturar los datos de una captación	seguridad.php	
captacionNr.php	Guardar los datos de una captación	seguridad.php adodb.inc.php	captación
colonia.php	Mostrar los datos del catálogo de colonias	seguridad.php adodb.inc.php adodb-ABC.inc.php	colonia delegacion
coloniaD.php	Mostrar los detalles de una colonia	seguridad.php adodb.inc.php	colonia delegacion
coloniaE.php	Mostrar datos de la colonia antes de eliminarla	seguridad.php adodb.inc.php	colonia delegacion
coloniaEr.php	Eliminar una colonia	seguridad.php adodb.inc.php	colonia
coloniaM.php	Mostrar los datos de una colonia para modificarla	seguridad.php adodb.inc.php	colonia delegacion
coloniaMr.php	Modificar los datos de una colonia	seguridad.php adodb.inc.php	colonia
coloniaN.php	Capturar los datos de una colonia	seguridad.php	delegacion
coloniaNr.php	Guardar los datos de una colonia	seguridad.php adodb.inc.php	colonia
config.php	Mostrar los datos de la configuración del sistema	seguridad.php	dependencia
configM.php	Capturar los datos de la configuración del sistema	seguridad.php	dependencia
index.php	Capturar datos de acceso del usuario al sistema	control.php	
control.php	Validar acceso del usuario al sistema creando sesión	adodb.inc.php index.php	usuario
principal.php	Muestra opciones del sistema	seguridad.php	
encabezado.php	Muestra encabezado	seguridad.php	
contenido.php	Despliega el resultado de las operaciones del sistema	seguridad.php	
salir.php	Elimina sesión del usuario		
seguridad.php	Valida sesión del usuario	control.php	

Nombre del Programa	Función	Programas Requeridos	Tablas
delegacion.php	Mostrar los datos del catálogo de delegaciones	seguridad.php adodb.inc.php adodb-ABC.inc.php	delegacion
delegacionD.php	Mostrar los detalles de una delegación	seguridad.php adodb.inc.php	delegacion
delegacionE.php	Mostrar datos de la delegación a eliminar	seguridad.php adodb.inc.php	delegacion
delegacionEr.php	Eliminar una delegación	seguridad.php adodb.inc.php	delegacion
delegacionM.php	Mostrar los datos de una delegación para modificarla	seguridad.php adodb.inc.php	delegacion
delegacionMr.php	Modificar los datos de una delegación	seguridad.php adodb.inc.php	delegacion
delegacionN.php	Capturar los datos de una delegación	seguridad.php	
delegacionNr.php	Guardar los datos de una delegación	seguridad.php adodb.inc.php	delegacion
interesado.php	Mostrar los datos del catálogo de interesados	seguridad.php adodb.inc.php adodb-ABC.inc.php	interesado
interesadoD.php	Mostrar los detalles de un interesado	seguridad.php adodb.inc.php	interesado
interesadoE.php	Mostrar los datos de un interesado a eliminar	seguridad.php adodb.inc.php	interesado
interesadoM.php	Mostrar los datos de un interesado para modificar	seguridad.php adodb.inc.php	interesado
interesadoN.php	Capturar los datos de un interesado	seguridad.php	
materia.php	Mostrar los datos del catálogo de materias	seguridad.php adodb.inc.php adodb-ABC.inc.php	materia
materiaD.php	Mostrar los detalles de una materia	seguridad.php adodb.inc.php	materia
materiaE.php	Mostrar datos de la materia a eliminar	seguridad.php adodb.inc.php	materia
materiaEr.php	Eliminar una materia	seguridad.php adodb.inc.php	materia
materiaM.php	Mostrar los datos de una materia para modificarla	seguridad.php adodb.inc.php	materia
materiaMr.php	Modificar los datos de una materia	seguridad.php adodb.inc.php	materia
materiaN.php	Capturar los datos de una materia	seguridad.php	
materiaNr.php	Guardar los datos de una materia	seguridad.php adodb.inc.php	materia

