



UNIVERSIDAD NACIONAL AUTONOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN

**"SISTEMA PARA EL REGISTRO DE LOS SERVICIOS
PROPORCIONADOS POR LA UNIDAD DE SISTEMAS
DEL TRIBUNAL ELECTORAL (SIRESE)"**

**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
INGENIERO EN COMPUTACIÓN
P R E S E N T A
JUAN CARLOS CARMONA MEDINA**

ASESOR: MTI. OMAR MENDOZA GONZÁLEZ



MÉXICO, 2005.

m. 346793



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

*A mi mamá y a mi papá por darme la vida,
una familia y la oportunidad de triunfar.*

*A mis hermanos Julio, Erika y David
por su invaluable presencia.*

INDICE

Introducción	5
Capítulo I. Entorno Institucional	7
1.1 El Tribunal Electoral	7
La Unidad de Sistemas del Tribunal Electoral	8
La infraestructura informática	10
1.2 Importancia del Registro de los Servicios Proporcionados	11
1.3 Planteamiento del Problema	13
1.4 Análisis de Factibilidad	14
Capítulo II. Fundamentos Teóricos	17
2.1 Sistemas Web	17
Historia de Internet	17
La Familia de Protocolos TCP/IP	18
Tecnologías para la implementación de un Sistema Web	20
2.2 Análisis de Sistemas	23
Análisis de Requerimientos	23
Modelado del Sistema	24
Casos de Uso	24
Diagramas Entidad-Relación	27
Normalización de la Estructura de una Base de Datos	29
Diccionario de Datos	31
2.3 Metodología de Tres Capas	35
Interfase de Usuario	37
Capa de Datos	40
Capa de Reglas de Negocio	43
2.4 Implementación	45
Servidor Web Apache	46
El Manejador de Base de Datos MySQL	48
Lenguaje de Programación: PHP	49

Capítulo III. Análisis del Sistema SIRESE	53
3.1 Casos de Uso.....	54
Diagrama Casos de Uso.....	54
Definición de Actores.....	55
Definición y Descripción de Casos de Uso.....	56
Referencias Cruzadas.....	63
3.2 Diagrama Entidad-Relación.....	64
3.3 Diccionario de Datos.....	65
Capítulo IV. Diseño e Implementación del Sistema SIRESE	71
4.1 Generalidades para la Implementación.....	71
4.2 Procedimiento para el Registro de los Servicios.....	72
4.3 Estructura y Descripción de los Módulos y Procesos del Sistema.....	74
Módulo de Seguridad.....	74
Módulo de Catálogos.....	76
Módulo de Administración de Servicios.....	77
Módulo Monitor de Servicios.....	78
Módulo de Reportes.....	80
4.4 Reglas de Negocio.....	81
Grupos y Roles.....	81
Clasificación de los Servicios.....	85
Conclusiones	89
Índice de Figuras	92
Índice de Tablas	93
Bibliografía	94
Bibliografía Electrónica	96

INTRODUCCIÓN

En la actualidad contamos con un novedoso y muy eficiente medio de comunicación que ha revolucionado la forma de transmitir información electrónica desde y hacia cualquier parte del mundo. A este medio de comunicación se le denominó Internet.

Internet es el conjunto de redes de computadoras que se encuentran interconectadas alrededor del mundo. Por medio de Internet se pueden utilizar algunos servicios para realizar actividades muy variadas como por ejemplo: enviar y recibir correo electrónico, publicar y leer noticias en línea, transferir archivos, etc.

El servicio más popular de Internet, sin lugar a dudas es el acceso a la World Wide Web, o Web como se le llama comúnmente. Este servicio nos ofrece la posibilidad de tener acceso a la información publicada en línea, es decir información que estará disponible para ser consultada, a cualquier hora, con la singularidad de que el contenido, además de ser multimedia estará organizado de tal forma que mediante enlaces elegiremos la información que nos interesa consultar, a este formato de información se le denomina "Hipertexto". Mediante mecanismos de hipertexto es posible implementar sistemas de información distribuidos. Hoy en día este servicio a evolucionado a tal grado que nos permite llevar a cabo operaciones financieras y de comercio electrónico en línea, desde cualquier computadora con acceso a Internet.

El objetivo de participar en el "Diplomado de Desarrollo de Sistemas en Web" fue el de adquirir los conocimientos y habilidades para analizar, diseñar e implementar sistemas de información en Web, aprovechando las ventajas que proporciona la Web para el intercambio de información.

Durante el desarrollo del Diplomado los conocimientos teóricos se pusieron en práctica llevando a cabo el análisis, diseño, desarrollo e implementación del Sistema para el Registro de los Servicios Proporcionados por la Unidad de Sistemas del Tribunal Electoral el cual, será implementado en Web.

En el primer capítulo y a manera de introducción se mencionan algunos aspectos de la institución para la cual será elaborado el sistema, en particular de la Unidad de Sistemas donde será implantado y cuyos integrantes serán los usuarios del sistema.

El segundo capítulo, denominado "Fundamentos Teóricos", está compuesto por la recopilación de información que sirve de marco teórico para el desarrollo del caso práctico. Este capítulo se divide en tres partes, en la primera se recopila información de los fundamentos para el desarrollo de Sistemas en Web, así como la fase de análisis y diseño de sistemas en general, abarcando el contenido del *Módulo 1: Fundamentos del Desarrollo de Sistemas en Web* y *Módulo 2: Análisis y Diseño de Sistemas Orientados al Web*, del diplomado. En la segunda parte se obtienen las bases para comprender la arquitectura de tres capas para, el diseño de un sistema de información, lo cual corresponde al contenido del *Módulo 3: Interfase de Usuario*, *Módulo 4: Capa de Datos* y *Módulo 5: Capa de Reglas de Negocio*. En la última parte de este capítulo se revisan las características principales del software que se va a utilizar para la implementación del sistema, llevado a cabo durante el desarrollo del *Módulo 6: Implementación*.

En el tercer y cuarto capítulo se presenta la documentación correspondiente al Análisis y Diseño del sistema denominado SIRESE, según corresponde, basado en los conocimientos adquiridos durante el desarrollo de los módulos del diplomado. Cabe señalar que el contenido del último capítulo en donde se presenta la documentación del Diseño e Implementación del sistema, únicamente contempla los aspectos considerados como fundamentales y sustanciales para el funcionamiento del sistema.

CAPÍTULO I ENTORNO INSTITUCIONAL

1.1 EL TRIBUNAL ELECTORAL

Durante el siguiente proyecto se desarrollará un sistema de información para el registro de los servicios proporcionados por la Unidad de Sistemas del Tribunal Electoral.

Para lo cual considero necesario mencionar algunos aspectos generales de la institución a la que hago referencia.

El Tribunal Electoral del Poder Judicial de la Federación es el organismo y autoridad encargada de interpretar, integrar y aplicar la ley en cuanto a materia electoral se refiere, en función de la solicitud de revocación de un acto o resolución que el demandante considere una violación a la ley.

Las funciones más importantes del Tribunal Electoral las podemos resumir de la siguiente forma: analizar y dar solución a las impugnaciones o instrumentos para modificar, revocar o anular las elecciones de senadores o diputados a nivel federal y de presidente; determinar o imponer sanciones en materia electoral o realizar el computo final de la elección de presidente entre otras, todo esto, con fundamento en la Constitución Política de los Estados Unidos Mexicanos, de una forma definitiva e irrevocable.

Como se puede apreciar las funciones del Tribunal Electoral son de suma importancia para el desarrollo, permanencia y transparencia de la democracia en nuestro país.

Debido a las características y dimensiones geográficas de México y con el objeto de descentralizar la impartición de justicia en cuanto a materia electoral se refiere, el Tribunal Electoral se conforma de una Sala Superior y cinco Salas Regionales.

La Sala Superior es permanente y está integrada por siete Magistrados Electorales cuya sede se encuentra ubicada en el Distrito Federal.

Las Salas Regionales funcionan únicamente durante el año de elecciones federales y cada una de ellas se integra por tres Magistrados Electorales; se ubican en las siguientes ciudades: Guadalajara, Monterrey, Xalapa, Distrito Federal y Toluca.

La Unidad de Sistemas del Tribunal Electoral.

El objetivo principal de la Unidad de Sistemas es atender los requerimientos relacionados con el procesamiento de información que demandan las actividades del personal del Tribunal Electoral proporcionando los servicios informáticos necesarios para eficientar las funciones que las distintas áreas llevan a cabo.

Las funciones y actividades de la Unidad de Sistemas, se enfocan principalmente en la adecuada administración de la infraestructura informática del Tribunal Electoral.

Los objetivos generales de las funciones que desempeña el personal de Sistemas son los siguientes:

- Diseñar e implementar sistemas de información necesarios para satisfacer los requerimientos para el procesamiento de información.
- Mantener el adecuado funcionamiento de los equipos y dispositivos de comunicación instalados en el Tribunal Electoral.
- Administrar los servicios de correo electrónico, servidores de información, etc.
- Capacitar a los usuarios del equipo de cómputo en el adecuado uso y manejo de los mismos.
- Definir e implementar las políticas informáticas y los estándares aplicables al diseño e implementación de redes de cómputo y al desarrollo de sistemas.

Estructura de la Unidad de Sistemas.

La Unidad de Sistemas se encarga de administrar y configurar los recursos informáticos, así como de proporcionar los servicios informáticos que demanden las actividades del personal del Tribunal Electoral.

Para que la infraestructura informática cumpla con su objetivo el personal de la Unidad de Sistemas es un equipo de profesionales encargados de mantener en óptimas condiciones su funcionamiento, para que los servicios que se proporcionan estén disponibles en todo momento, así como mantenerla a la vanguardia en cuanto al desarrollo tecnológico se refiere, para que el Tribunal Electoral cumpla con sus funciones eficientemente.

La importancia de conocer la estructura de la Unidad de Sistemas radica en el hecho de que el personal de sistemas, serán los usuarios del sistema, estarán agrupados con la finalidad de que cada departamento genere reportes con la información de los servicios que proporcione. Esta individualidad es importante ya que cada departamento funciona de manera independiente, además será más sencillo detectar fallas en la operación de cada departamento.