Nombre del Programa	Función	Programas Requeridos	Tablas
organizacion.php	Mostrar los datos del catálogo de organizaciones	seguridad.php adodb.inc.php adodb-ABC.inc.php	tipo_organizacion organizacion
organizacionD.php	Mostrar los detalles de una organización	seguridad.php adodb.inc.php	tipo_organizacion organizacion
organizacionE.php	Mostrar datos de la organización a eliminar	seguridad.php adodb.inc.php	tipo_organizacion organizacion
organizacionEr.php	Eliminar una organización	seguridad.php adodb.inc.php	organizacion
organizacionM.php	Mostrar los datos de una organización para modificarla	seguridad.php adodb.inc.php	tipo_organizacion organizacion
organizacionMr.php	Modificar los datos de una organización	seguridad.php adodb.inc.php	organizacion
organizacionN.php	Capturar los datos de una organización	seguridad.php	tipo_organizacion
organizacionNr.php	Guardar los datos de una organización	seguridad.php adodb.inc.php	organizacion
orienta.php	Mostrar formulario de selección de servicio	seguridad.php adodb.inc.php	materia servicio
repcion.php	Mostrar los datos de las solicitudes turnadas a cada una de las áreas	seguridad.php adodb.inc.php adodb-ABC.inc.php	solicitud interesado
repcionB.php	Mostrar formulario de búsqueda de solicitudes por recibir	seguridad.php adodb.inc.php	solicitud interesado materia servicio tipo_captacion delegacion colonia calle
repcionM.php	Mostrar los datos de una solicitud para recibirla	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion delegacion colonia calle
reportes.php	Mostrar formulario de selección de las opciones para generar el reporte	seguridad.php adodb.inc.php	edo_solicitud tipo_captacion tipo_organizacion organización delegacion colonia calle

Nombre del Programa	Función	Programas Requeridos	Tablas
seguim.php	Mostrar los datos de las solicitudes turnadas y recibidas en las áreas	seguridad.php adodb.inc.php adodb-ABC.inc.php	solicitud interesado
seguimB.php	Mostrar formulario de búsqueda de solicitudes recibidas	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion edo_solicitud delegacion colonia calle
seguimM.php	Mostrar los datos de una solicitud para darle seguimiento	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion edo_solicitud delegacion colonia calle
usuario.php	Mostrar los datos de los usuarios del sistema	seguridad.php adodb.inc.php adodb-ABC.inc.php	usuario area
usuarioD.php	Mostrar los detalles de un usuario	seguridad.php adodb.inc.php	usuario area
usuarioE.php	Mostrar los datos de un usuario a eliminar	seguridad.php adodb.inc.php	usuario area
usuarioM.php	Mostrar los datos de un usuario para modificar	seguridad.php adodb.inc.php	usuario area
usuarioN.php	Capturar los datos de un usuario	seguridad.php	area
usuarioNr.php	Guardar los datos de un usuario	seguridad.php adodb.inc.php	usuario

Nombre del Programa	Función	Programas Requeridos	Tablas
solicitud.php	Mostrar los datos de las solicitudes ingresadas	seguridad.php adodb.inc.php adodb-ABC.inc.php	solicitud interesado edo_solicitud
solicitudD.php	Mostrar los detalles de una solicitud	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion edo_solicitud delegacion colonia calle
solicitudE.php	Mostrar los datos de una solicitud a eliminar	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion edo_solicitud delegacion colonia calle
solicitudM.php	Mostrar los datos de una solicitud para modificar	seguridad.php adodb.inc.php	solicitud solicitud_detalle interesado materia servicio tipo_captacion edo_solicitud delegacion colonia calle
solicitudN.php	Capturar los datos de una solicitud servicio	seguridad.php	interesado materia servicio tipo_captacion delegacion colonia calle