Departamento de Atención a Usuarios.

Los integrantes del departamento de Atención a Usuarios son los encargados de atender en sitio a los usuarios para solucionar algún problema que se presente en el funcionamiento de los equipos de cómputo o asesorar al usuario en el uso y manejo del equipo. También son los encargados de configurar e instalar los equipos de cómputo.

Otra de las actividades de este departamento es el recibir las solicitudes de servicio de parte de los usuarios y canalizarlas al departamento correspondiente, aspecto importante, a tomar en cuenta, en el diseño del sistema, lo cual se analizará a detalle en el diagrama de flujo del registro de los servicios¹.

Departamento de Centro de Computo.

Los integrantes de este departamento administran los servidores de archivos, correo electrónico, etc.

¹ Ver Capítulo IV. Diseño e Implementación del Sistema SIRESE.

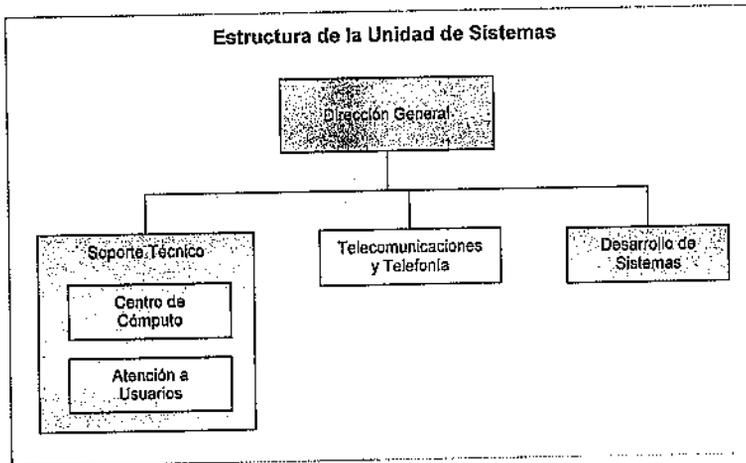


Figura 1. Estructura de la Unidad de Sistema

Departamento de Telefonía.

El personal del Departamento de Telecomunicaciones y Telefonía, son los encargados de mantener en buenas condiciones el funcionamiento de la red de voz y datos del Tribunal, así como de los enlaces con las salas regionales.

Departamento de Desarrollo de Sistemas.

Son los encargados de diseñar y desarrollar los sistemas de información del Tribunal, así como de dar mantenimiento a los ya existentes.

La infraestructura informática.

La infraestructura informática provee de servicios y herramientas de cómputo para el procesamiento de información, al personal del Tribunal Electoral para llevar a cabo sus actividades.

La infraestructura informática se compone de:

- Equipos de cómputo. Computadoras personales, terminales, impresoras, escáneres, etc.

- Red de Computadoras: Los equipos de cómputo están conectados a una red de computadoras para compartir información y recursos. Cabe señalar que una red de computadoras es toda una subestructura de componentes y cableado para conformar una red de computadoras.
- Red Nacional de Cómputo: Es un enlace digital que se tiene con las Salas Regionales, mediante el cual se realiza la transferencia de archivos, y el acceso a los servicios de Intranet y correo electrónico.
- Acceso a Internet: Para la consulta de sitios Web.
- Intranet: Sitio Web de acceso restringido, utilizado para distribuir información clasificada, por ejemplo: Avisos importantes, información de las sesiones públicas, directorio, consultas al acervo de la biblioteca, etc.
- Correo Electrónico: Los usuarios del equipo de cómputo cuentan con el servicio de correo electrónico institucional.
- Soporte Técnico: Servicio que proporciona atención en sitio al usuario, para resolver algún mal funcionamiento del equipo de cómputo, o capacitar al usuario en el uso y manejo del equipo de cómputo y aplicaciones para llevar a cabo sus actividades. Así como llevar cabo el mantenimiento preventivo y correctivo del equipo.

1.2 IMPORTANCIA DEL REGISTRO DE LOS SERVICIOS PROPORCIONADOS.

Sin lugar a dudas existe una razón de suma importancia para registrar los servicios proporcionados por la Unidad de Sistemas. El registrar la información de los servicios proporcionados por la unidad de sistemas, permite generar una base de datos de donde se puede extraer conocimiento acerca del comportamiento de los sistemas de cómputo en la institución. Con lo que se pueden identificar los puntos vulnerables de la infraestructura informática y así tomar las medidas necesarias para corregir el problema. Por ejemplo, al registrar la información de los servicios proporcionados por la unidad de sistemas, se pueden generar reportes con el siguiente criterio: la frecuencia

de falla en las aplicaciones instaladas en las computadoras personales de algún área en específico, con los resultados obtenidos, elaborar un estudio acerca de estas aplicaciones para verificar su compatibilidad con otras aplicaciones o con el sistema operativo, así como, determinar si lo que realmente se requiere es que el usuario conozca la forma adecuada de utilizar la aplicación, programando cursos para este fin.

Por lo anterior es importante registrar la información para que los esfuerzos realizados por la unidad de sistemas en pro de brindar un mejor servicio a los usuarios se fundamente en estadísticas generadas a partir de información real, es decir, resultado del comportamiento de los equipos de cómputo de acuerdo a los requerimientos de los usuarios, así como, medir el conocimiento de los usuarios para reforzar la capacitación particularmente en el área donde se presente el mayor número de problemas, permitiendo, conocer hacia dónde se deben dirigir los esfuerzos y que tipo de medidas tomar, así como que tipo de investigaciones realizar, en cuanto a nuevas tecnologías se refiere para continuar a la vanguardia, siempre basándose en el conocimiento o experiencia generado a partir de las actividades diarias del personal del Tribunal Electoral.

Además de guiar el curso de la evolución de la infraestructura informática, el hecho de registrar los servicios proporcionados por la Unidad de Sistemas, servirá para mejorar el servicio que día a día se proporciona a los usuarios ya que será posible:

- Dar seguimiento a las solicitudes de servicio en tiempo real.
- Medir tiempos de respuesta.
- Conocer los principales requerimientos de los usuarios.
- Distribuir al personal de acuerdo a las cargas de trabajo.
- Distribuir las actividades de una forma más equitativa.

Otra de las ventajas de registrar los servicios que se proporcionan, es la de contar con una base de conocimiento la cual sirve para fijar estándares de solución a problemas comunes, para ser consultada por personal de nuevo ingreso, sirviendo como un medio de entrenamiento. Sin dejar de mencionar que se cuenta con el historial de los servicios

proporcionados, lo que permitirá conocer antecedentes del comportamiento del equipo de cómputo así como de los usuarios ante ciertas eventualidades.

1.3 PLANTEAMIENTO DEL PROBLEMA

Actualmente no se cuenta con un medio para registrar los servicios proporcionados por la unidad de sistemas, lo cual trae consigo la siguiente problemática:

- No existe algún medio para saber que problemática se está presentando de acuerdo a los reportes de los usuarios, a menos que se mantenga constante comunicación con el personal encargado de recibir los reportes de los usuarios.
- Dar seguimiento a los reportes atendidos por parte de los supervisores es casi imposible, sobretodo con el personal encargado de atender en sitio a los usuarios ya que difícilmente se les encuentra en su lugar de trabajo.
- El conteo de los servicios proporcionados es llevado a cabo manualmente de la bitácora de cada asesor cuando se requiere generar un reporte.
- No existe un mecanismo de retroalimentación entre los integrantes de los departamentos de la Unidad de Sistemas.

Además de lo anterior la *Política Informática*² definida por la Unidad de Sistemas, señala como un requerimiento, establecer un mecanismo para registrar los servicios proporcionados por el mismo.

² Es el documento donde se establecen los lineamientos que se deberán aplicar en toda acción llevada a cabo en torno a la informática que opera en el Tribunal Electoral para optimizar los servicios de la Unidad de Sistemas.

Por lo cual se detecta la necesidad de implementar un sistema de información, destinado a cumplir los siguientes objetivos:

- Administrar el registro de los servicios proporcionados por la Unidad de Sistemas, permitiendo monitorear en tiempo real el estatus de los servicios solicitados por los usuarios, medir tiempos de repuesta, distribuir eficientemente al personal de acuerdo a las cargas de trabajo, en todas las sedes.
- Basar la toma de decisiones en información real, generada mediante los reportes del sistema, de acuerdo a los requerimientos de las actividades desarrolladas por el personal del Tribunal Electoral.
- Generar una base de conocimiento de donde se puedan generar los procedimientos de solución a problemas comunes o ayudar en la capacitación del personal de nuevo ingreso.

1.4 ANÁLISIS DE FACTIBILIDAD.

Para el desarrollo del sistema se tomaron en cuenta tres posibilidades: el desarrollo de una aplicación distribuida, el desarrollo de un sistema Web y la adquisición de una aplicación que cubriera la mayor parte de los requerimientos.

Es conveniente destacar que a pesar de las ventajas señaladas el proyecto para elaborar el sistema no tiene todo el apoyo de parte de la dirección de la Unidad de Sistemas, en cuanto al suministro de recursos materiales, ya que será un proyecto que no impactará directamente en las actividades sustanciales que desarrolla el personal del Tribunal Electoral y por lo tanto se decide destinar recursos para la adquisición de equipo o al desarrollo de un sistema que utilicen los usuarios.

Al inicio del análisis parecía muy factible el echo de generar una aplicación distribuida con Microsoft Visual Basic utilizando como manejador de bases de datos Microsoft SQL Server 2000, la institución cuenta con el licenciamiento correspondiente por lo que no

habría que destinar recursos para este fin, además de que el personal del Área de Desarrollo de Sistemas utiliza como estándar estas herramientas.

La alternativa no fue factible cuando se tomo en cuenta el hecho de que el personal encargado de la implantación debería desplazarse a las sedes para llevar a cabo la instalación del sistema y capacitar a los usuarios del mismo.

Por lo que se tomo en cuenta una segunda alternativa, desarrollar un sistema basado en Web, donde, mediante páginas dinámicas generadas con PHP se implementará la aplicación. Los datos estarían almacenados en una base de datos generada en el manejador de bases de datos MySQL, el cual no requiere licenciamiento. Para el lenguaje de páginas dinámicas PHP, como la base de datos MySQL se recomienda utilizar como plataforma el sistema operativo Linux en alguna de sus versiones, ya que fueron contruidos en base a su arquitectura, cabe señalar que también es de libre distribución y no requiere licenciamiento. El servidor que se tiene disponible para el sistema cumple con los requerimientos³ para funcionar con el sistema operativo Linux y soportar cincuenta clientes concurrentes mismos que se calculan como usuarios iniciales del sistema.

La tercer alternativa de solución, quizá la menos factible, es la de adquirir una aplicación, debido a las restricciones de asignar presupuesto al proyecto.

Se han analizado algunas propuestas, las cuales presentan una desventaja ya que por lo general la empresa que otorga el licenciamiento hace crear dependencia ya sea al tener que renovar licencias año con año para continuar recibiendo las actualizaciones del producto adquirido y el servicio de asistencia para el mantenimiento del mismo. Por ejemplo:

³ Para consultar las características del servidor, vea el Capítulo IV.- Diseño e implementación del Sistema SIRESE.

Cotización	
Nombre	Magig TSD
Compañía	Network Associates
Costo	\$150,000.00
Descripción	<ul style="list-style-type: none"> • Licenciamiento para 10 clientes • Soporte Técnico (1 año a partir de la fecha de inicio de la implantación) • Capacitación: 2 administradores, 5 usuarios. • Promoción especial para gobierno
Tiempo de Implantación	6 meses
Actualización o Renovación de la Licencia	\$50,000.00
Requerimientos Mínimos	<ul style="list-style-type: none"> - Servidor: Procesador a 600 Mhz., 512 Mb. RAM, 10Gb. Espacio libre en disco duro. - Sistema Operativo: Windows 2000 Server. - Microsoft SQL 2000 Estándar.

Tabla 1. Datos de la cotización de la empresa Network Associates.

Como se puede apreciar la opción más factible para el desarrollo del sistema es implementar un sistema en Web. El cual, de acuerdo al software seleccionado para la implementación, no requiere licenciamiento alguno y además su distribución no representa problema alguno. Los usuarios del sistema accederán desde un explorador de Internet mediante la red Nacional del Tribunal Electoral. Las configuraciones o cambios que se requieran realizar serán implementados por el personal de la Sala Superior donde se encontrara el servidor dedicado al sistema.

CAPÍTULO II FUNDAMENTOS TEÓRICOS

2.1 SISTEMAS WEB

Historia de Internet.

Internet nació a mediados de la década de los setenta, como resultado del interés del Departamento de Defensa de Estados Unidos para interconectar varias redes por medio de comunicación vía satélite y ondas de radio. Se buscaba mantener una red de información, sobre todo con fines de investigación militar pero además pensando en la eventualidad de una interrupción súbita de las comunicaciones, por ejemplo, en caso de una crisis bélica. La red creció y pronto fue dominada por la información que fluía de y hacia las universidades. El primer paso se había dado, en 1969, con la creación de ARPANET, por parte de la Agencia de Proyectos de Investigación Avanzados del Departamento de la Defensa (Advanced Research Projects Agency). La red, con el propósito de facilitar el trabajo de investigadores universitarios que estuvieran al servicio del Departamento de Defensa, conectaba a las computadoras de cuatro instituciones académicas: UCLA, el Instituto de Investigaciones de la Universidad de Stanford, la Universidad de Utah y la Universidad de California en Santa Barbara. La red creció, hasta que en 1983 fue denominada, aún con funciones militares, MILNET. Entonces, sucedió algo extraño: más que una vía para enviar archivos técnicos o documentos científicos, la red se convirtió en un medio de comunicación en línea.

La ARPANET original evolucionó hacia Internet. Internet se basó en la idea de que habría múltiples redes independientes, de diseño casi arbitrario, empezando por ARPANET como la red pionera de conmutación de paquetes, pero que pronto incluiría redes de paquetes por satélite, redes de paquetes por radio y otros tipos de red. Internet como ahora la conocemos encierra una idea técnica clave, la de una arquitectura abierta de trabajo en red. Bajo este enfoque, la elección de cualquier tecnología de red individual no respondería a una arquitectura específica de red sino

que podría ser seleccionada libremente por un proveedor e interactuar con las otras redes a través del metanivel de la arquitectura de *Internetworking* o trabajo entre redes.

La Familia de Protocolos TCP/IP.

Internet se basa en la utilización de los protocolos de comunicación TCP/IP que son las normas que posibilitan la interconexión de computadoras de diferentes plataformas, utilizando cualquier tipo de tecnología: Ethernet, líneas telefónicas conmutadas o dedicadas, X25, RDSI, etc.

Los protocolos más importantes son IP (Internet Protocol) y TCP (Transmission Control Protocol).

El Protocolo IP.

Define una red de conmutación de paquetes donde la información que se quiere transmitir está fragmentada en paquetes. Cada paquete se envía a la dirección del ordenador destino y viaja independientemente del resto. La característica principal de los paquetes IP es que pueden utilizar cualquier medio y tecnología de transporte.

El Protocolo TCP.

Se encarga de subsanar las deficiencias en la llegada de los paquetes de información a su destino, para conseguir un servicio de transporte fiable.

Este mecanismo de funcionamiento requiere que todas las computadoras conectadas a la red tengan direcciones distintas.

Las direcciones IP

Cada computadora conectada a la red tiene una dirección asociada denominada "Dirección IP". Estas direcciones son números de 32 bits que normalmente se escriben como a.b.c.d donde a.b.c.d son números menores de 255.

Una parte de la dirección (a y b) identifica la red entre todas las redes conectadas a Internet y la que utilizan los ruteadores para encaminar los paquetes. La otra parte de la dirección (c y d), identifica a cada computadora de las conectadas en la misma red.

Una dirección IP identifica una computadora. Las aplicaciones dentro de una computadora se identifican mediante un número contenido en la cabecera de los paquetes TCP/IP, llamado puerto.

Aunque podamos utilizar estas direcciones de Internet numéricas para acceder a los servicios y computadoras, normalmente utilizamos nombres que son más fáciles de recordar. Esto es posible mediante la utilización del servicio de nombres de Internet o DNS (Domain Name System) que traduce los nombres a direcciones numéricas.

El DNS es una base de datos distribuida de forma jerárquica por toda la red y que es consultada por las aplicaciones para traducir los nombres a direcciones numéricas. Esta jerarquía permite distribuir la responsabilidad de garantizar que no existen nombres repetidos dentro del mismo nivel o dominio ya que el administrador de cada nivel es responsable del registro de nombres dentro de su nivel y garantiza que éstos sean únicos.

Internet es un concepto, más que un producto o servicio. Es la infraestructura en la cual se asienta, se reproduce y extiende, el ciberespacio, es decir, el espacio creado por la comunicación entre computadoras. A ciberespacio, lo podemos definir como el espacio conceptual en donde palabras, relaciones humanas y datos, son manifestadas por la gente empleando tecnología de comunicaciones a través de computadoras.

Internet es un conjunto de servicios y el Web, que es lo que tratamos de analizar, no es más que uno de ellos. Probablemente sea el más popular en estos momentos y a veces parezca que Internet se limita al Web. La Web es entonces un servicio más que

consiste en un inmenso conjunto de páginas conectadas unas a otras por un sistema de enlaces.

Tecnologías para la implementación de un Sistema Web.

Definimos un sistema distribuido como aquel en el que los componentes hardware o software, localizados en computadoras conectadas a una red, comunican y coordinan sus acciones sólo mediante la transferencia de mensajes, esta definición sencilla cubre el rango completo de sistemas en los que se utilizan computadoras en red, por lo tanto los Sistemas en Web son sistemas distribuidos.

Los sistemas de información implementados mediante un sitio Web, tienen las siguientes características, es una tecnología fácil de aprender y se pueden enriquecer los contenidos con multimedia, lo cual significa una ventaja para que sean aceptados por el usuario.

Las páginas que componen un sistema en Web pueden ser dinámicas o estáticas, básicamente la diferencia radica en que las páginas dinámicas permiten interactuar con el usuario para intercambiar información, de la misma forma el contenido de la página Web cambiara de acuerdo a las opciones que el usuario seleccione. En cambio el contenido de las páginas Web estáticas puede contemplar animaciones y contenidos multimedia pero el contenido siempre es el mismo y no permite interactuar con el usuario.

Para implementar páginas Web dinámicas se pueden utilizar dos tipos de tecnologías una de ellas denominada de Servidor y la otra de Cliente.

Tecnologías de Cliente para la generación de páginas Web dinámicas.

En las páginas dinámicas que utilizan tecnologías del lado del cliente, toda la carga de procesamiento, de los efectos y funcionalidades la soporta directamente el navegador de la computadora desde donde se solicita la

consulta. Las tecnologías del lado del servidor, se utilizan para el control de ventanas, presentaciones en las que se pueden mover objetos por la página, para el control de formularios, cálculos, etc. El código necesario para crear los efectos y funcionalidades se incluye dentro del mismo archivo HTML y es llamado SCRIPT. Las páginas dinámicas de cliente se escriben principalmente en dos lenguajes de programación: JavaScript y Visual Basic Script (VBScript).

Tecnologías de Servidor para la generación de páginas Web dinámicas.

Las tecnologías del lado del servidor se emplean principalmente para la elaboración de sistemas en Web donde se requiere tener acceso a recursos externos al cliente, por ejemplo acceder a una base de datos y también por cuestiones de seguridad por ejemplo cuando por esta razón los cálculos no se pueden realizar en la computadora del usuario.

El código que le da dinamismo a las páginas Web que utilizan tecnologías de servidor se escribe en el archivo HTML, de tal manera que el código HTML y el código del lenguaje de programación estarán mezclados. Cuando una página es solicitada por parte de un cliente, el servidor ejecuta las instrucciones, procesa los datos y se genera una página con el resultado de todas las operaciones, que solamente contiene código HTML. Este resultado final es el que se envía al cliente y puede ser interpretado sin lugar a errores ni incompatibilidades, puesto que sólo contiene código HTML.