Nombre del Programa	Función	Programas Requeridos	Tablas
servicio.php	Mostrar los datos del catálogo de servicios	seguridad.php adodb.inc.php adodb-ABC.inc.php	servicio materia
servicioD.php	Mostrar los detalles de un servicio	seguridad.php adodb.inc.php	servicio materia
servicioE.php	Mostrar los datos de un servicio a eliminar	seguridad.php adodb.inc.php	servicio materia
servicioM.php	Mostrar los datos de un servicio para modificar	seguridad.php adodb.inc.php	servicio materia
servicioN.php	Capturar los datos de un servicio	seguridad.php	materia
servicioNr.php	Guardar los datos de un servicio	seguridad.php adodb.inc.php	servicio
torganizacion.php	Mostrar los datos del catálogo de tipos de organizaciones	seguridad.php adodb.inc.php adodb-ABC.inc.php	tipo_organizacion
torganizacionD.php	Mostrar los detalles de un tipo de organización	seguridad.php adodb.inc.php	tipo_organizacion
torganizacionE.php	Mostrar los datos de un tipo de organización a eliminar	seguridad.php adodb.inc.php	tipo_organizacion
torganizacionEr.php	Eliminar un tipo de organización	seguridad.php adodb.inc.php	tipo_organizacion
torganizacionM.php	Mostrar los datos de un tipo de organización a modificar	seguridad.php adodb.inc.php	tipo_organizacion
torganizacionMr.php	Modificar los datos de un tipo de organización	seguridad.php adodb.inc.php	tipo_organizacion
torganizacionN.php	Capturar los datos de un tipo de organización	seguridad.php	
torganizacionNr.php	Guardar los datos de un tipo de organización	seguridad.php adodb.inc.php	tipo_organizacion

## **CONCLUSIONES DEL DIPLOMADO**

En el Diplomado se cubrieron todos los objetivos planteados al inicio del mismo, lo cual contribuyó a contar con los conocimientos y habilidades necesarias para el diseño, desarrollo e implementación de un sistema de información basado en Web.

Por lo que se refiere al caso práctico, el mismo pudo diseñarse y comenzar su diseño con base a los conocimientos adquiridos y aplicados durante la duración del Diplomado.

## **ANEXO A. SERVICIOS DEL MTYSP**

Listado de servicios por materia del Manual de Trámites y Servicios al Público, publicado en la Gaceta Oficial del Distrito Federal No 44-Bis del día 3 de Junio del 2003 con la modificación del 30 de Julio del 2004 publicado en la Gaceta Oficial del Distrito Federal No 74-Bis.

### **Agraria**

Asesoría jurídica y conciliación agraria asesoría  
Organización y capacitación agraria  
Regularización de la propiedad privada rural  
Trabajos técnicos topográficos y expedición de opinión técnica 5 días hábiles

### **Agua Potable y Servicios Hidráulicos**

Atención a faltas de agua  
Atención de fugas en medidor y/o cuadro  
Desazolve de drenaje  
Suministro de agua en carros-tanque  
Suministro de agua potable y residual tratada para uso no doméstico en toma tipo cuello de garza

### **Asesoría Telefónica**

Apoyo en la localización de personas extraviadas reportadas a Locatel vía telefónica  
Asesoría jurídica vía telefónica  
Asesoría medica vía telefónica sobre toxicología y urgencias médicas  
Asesoría psicológica vía telefónica  
Comprobación de órdenes de visita a establecimientos mercantiles vía telefónica  
Información general vía telefónica de códigos postales  
Información general vía telefónica sobre el índice metropolitano de la calidad del aire (IMECA)  
Información general vía telefónica sobre el programa integrado territorial (PIT)  
Información vía telefónica sobre el programa de donación de órganos y tejidos  
Información vía telefónica sobre el programa hoy no circula  
Información vía telefónica sobre instituciones públicas y privadas  
Información vía telefónica sobre precios de productos y servicios básicos  
Información vía telefónica sobre verificación vehicular  
Información y reportes vía telefónica sobre vehículos extraviados  
Línea ciudadana  
Otorgamiento vía telefónica de línea de captura para el pago de impuestos, derechos y recargos en bancos  
Quejas vía telefónica sobre servicios de autotransporte urbano  
Quejatel  
Reporte telefónico sobre la probable comisión de algún acto ilícito  
Reporte vía telefónica de daños a la infraestructura urbana 39  
Reporte vía telefónica de emergencias o desastres 41  
Tramitel: información vía telefónica sobre tramites y servicios del Gobierno del Distrito Federal