Para escribir páginas dinámicas de servidor existen varios lenguajes de programación: Common Gateway Interface (CGI) comúnmente escritos en Perl, Active Server Pages (ASP), Hipertext Preprocesor (PHP), y Java Server Pages (JSP).

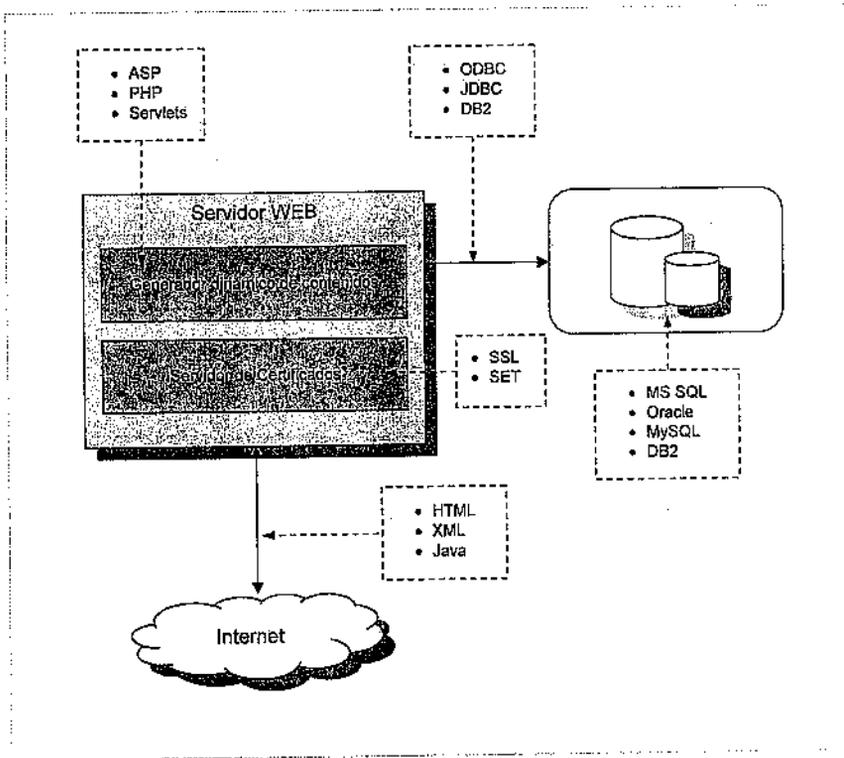


Figura 2. Diagrama de la arquitectura de implementación de las tecnologías de Cliente y de Servidor para páginas Web dinámicas.

2.2 ANÁLISIS DE SISTEMAS

*"Una buena especificación del sistema crea los cimientos en los cuales deben sostenerse el diseño y la programación."*⁴

Los objetivos principales del análisis de sistemas son los siguientes:

- Detectar las propiedades importantes del sistema.
- Definir los requerimientos con el usuario.
- Comprender perfectamente el ambiente del usuario y documentario, elaborando modelos del mismo.

Durante el análisis del sistema, se identifican las siguientes fases:

- Análisis de Requerimientos.
- Planteamiento del Problema.
- Modelado del Sistema.

Análisis de Requerimientos

- Delimitar el problema. Definiendo qué será parte del sistema y qué no. No debemos olvidar que *"todo sistema es parte de un sistema mayor"*.
- Identificar las interfases entre el sistema y el ambiente exterior, es decir, se necesita saber que información alimenta al sistema, desde el ambiente exterior, y a su vez que información se genera como respuesta, para retroalimentar al ambiente externo.
- También se definirán los acontecimientos que ocurren en el ambiente exterior al sistema, para los cuales deberá generar una respuesta.

⁴Yourdon, 1986.

Modelado del Sistema

Consiste en modelar el comportamiento final que el sistema debe tener para responder con certeza al ambiente donde será implantado. Con lo que se tratarán de definir y modelar las entradas de información, los almacenes de datos y la información de salida, así como los acontecimientos a los que el sistema reaccionará y el proceso que se llevara a cabo en respuesta a los datos de entrada. Para este proyecto se utilizaron las siguientes herramientas de modelado:

- Casos de Uso.
- Diagramas Entidad-Relación.
- Diccionario de datos.

Las cuales se describen a continuación:

Casos de Uso.

Los diagramas de *Casos de Uso* son una herramienta indispensable durante el análisis, ya que nos ayudan a entender los requerimientos del usuario. Son parte del Lenguaje Unificado de Modelado o UML por sus siglas en inglés *Unified Modelling Language*. Es un lenguaje gráfico para el modelado de sistemas, surge de la necesidad de tener un lenguaje común en la industria de la ingeniería de software.

Los Casos de Uso definen la arquitectura del sistema, modelan las interacciones entre actores y el sistema o subsistemas representados en varios niveles de abstracción. Se debe de tomar en cuenta que los casos de uso se utilizan para el análisis, diseño e implementación del sistema. La secuencia de mensajes y acciones que llevan a cabo los casos de uso son analizados y el sistema es diseñado e implementado para realizar específicamente esas acciones. Cada elemento es creado y usado por que contribuye en satisfacer los requerimientos del sistema. Al final los casos de uso se utilizan para elaborar la documentación del sistema ya que, en los casos de uso, también estará definido como utilizarán el sistema los usuarios finales.

Para emplear satisfactoriamente los diagramas de casos de uso es necesario conocer los tipos de elementos empleados en estos diagramas.

Actor.

El elemento *actor* es usado para modelar y representar los roles de los usuarios del sistema, incluyendo a personas, así como otros sistemas, que hicieran uso del sistema para su funcionamiento.

Para representar a los actores en los diagramas de casos de Uso se emplea la siguiente figura:



Figura 3. Elemento para representar un Actor en los diagramas de Casos de Uso.

Para identificar a un actor se puede considerar lo siguiente:

- Los actores son elementos internos o externos del sistema.
- Los actores interactúan con el sistema, de echo son los usuarios del sistema. Los actores se ven beneficiados directamente por la respuesta del sistema, pero a su vez los actores proveen información para que el sistema opere.

Casos de Uso.

Los elementos *Casos de Uso* son utilizados para modelar y representar procedimientos o servicios proporcionados por el sistema a los usuarios.

La figura para representar los casos de uso es la siguiente:

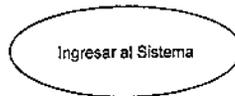


Figura 4. Ejemplo del elemento para representar un Caso de Uso.

Los *Casos de Uso* tienen las siguientes características:

- Son interacciones o diálogos entre el sistema y los actores, incluyendo los mensajes que intercambian entre ellos y las acciones llevadas a cabo por el sistema. Los *Casos de Uso* incluyen en su definición variantes de las secuencias del procedimiento que definen, incluyendo las alternativas a seguir.
- Los *Casos de Uso* son iniciados por los actores y pueden involucrar la participación de varios actores, de la misma forma un *Caso de Uso* entregará información a por lo menos un actor participante.
- Un *Caso de Uso* puede tener extensiones las cuales definen procedimientos muy específicos que no interactúan con otros casos de uso.
- A una instancia de *Caso de Uso* se le llama escenario representa interacciones específicas, dicho de otra forma representa secuencias de mensajes y acciones donde interactúan el sistema y el usuario.

Relaciones.

Los elementos relación se utilizan para representar la interacción entre los actores y los casos de uso, indicando que el *Actor* participa y se comunica con el sistema mediante el *Caso de Uso* involucrado en la relación.

Las relaciones se representan en los diagramas mediante líneas, las cuales pueden llevar una flecha en uno de sus extremos indicando quien inicia la comunicación, de tal forma, si la flecha apunta a un *Caso de Uso*, el actor que se encuentra en el otro extremo de la asociación inicia la comunicación con el sistema.

Diagramas Entidad-Relación

"Es un modelo que describe con alto nivel de abstracción la distribución de datos almacenados en un sistema"⁵, sin olvidar definir como están relacionados entre si.

Las estructuras de datos y las relaciones son tan complejas que se deben definir con precisión, para entenderse y no omitir detalles.

En los diagramas E-R se deja de lado el proceso que se llevara a cabo con los datos, basándose en definir la estructura de los datos y las relaciones existentes.

Elementos de los diagramas E-R.

- Entidades.

Las entidades son un conjunto de elementos del mundo real, se identifican por las siguientes características:

- Cada representante u. objeto de cada entidad se puede identificar de forma única, es decir, se puede identificar plenamente uno del otro.
- Las entidades juegan un papel necesario e indispensable en el sistema, de tal forma que si no se llegara a tomar en cuenta simplemente el sistema no podría funcionar.
- Cada entidad deberá poder describirse por uno o más atributos.

Toda entidad tiene existencia propia, es distinguible del resto de las entidades, tiene nombre y posee atributos definidos en un dominio determinado. Una entidad es todo aquello de lo que se desea almacenar información.

Para representar a las entidades se utiliza un rectángulo, titulado con el nombre de la entidad en singular, como se muestra en la Figura 5.

⁵ Yourdon, 1996.



Figura 5. Representación gráfica de una entidad.

- Atributos.

Serán características de las entidades con los cuales estas se describen. Podemos distinguir dos tipos de atributos, aquellos con los que se definirá la característica que hará único a cada atributo de los demás, a este atributo se le conoce como **llave primaria**. Otro tipo de atributo puede ser aquel mediante el cual se logre relacionar a dos entidades, a estos atributos se les conoce como **llaves foráneas**. El resto de los atributos serán solo una característica más de la entidad que definan.

Los atributos pueden estar representados por diferentes tipos de datos, de acuerdo a la información que definen. En forma muy general un atributo puede estar representado, por alguno de los siguientes tipos de datos:

Tipo de Dato	Definición	Símbolo
carácter	Puede contener cualquier tipo de carácter, generalmente se define un número máximo de los mismos.	(c)
numérico	Solo puede contener caracteres numéricos.	(n)
fecha	Valores de tipo fecha.	(f)

Tabla 2. Tipos de datos para definir a un atributo.

Finalmente cada atributo tendrá un nombre el cual será representativo de la característica de la entidad.

En la siguiente figura se muestra como se representan los atributos de una entidad en un diagrama Entidad-Relación.

USUARIOS		
c(15)	Clave	(PK)
c(30)	Nombre	
C(3)	ap_paterno	
c(30)	ap_materno	
c(45)	Puesto	
f	fecha ingreso	
n	telefono	
n	clave_area	(FK)

Figura 6. Representación de los atributos de una entidad en un diagrama E-R.

Se eligió realizar un diagrama E-R para modelar la base de datos del sistema ya que uno de los aspectos más importantes del sistema será almacenar la información generada para apoyar la toma de decisiones, por lo tanto esta información deberá estar estructurada de tal forma que los datos reflejen el comportamiento de la infraestructura informática de acuerdo a los requerimientos de los usuarios.

Normalización de una Base de Datos.

La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más sencillas, que además de ser más simples y más estables, son más fáciles de "mantener". También se puede entender la normalización como una serie de reglas que sirven para desarrollar un esquema que minimice los problemas de lógica.

Algunas ventajas que se obtienen al normalizar son las siguientes:

- Transforma un diagrama entidad relación para que sea más sencillo de entender.
- Simplifica al máximo una estructura de datos compleja.
- Una base de datos normalizada utilizará menos espacio para su almacenamiento.

Grados de normalización

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

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

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

Tabla 3. Breve descripción de los niveles de normalización.

Primera Forma Normal

La regla de la Primera Forma Normal establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.

Poner la base de datos en la Primera Forma Normal resuelve el problema de los encabezados de columna múltiples. Frecuentemente y sobretodo cuando no se tiene mucha experiencia en la elaboración de los diagramas Entidad-Relación, una y otra vez, se crearán columnas que representen los mismos datos. La normalización, en su primer nivel, ayuda a hacer más clara la base de datos y a organizarla en partes más pequeñas y más fáciles de entender. En lugar de tener que entender una tabla con todos los campos y que abarca muchos aspectos diferentes del ambiente analizado.

Segunda Forma Normal

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

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

Tercera Forma Normal

Una tabla estará normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes, por completo, de la llave primaria y no hay dependencias transitivas. Comentamos anteriormente que una dependencia transitiva es aquella en la cual existen columnas que no son llave que dependen de otras columnas que tampoco son llave.

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

No siempre es lo más recomendable normalizar una base de datos al más alto nivel, debido a que nos puede llevar a un nivel de complejidad que pudiera ser evitado si estuviera en un nivel más bajo de normalización.

Diccionario de Datos

Nos ofrece un modelo textual del sistema, es decir en el hallaremos las definiciones de la base de datos y comúnmente nos sirve como material de referencia.

El Diccionario de Datos es un listado organizado de todos los datos del sistema, con definiciones precisas y rigurosas, para que tanto el usuario como el analista comprendan todas las entradas y salidas, así como la información contenida en las bases de datos y los cálculos intermedios para obtener algún dato. En el diccionario también se definen los detalles de las relaciones entre entidades, del diagrama Entidad-Relación.

Podemos decir que el diccionario de datos define las reglas de negocio para definir los datos válidos.

Notación del diccionario de datos.

Es importante conocer el significado de los símbolos utilizados en el diccionario de datos para entender lo que se representa, por lo tanto a continuación se define la aplicación de estos símbolos.

Símbolo	Se lee
=	está compuesto de
+	Y
()	optativo
{}	iteración
[]	seleccionar una o varias alternativas
* ... *	comentario
@	identificador (campo llave)
	separa opciones alternativas

Tabla 4. Notación empleada en el diccionario de datos.

Para que la definición de un dato este completa, se deberá incluir lo siguiente:

- Significado del dato. Dentro del entorno del análisis del sistema. El comentario al respecto deberá aparecer entre asteriscos. Así como las unidades de medida y escala o rango del dato, especificando la precisión (por las posiciones decimales).
- Composición del dato. En caso de que se componga de partes elementales con significado.
- Valores que puede tomar el dato.

Datos Compuestos.

Un elemento o dato puede estar compuesto de varios datos básicos, los cuales ya no se descomponen en otras unidades. Usualmente esto es definido por la aplicación, se deberán definir conjuntamente con el usuario. Por ejemplo: el termino nombre puede descomponerse en nombre, apellido paterno y apellido materno.

Todo dependerá de cómo lo requiera el ambiente del usuario, en ocasiones el hecho de no descomponer el campo nombre significa que esto no es relevante para el ambiente del usuario.

Datos Optativos.

Son datos que pueden estar o no presentes en un dato compuesto, lo cual es importante definir en conjunto con el usuario y se debe cuidar mucho la notación para que no se mal interprete. Por ejemplo, la notación:

$$\text{domicilio del cliente} = (\text{domicilio envió}) + (\text{domicilio cuenta})$$

significa, que el domicilio del cliente consiste en:

- sólo un domicilio de envío
- sólo un domicilio para enviar cuentas
- uno para envío y otro para cuentas.

Iteración.

Se utiliza para representar la ocurrencia repetida de un dato. Mediante esta notación se pueden definir límites inferiores y superiores para un valor, la notación la podemos apreciar en el siguiente ejemplo:

$$a = 1\{b\}10$$

Lo cual significa que el elemento a deberá contener como mínimo un elemento b y como un máximo diez elementos b. Se puede especificar únicamente el límite superior o el inferior de acuerdo a los requerimientos.

Selección.

Esta notación indica que un dato consistirá en un elemento del conjunto de opciones que se describe. Las opciones se encierran entre corchetes y se separan por una línea vertical. Como ejemplo, consideremos lo siguiente:

sexo = [Femenino | Masculino]

En este caso al dato sexo, se le asociara el valor de Masculino o Femenino.

En conclusión:

- La información sobre los datos se almacena de un modo centralizado. Esto ayuda a mantener el control sobre los datos.
- El significado de los datos se puede definir, lo que ayudará a los usuarios a entender el propósito de los mismos.
- La comunicación se simplifica ya que se almacena el significado exacto. El diccionario de datos también puede identificar al usuario o usuarios que poseen los datos o que los acceden.
- Las redundancias y las inconsistencias se pueden identificar más fácilmente ya que los datos están centralizados.
- Se puede tener un historial de los cambios realizados sobre la base de datos.
- El impacto que puede producir un cambio se puede determinar antes de que sea implementado, ya que el diccionario de datos mantiene información sobre cada tipo de dato, todas sus relaciones y todos sus usuarios.
- Se puede hacer respetar la seguridad.
- Se puede garantizar la integridad.
- Se puede proporcionar información para auditorías.

2.3 METODOLOGÍA DE TRES CAPAS

Como resultado de las limitaciones de la arquitectura basado en compartir archivos, surgió la arquitectura cliente/servidor. Esta propuesta introduce un servidor de bases de datos para reemplazar al servidor de archivos. Utilizando un manejador de bases de datos relacionales, donde el usuario hacía las búsquedas directamente. La arquitectura cliente/servidor redujo el tráfico en la red, transfiriendo únicamente el resultado de un *query* o consulta a la base de datos, preferible a transferir el archivo en su totalidad. Mejorando la actualización simultánea, interactuando con el sistema mediante una interfaz de usuario gráfica con una base de datos compartida. En la arquitectura cliente/servidor los llamados a procedimientos remotos o la utilización de sentencias SQL son el medio para comunicar al cliente con el servidor.

La arquitectura de software cliente/servidor es versátil, y con una infraestructura modular, con ella se introducen los términos de usabilidad, flexibilidad, interoperabilidad y escalabilidad comparado con la arquitectura centralizada de los *mainframe* o de los rudimentarios sistemas basados en compartir archivos.

En una arquitectura cliente/servidor de dos capas, la interfaz del usuario del sistema se encuentra instalada en la computadora del usuario y los servicios para la administración o manejo de la base de datos se encuentra en el servidor el cual es una computadora más robusta para atender las solicitudes de varios clientes a la vez. La administración de procesos esta dividida entre el ambiente de la interfaz de usuario y el ambiente del administrador del servidor de la base de datos, por ejemplo se pueden implementar procedimientos almacenados y desencadenadores de eventos, que ayudan con la implementación de las reglas de negocio.

La arquitectura cliente/servidor de dos capas es una buena solución para sistemas distribuidos cuando el número de usuarios se contempla desde 10 hasta 100 usuarios interactuando de manera simultánea con el sistema. Cuando el número de usuarios excede los cien, el rendimiento comienza a ser deficiente. Esta limitación es resultado

de que el servidor mantiene una conexión mediante mensajes *keep-alive*⁶ con cada cliente, aunque no se este interactuando con la sesión.

Otra limitante de la arquitectura de dos capas es la implementación del servicio de administración de procesos (reglas de negocio), utilizando únicamente los procedimientos para interactuar con la base de datos proporcionados por el fabricante del software utilizado para el desarrollo de la interfaz de usuario, restringiendo la flexibilidad y libre elección del Sistema Gestor de Bases de Datos (SGBD) para la aplicación.

Finalmente las implementaciones que se basan en la arquitectura de dos capas tienen una limitada flexibilidad en cuanto a traspasar, reutilizar o modificar la funcionalidad de los programas de un servidor a otro sin regenerar manualmente el código de los procedimientos.

La arquitectura de tres capas surgió alrededor de los años noventa para superar las limitaciones de la arquitectura de dos capas. En la arquitectura de tres capas, se agrega una capa entre la interfaz de usuario y del manejador de la base de datos. Esta capa provee la administración de procesos de la aplicación mediante la implementación de la lógica y reglas de negocio, exclusivamente, en esta capa. Mediante la implementación de funciones como: cola de procesos, ejecución de aplicaciones, y varias vistas de la base de datos, se incrementa el rendimiento de la aplicación. Además, la flexibilidad en cuanto a la utilización de varios lenguajes del lado del servidor o del cliente, incluso para extraer datos de diferentes manejadores de bases de datos, empleando el mismo código; son otras de las ventajas de aplicar esta tecnología. Otra de ellas, la reusabilidad, ya que los procedimientos definidos se pueden reutilizar en todos los módulos de la aplicación, además de proporcionar ventajas en la etapa de mantenimiento, ya que el agregar o eliminar componentes o procesos al sistema se realizará de manera sencilla y sobre todo sin afectar a lo ya implementado.