### Asesoría y Asistencia Social

Asesoramiento por parte de la procuraduría social del D. F. en asambleas de condominios  
Asesoría, orientación y patrocinio de juicios en materia de Derecho Familiar  
Atención de reportes ciudadanos para el rescate y localización de personas que vivan en la calle  
Atención del maltrato infantil  
Atención y tratamiento de personas con algún problema de adicción a las drogas  
Atención, orientación y quejas del programa de desayunos escolares  
Conciliación administrativa  
Conciliación condominal  
Curso general de orientación condominal  
Información y asistencia sobre los derechos y obligaciones de los tutores y curadores  
Orientación ciudadana  
Orientación condominal  
Orientación y apoyo económico a personas para la reintegración a su lugar de origen  
Orientación y atención telefónica para brindar asesorías de apoyo a personas y familiares en riesgo de calle

### Industria

Consulta automatizada telefónica del Distrito Federal (CONATEL-DF) para trámites empresariales  
Orientación y gestión de trámites empresarial

### Medio Ambiente

Asesoría y orientación a la población sobre el cumplimiento de la legislación ambiental y del ordenamiento territorial y su publicación por parte de las autoridades del Gobierno del Distrito Federal  
Atención a consultas sobre servicios públicos diversos: "Hoy no Circula"  
Atención a consultas sobre servicios públicos diversos: "IMECA"  
Atención a denuncia relacionada con la operación de los Verificentros autorizados en el Distrito Federal  
Atención a solicitudes de visitas a la red automática de monitoreo atmosférico (R.A.M.A)  
Atención de la denuncia ciudadana en materia de conservación del ambiente, protección ecológica y restauración del equilibrio ecológico, cuando se trate de fuente fijas del orden del Distrito Federal  
Atención y seguimiento a la denuncia ciudadana por la realización de programas, obras o actividades que pudieran estar sujetas a la previa autorización en materia de impacto ambiental y riesgo  
Aviso de cumplimiento de la sanción establecida por incumplir el Programa de vehículos contaminantes y unidades sin verificar  
Campamentos ecológicos  
Cursos de formación y capacitación en materia ambiental  
Denuncias ciudadanas por presuntos incumplimientos en la legislación ambiental y del ordenamiento territorial del Distrito Federal  
Donación de plantas



Curso para técnicos instructores en seguridad e higiene

Inspección a centros de trabajo derivado de quejas sobre incumplimiento a la normatividad laboral

Otorgamiento de las normas oficiales mexicanas en seguridad e higiene; y asesorar sobre la aplicación de las mismas en los centros de trabajo

Valoraciones médicas a menores de edad, aspirantes a trabajar

Turismo

Información turística de la Ciudad de México

Recorridos turísticos en la Ciudad de México

## BIBLIOGRAFÍA

- Manual de Trámites y Servicios al Público publicado en la Gaceta Oficial del Distrito Federal No 44-Bis del día 3 de Junio del 2003 con la modificación del 30 de Julio del 2004 publicado en la Gaceta Oficial del Distrito Federal No 74-Bis.
- Ley de Procedimiento Administrativo del Distrito Federal.
- Web Application Development with PHP 4.0, Tobias Ratschiller & Till Gerken, Ed. New Riders Publishing, 200.
- Web Database Applications with PHP & MySQL, Hugh E. Williams & David Lane, Ed. O'Rilley, 2002.
- Linux 6ª Edición, David Bandel & Rober Napier, Ed. Prentice Hall, 2001.
- MySQL/PHP Database Applications, Jay Greenspan & Brad Bulger, Ed. M&T Books, 2001.
- The Web Server Book
- LAN TIMES Guía de SQL, James R. Groff & Paul N. Weinberg, Ed. Osborne McGraw-Hill, 1988.
- Sam's Teach Yourself MySQL in 21 Days, Mark Maslakowski, Ed. Sams, 2000.
- Aprendiendo UML en 24 horas, Joseph Schmuller, Ed. Prentice Hall
- Introduction to the Unified Modeling Language, Terry Quatrani, UML Evangelist, 06/11/03 <http://www-306.ibm.com/software/rational/uml/>
- The Unified Modeling Language Reference Manual, James Rumbaugh, Ivar Jacobson & Grady Booch, Ed. Addison-Wesley, 1998
- UML y patrones, Craig Larman, Ed. Prentice Hall, 1999
- Metodología de Planificación y Desarrollo de Sistemas de Información, Métrica Versión 3, Ministerio de Administraciones Públicas, España. <http://www.csi.map.es>.
- Documentación del Proyecto GNU, Free Software Foundation. <http://www.gnu.org/>