⁶ Tiempo máximo (en segundos) que permitirá una conexión sin uso ("idle") antes de ser desconectada.

Gracias a estas ventajas se tiene la posibilidad de soportar a más de cien usuarios, comparando con la arquitectura de dos capas la cual únicamente soportaba a lo más 100 usuarios concurrentes, que seguramente son los estimados para un sistema en Web, donde una de sus mejores características es la facilidad de ser distribuido.

La implementación de una arquitectura de tres capas no siempre es la más sencilla, en ocasiones el lograrlo será más complicado y costoso pero sin lugar a dudas las ventajas mencionadas pueden valer el esfuerzo extra requerido.

Interfase de Usuario

También conocida como *Capa de Presentación*, contempla las ventanas o páginas Web del sistema, es la parte con la cual interactuará el usuario, y podemos decir que los elementos que la componen encapsulan la lógica de navegación y control de eventos de la interfase.

Los sistemas en Web al ser diseñados deben seguir los principios de *usabilidad*. Podemos definir a la *usabilidad* como las técnicas que ayudan a los seres humanos a realizar tareas en entornos gráficos de computadora, para lo cual se deben tomar en cuenta tres conceptos muy importantes. El sistema estará destinado para seres humanos, que quieren realizar una tarea de una forma sencilla y eficaz y en este caso en particular, la deben realizar frente a una computadora en un entorno gráfico, en este caso la Web. La usabilidad ayuda a que esta tarea se realice de una forma sencilla, analizando el comportamiento humano, y los pasos necesarios para ejecutar la tarea de una forma eficaz.

Principios generales de usabilidad en Sistemas en Web:

1. Anticipación, el sitio Web debe anticiparse a las necesidades del usuario.
2. Autonomía, los usuarios deben tener el control sobre el sitio Web. Los usuarios sienten que controlan un sitio Web si conocen su situación en un entorno abarcable y no infinito.
3. Los colores han de utilizarse con precaución para no dificultar el acceso a los usuarios con problemas de distinción de colores.
4. Consistencia, las aplicaciones deben ser consistentes con las expectativas de los usuarios, es decir, con su aprendizaje previo.
5. Eficiencia del usuario, los sitios Web se deben centrar en la productividad del usuario, no en la del propio sitio Web. Por ejemplo, en ocasiones tareas con mayor número de pasos son más rápidas de realizar para una persona que otras tareas con menos pasos, pero más complejas.
6. Reversibilidad, un sitio Web ha de permitir deshacer las acciones realizadas
7. Ley de Fitts indica que el tiempo para alcanzar un objetivo con el ratón esta en función de la distancia y el tamaño del objetivo. A menor distancia y mayor tamaño más facilidad para usar un mecanismo de interacción.
8. Reducción del tiempo de latencia. Hace posible optimizar el tiempo de espera del usuario, permitiendo la realización de otras tareas mientras se completa la previa e informando al usuario del tiempo pendiente para la finalización de la tarea.
9. Aprendizaje, los sitios Web deben requerir un mínimo proceso de aprendizaje y deben poder ser utilizados desde el primer momento.
10. El uso adecuado de metáforas facilita el aprendizaje de un sitio Web, pero un uso inadecuado de estas puede dificultar enormemente el aprendizaje.
11. La protección del trabajo de los usuarios es prioritario, se debe asegurar que los usuarios nunca pierden su trabajo como consecuencia de un error.
12. Legibilidad, el color de los textos debe contrastar con el del fondo, y el tamaño de fuente debe ser suficientemente grande.
13. Seguimiento de las acciones del usuario. Conociendo y almacenando información sobre su comportamiento previo se ha de permitir al usuario realizar operaciones frecuentes de manera más rápida.
14. Interfaz visible. Se deben evitar elementos invisibles de navegación que han de ser inferidos por los usuarios, menús desplegados, indicaciones ocultas, etc.

Tabla 5. Principios Generales de Usabilidad.

Al lenguaje con el que está construido el Web se le denomina hipertexto y es un entramado de páginas conectadas con enlaces. Los sistemas de hipertexto se utilizan en otros contextos, como por ejemplo los tan populares motores de búsqueda. La Web no solo se limita a presentar textos y enlaces, sino que también puede ofrecernos imágenes, videos, sonido y todo tipo de presentaciones, llegando a ser el servicio más rico en medios que tiene Internet. Por esta razón, para referirnos al sistema que implementa el Web (hipertexto), se ha empleado el término: multimedia.

Básicamente un archivo HTML esta constituido por un conjunto de etiquetas que sirven para definir la forma en la que se presentará el texto y otros elementos de la página.

Este lenguaje se escribe en un archivo de texto, para lo cual necesitamos un editor de textos para escribir una página Web. Así, el archivo donde está contenido el código HTML es un archivo de texto, con extensión .html o .htm, es indiferente cuál utilizar.

El lenguaje HTML consta de etiquetas que tienen la siguiente forma: o <P>. Cada etiqueta define una característica del texto al que hace referencia, por ejemplo significa que se escriba en negrita (bold) o <P> significa un párrafo, <A> es un enlace, etc. Casi todas las etiquetas tienen su correspondiente etiqueta de cierre, que indica que a partir de ese punto no debe de afectar la etiqueta. Por ejemplo se utiliza para indicar que se deje de escribir en negrita. Así que el HTML no es más que una serie de etiquetas que se utilizan para definir la forma o estilo que queremos aplicar a nuestro documento.

Por ejemplo, el siguiente archivo de texto con el nombre "ejemplo.html" y cuyo contenido es el siguiente:

```
<html>
<title>Ejemplo</title>
<B>Esto está en negrita</B>
</html>
```

Figura 7. Contenido del archivo ejemplo.html

Al cargar este archivo html en el explorador de Internet genera el resultado mostrado en la figura 8.

La interfaz de usuario del sistema, contiene los servicios del usuario como son: la sesión de trabajo, donde puede identificarse y personalizar su entorno; campos para la introducción de información, diálogos y el manejo de la transición de pantallas que conformarán la aplicación.

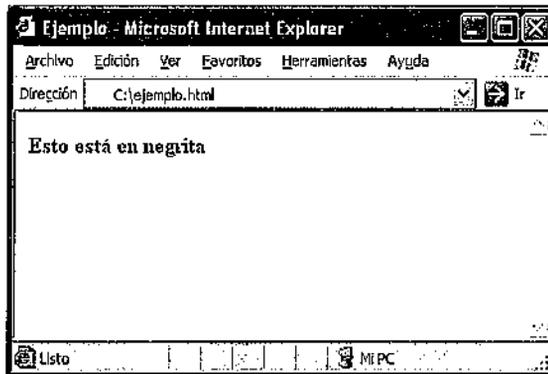


Figura 8. Vista del archivo ejemplo.html interpretado por un visor de páginas Web.

Capa de Datos

La capa de acceso a datos contiene componentes que interactúan con la base de datos, así como a un conjunto de componentes estáticos que permiten uniformar las operaciones de acceso a datos a través de un único conjunto de métodos.

Los componentes de la capa para la administración de los datos deben asegurar que los datos serán consistentes a lo largo de su distribución y almacenamiento mediante la utilización de medidas para: la encriptación de datos, consistencia y replicación. Así mismo como parte de las características de la arquitectura, la conectividad entre capas debe tener la característica de cambiarse dinámicamente dependiendo de las necesidades del usuario en cuanto a la búsqueda de información y servicios de datos.

La Capa de Datos provee la funcionalidad para la administración de la base de datos y está dedicada a los servicios de archivos y datos los cuales pueden ser optimizados utilizando un Sistema Gestor de Bases de Datos apropiado.

Las funciones de un Sistema Gestor de Bases de Datos (SGBD) son las siguientes.

La función fundamental es proporcionar a los usuarios la capacidad de almacenar datos en la base de datos, acceder a ellos y actualizarlos, así como proporcionar un catálogo en el que se almacenen las descripciones de los datos y que sea accesible por los usuarios autorizados. Este catálogo es lo que se denomina **diccionario de datos** y contiene información que describe los datos que componen la base de datos es decir los *metadatos*.

Un SGBD deberá contar con un mecanismo que garantice que todas las actualizaciones correspondientes a una determinada transacción se realicen, o que no se realice ninguna. Podemos definir a una *transacción* como un conjunto de acciones que cambian el contenido de la base de datos. Si la transacción falla durante su realización, por ejemplo porque falla el hardware, la base de datos quedará en un estado inconsistente. Algunos de los cambios se habrán hecho y otros no, por lo tanto, los cambios realizados deberán ser deshechos para devolver la base de datos a un estado consistente.

Un SGBD debe proporcionar un mecanismo que asegure que la base de datos se actualice correctamente cuando varios usuarios la están actualizando de manera concurrente. Uno de los principales objetivos de los SGBD es el permitir que varios usuarios tengan acceso concurrente a los datos que comparten. El acceso concurrente es relativamente fácil de gestionar si todos los usuarios se dedican a leer datos, ya que no pueden interferir unos con otros. Sin embargo, cuando dos o más usuarios están accediendo a la base de datos y al menos uno de ellos está actualizando datos, puede interferir de modo que se produzcan inconsistencias en la base de datos. El SGBD se debe encargar de que estas interferencias no se produzcan en el acceso simultáneo.

En cuanto a seguridad un SGBD deberá contar con un mecanismo que garantice que únicamente los usuarios autorizados pueden acceder a la base de datos. La protección debe ser contra accesos no autorizados, tanto intencionados como accidentales.

Un SGBD debe proporcionar los medios necesarios para garantizar que tanto los datos de la base de datos, como los cambios que se realizan sobre estos, sigan ciertas reglas. La integridad de la base de datos requiere la validez y consistencia de los datos almacenados. La integridad se refiere a la calidad de los datos. Normalmente se expresa mediante restricciones, que son una serie de reglas que la base de datos no puede violar. Por ejemplo, se puede establecer la restricción de que cada *asesor* no puede estar asignado a más de un *grupo de trabajo*. En este caso sería deseable que de forma automática, el SGBD controlara que no se sobrepase este límite, cada vez que un *asesor* ya se encuentre asignado a un *grupo*.

También será importante que el SGBD incorpore como herramienta fundamental dos lenguajes, uno para la definición y otro para la manipulación de los datos. El lenguaje de definición de datos (DDL, Data Definition Language) provee de los medios necesarios para definir los datos con precisión, especificando las distintas estructuras.

El lenguaje de manipulación de datos (DML, Data Manipulation/ Management Language), que es el encargado de facilitar a los usuarios el acceso y manipulación de los datos, encargándose de la recuperación de los datos almacenados, de la inserción y supresión de datos en la base de datos, y de la modificación de los existentes.

Debido a la diversidad de lenguajes y de bases de datos existentes, la manera de comunicarse entre unas y otras sería realmente complicada de culminarse a no ser por la existencia de estándares que nos permiten realizar las operaciones básicas de una forma universal.

Ese es el objetivo del Structured Query Language (SQL) que no es más que un lenguaje estándar de comunicación con bases de datos. Por lo tanto hablamos de un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje (ASP o

PHP) en combinación con cualquier tipo de base de datos: MS Access, SQL Server, MySQL, etc.

Cabe señalar que SQL contempla instrucciones para la definición y la manipulación de los datos.

Capa de Reglas de Negocio

La capa de reglas de negocio encapsula la lógica de procesos. Podemos agrupar los servicios de esta capa en dos tipos de componentes: Las *entidades empresariales* y los *componentes empresariales*.

Las entidades empresariales, que representan objetos que van a ser manejados o consumidos por toda la aplicación, estos podrían ser un modelo de objetos, datasets con tipo o estructuras de datos, que permitan representar objetos que han sido identificados durante el modelado del sistema.

Los otros tipos de objetos son los componentes empresariales que contienen la lógica de negocio, y en algunos casos son los objetos que definen las transacciones.

En la capa de regla de negocios se definen los procesos de todo tipo, procesos para efectuar cálculos, procesos para el monitoreo del valor de las variables, procesos de búsqueda o consultas de información. Los cuales pueden ser utilizados una o varias veces por la aplicación o inclusive por otras aplicaciones.

Como se puede observar el centralizar la lógica de procesos en una capa de la arquitectura nos ofrece mayor rendimiento, flexibilidad, resusabilidad y escalabilidad, debido a que la administración y el manejo de cambios son más sencillos ya que la funcionalidad del sistema solo se localiza en una parte y es ahí donde se deben llevar a cabo las modificaciones para que se reflejen en todo el sistema. Utilizando otro tipo de

arquitectura, la modificación a una función o servicio se tendría que realizar en todos y cada uno de los procesos que estuvieran relacionados a lo largo de toda la aplicación.

Además, mediante la capa de procesos, se pueden implementar controles auxiliares para el acceso concurrente a datos y para administrar las transacciones, asegurando la integridad de la información, incluso implementar reglas de negocio para la administración de bases de datos distribuidas.

Otra de las ventajas, provee acceso a recursos basados en nombre en lugar de archivos, lo cual deriva las ventajas de escalabilidad y flexibilidad al agregar o eliminar componentes del sistema.

2.4 IMPLEMENTACIÓN

Por las características que ya se mencionaron se decidió integrar el sistema utilizando las siguientes tecnologías, como nos podemos dar cuenta, existe una coherencia entre el Servidor Web Apache, el Sistema Manejador de Base de Datos MySQL y el lenguaje de programación PHP, lo cual facilitará la integración de la aplicación a la hora de implementar el sistema.

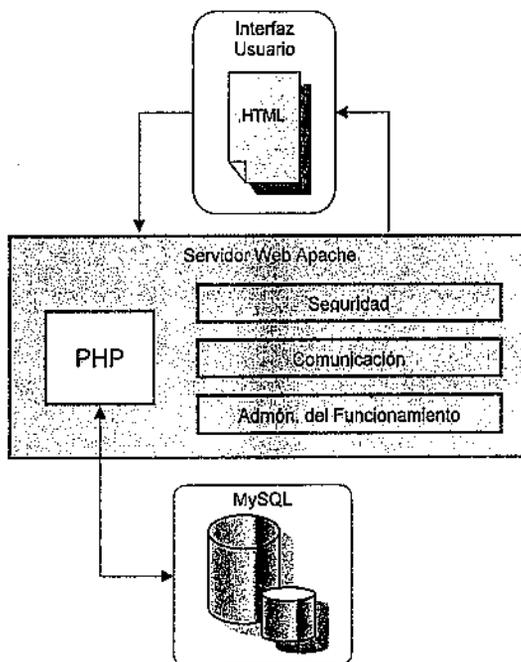


Figura 9. Arquitectura utilizada para la implementación del sistema SIRESE.

A continuación se presentan las características más importantes del Servidor Web, Base de Datos y Lenguaje de Programación utilizados para implementar el sistema.

Servidor Web Apache.

El servidor Web Apache es el producto resultado de un proyecto de la "Apache Software Foundation" para crear un servidor Web estable, fiable y veloz; enfocado, principalmente a plataformas Unix.

Es un servidor Web de código abierto, cuando comenzó su desarrollo en 1995 se basó inicialmente en el código del popular servidor Web NCSA HTTPd 1.3, aunque más tarde fue codificado por completo.

Algunas de las características más importantes que proporciona el servidor Web Apache son las siguientes:

- Es compatible con la última versión del protocolo HTTP/1.1.
- Permiten que se le agreguen módulos de terceros para aumentar sus funciones y alcances, por ejemplo:

Nombre del Módulo	Descripción
mod_perl	Páginas dinámicas en Perl.
mod_php	Páginas dinámicas en PHP.
mod_python	Páginas dinámicas en Python.
mod_jk	Conector para enlazar con el servidor Jakarta Tomcat de páginas dinámicas en Java (servlets y JSP).
mod_ssl	Comunicaciones Seguras.
mod_rewrite	Reescritura de direcciones servidas.

Tabla 6. Módulos adicionales para el Servidor Web Apache.

Así mismo se pueden escribir módulos utilizando el módulo API de Apache.

- La licencia no impone restricciones.
- Es multiplataforma ya que se puede implementar en plataformas Windows, Netware, OS2 y varias versiones de UNIX.
- Se encuentra continuamente en desarrollo.

- Cuenta con la retroalimentación de los usuarios para implementar nuevas ideas, corregir errores y generar parches.
- Permite accesos controlados a las páginas, la seguridad se maneja a través de una base de datos, admitiendo a un número considerable de usuarios concurrentes.
- Las repuestas a errores y problemas al enviar un archivo se pueden configurar, cambiando la leyenda que se presenta por *default* o si se decide ejecutar un script que proporcione un diagnóstico para el usuario.

Aunque aun es muy criticado por la falta de una interfaz gráfica que ayude en su configuración, en la actualidad, Apache es el servidor Web más utilizado, se encuentra implementado en el 68% de los sitios Web activos en el mundo, como se puede apreciar en las siguientes gráficas.

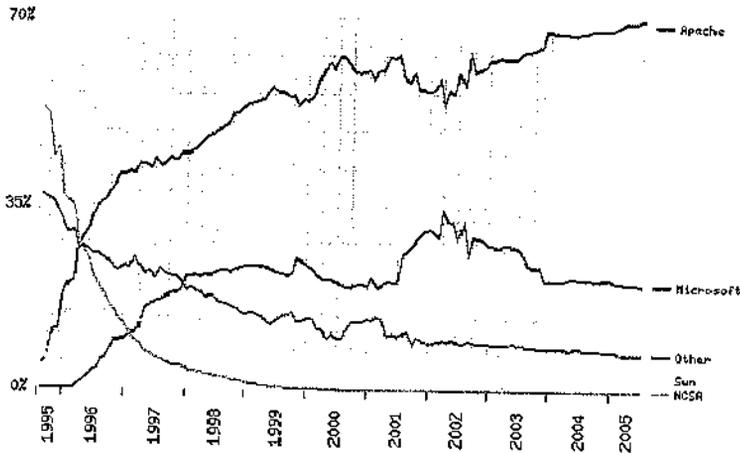


Figura 10. Porcentaje de los Servidores Web implementados en los dominios de Internet (1995 - 2005).⁷

⁷ Fuente: <http://news.netcraft.com/> - June 2005 Web Server Survey.

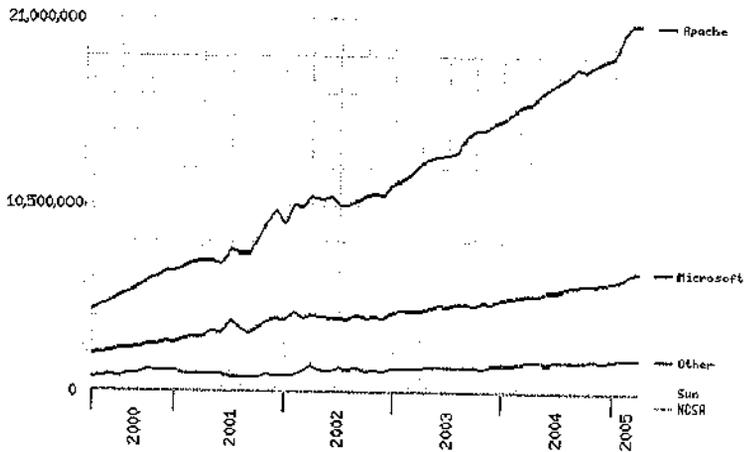


Figura 11. Total de Servidores Web activos. Junio 2000 – Junio 2005⁸

Para el proyecto es la mejor elección ya que se ha tomado la decisión de implementar el sistema utilizando software de libre distribución por lo que se trabajara sobre la plataforma del Sistema Operativo Linux, donde Apache es la mejor elección hablando de servidores de Internet.

El Manejador de Base de Datos MySQL.

Los aspectos ya mencionados en el apartado "Capa de Datos" de la sección 2.3 "Metodología de Tres Capas", se tomaron en cuenta para seleccionar el Sistema Manejador de Bases de Datos (SMBD) para el proyecto, concluyendo que MySQL cumple con la mayoría de ellos. Aunque, debido a que, el objetivo principal del grupo de investigadores que desarrollaron MySQL fue la velocidad de respuesta, dejaron de lado algunas características importantes, como por ejemplo: no soporta transacciones ni "roll-backs", ignora la integridad referencial, dejándola en manos del programador, aspecto que se debe tomar en cuenta al momento de integrar la aplicación.

⁸ Fuente: <http://news.netcraft.com/> - June 2005 Web Server Survey.

Otra característica importante es que requiere pocos recursos, tanto de CPU como de memoria, aspecto importante ya que el servidor de base de datos destinado para la implementación del proyecto cumple con los requerimientos mínimos (64Mb. en memoria RAM, procesador P-III a 600Mhz, 10Gb. de espacio libre en disco duro).

Por otro lado, MySQL a partir de la versión 3.23.19. cuenta con licenciamiento GPL, la cual será utilizada para el proyecto, por lo que no se requiere invertir recursos en la adquisición de licencias.

Otro aspecto que se tomó en cuenta es la facilidad con que se integra con la tecnología para la generación de páginas Web dinámicas: PHP. Podemos decir que ambas tecnologías se han desarrollado conjuntamente.

Lenguaje de Programación: PHP

Una aplicación Web es "invocada" desde un navegador (por ejemplo Netscape o Internet Explorer), la aplicación consta de archivos html los cuales contienen scripts que definen las reglas de negocios para llevar a cabo transacciones y la lógica de procesos de la aplicación. De acuerdo a la arquitectura de tres capas, la capa de reglas de negocio recibe la solicitud de los clientes de Internet y genera una página html a partir de los scripts de lenguaje PHP, como se puede apreciar, esta capa proporciona el medio de comunicación entre el formato de la aplicación y la lógica de la aplicación.

El lenguaje PHP fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI escrito en lenguaje C que permitía la interpretación de un número limitado de comandos. El sistema fue denominado Personal Home Page Tools y tuvo éxito gracias a que terceras personas solicitaron a Rasmus que les permitiera utilizar sus programas en sus propias páginas. Dada la aceptación de la primera versión de PHP y de manera adicional, su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (Form Interpreter), el conjunto de estas dos herramientas sería la primera versión del lenguaje: PHP/FI.

En el último año, el número de servidores que utilizan PHP se ha disparado, logrando situarse cerca de los cinco millones de sitios Web y 800,000 direcciones IP, lo que ha convertido a PHP en una tecnología popular. Debido, entre otras razones, a que PHP es el complemento ideal para que el *tandem* Linux-Apache sea compatible con la programación del lado del servidor de sitios Web. Podemos decir que PHP ya comparte los éxitos augurados al conjunto de sistemas desarrollados en código abierto.

PHP nos permite insertar fragmentos de código dentro de una página HTML y realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas en un lenguaje distinto al HTML.

Principales funciones del Lenguaje PHP

Poco a poco el lenguaje PHP se ha convertido en un lenguaje completo y aceptado por un número considerado de desarrolladores de sitios Web. Sin lugar a duda su aceptación recae en las librerías de funciones, con las cuales se pueden realizar una multitud de tareas útiles para el desarrollo de sistemas en Web. A continuación se enuncian algunas de las funcionalidades más utilizadas:

Funciones de correo electrónico. Permite enviar un e-mail a un contacto o una lista, parametrizando toda una serie de metadatos tales como el e-mail de procedencia, asunto, persona a responder, etc.

Gestión de bases de datos. Resulta difícil concebir un sitio rico en contenido que no es gestionado por una base de datos. El lenguaje PHP ofrece interfaces para el acceso a la mayoría de las bases de datos comerciales y por ODBC a las bases de datos de Microsoft, a partir de las cuales se puede editar el contenido del sitio con absoluta sencillez.

Gestión de archivos. Crear, borrar, mover, modificar, etc., cualquier tipo de operación puede ser realizada a partir de una amplia librería de funciones para la gestión de archivos mediante PHP. También podemos transferir archivos por FTP a partir de sentencias en nuestro código, protocolo para el cual PHP ha

previsto también gran cantidad de funciones.

Muchas otras funciones pensadas **para Internet** como son el tratamiento de *cookies*, accesos restringidos, comercio electrónico, etc. Funciones de **propósito general** como: funciones matemáticas, explotación de cadenas, de fechas, corrección ortográfica, compresión de archivos, etc.

A manera de comparativo podemos decir que PHP y ASP son lenguajes parecidos en cuanto a potencia y dificultad, su sintaxis puede diferir sensiblemente.

PHP, aunque multiplataforma, ha sido concebido inicialmente para entornos UNIX y es en este sistema operativo donde se pueden aprovechar mejor sus prestaciones. ASP, siendo una tecnología Microsoft, está orientado hacia la plataforma Windows.

Las tareas fundamentales que puede realizar directamente el lenguaje son definidas en PHP como funciones, mientras que ASP invoca frecuentemente a objetos.

ASP realiza numerosas tareas sirviéndose de componentes (objetos) que deben ser programados y, subsecuentemente, comprados a empresas especializadas. PHP presenta una filosofía totalmente diferente, con un espíritu más generoso, es progresivamente construido por colaboradores desinteresados que implementan nuevas funciones en nuevas versiones del lenguaje, sin costo alguno.

PHP es un lenguaje de programación diseñado específicamente para aplicaciones Web; las características más destacables son su sintaxis fácilmente integrable dentro de código HTML, instalación en varios tipos de servidor Web, sintaxis similar a Perl y otros lenguajes de *scripting*.

Por las ventajas mencionadas se decidió integrar el sistema utilizando esta tecnología, como nos podemos dar cuenta, existe una coherencia muy marcada entre el Servidor Web Apache, el Sistema Manejador de Base de Datos MySQL y el lenguaje de

programación PHP, lo cual facilitara la integración de la aplicación a la hora de implementar el sistema.

CAPÍTULO III ANÁLISIS DEL SISTEMA SIRESE

Después de recabar información con los usuarios del sistema, el siguiente paso es definir los requerimientos del sistema.

Una vez que se han discutido las propiedades importantes de las que deberá constar el sistema, se tienen que generar modelos en donde se estructuren esquemáticamente los requerimientos definidos por el usuario.

El modelar un sistema, además de documentar los requerimientos y el análisis del sistema, también reafirma la comprensión del ambiente que se trata de sistematizar.

En esta sección del documento se presenta el diagrama de Casos de Uso, el modelo Entidad-Relación y el Diccionario de Datos del Sistema para el Registro de los Servicios proporcionados por la Unidad de Sistemas del Tribunal Electoral (SIRESE) que en este caso sirvió como caso práctico durante el desarrollo del diplomado.

3.1 CASOS DE USO

SISTEMA PARA EL REGISTRO DE LOS SERVICIOS PROPORCIONADOS POR LA UNIDAD DE SISTEMAS DEL TRIBUNAL ELECTORAL

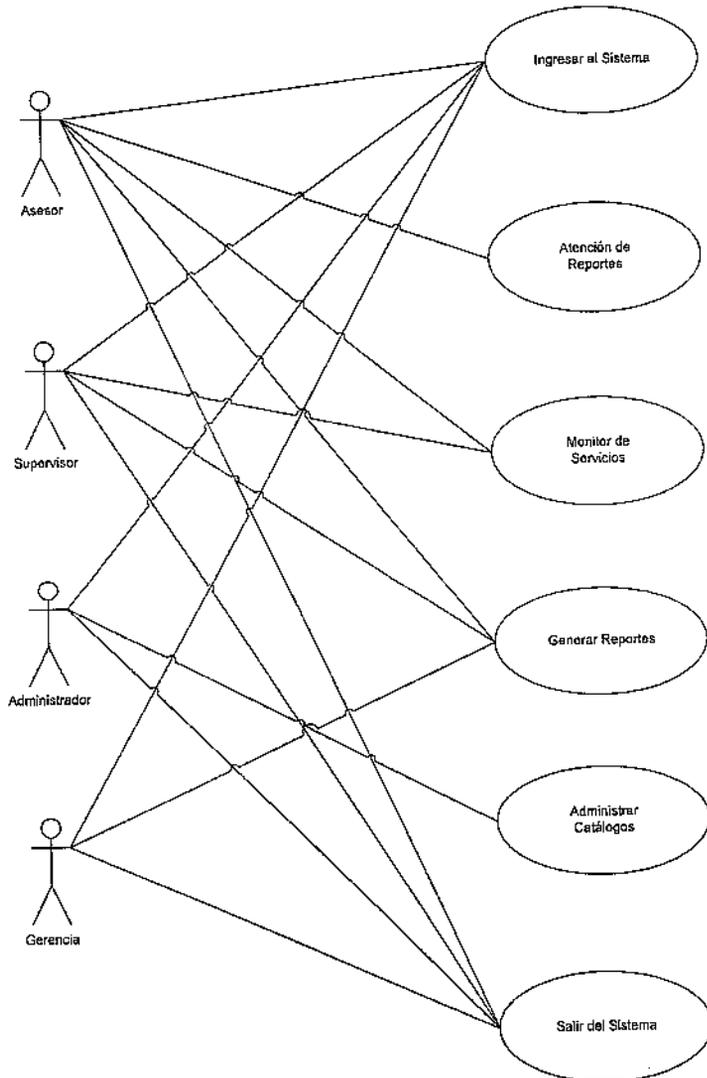


Figura 12. Casos de Uso del Sistema

Definición de Actores.

Actor	Descripción
Asesor	Es el personal encargado de atender o dar solución a las solicitudes de los usuarios. Dentro del sistema, deberá tener los permisos correspondientes para registrar servicios; así como: asignarlo a otro asesor, cerrar los servicios que tiene asignados, para lo cual deberá tener la facultad de monitorear los servicios abiertos y cerrados correspondientes a su grupo y en un primer plano los que tiene asignados y que están en espera de ser atendidos. Una vez que atiende o da solución a un servicio, regresará al sistema para recuperar el registro del mismo y documentar la información correspondiente a la solución y finalmente cerrar el servicio.
Supervisor	Son los coordinadores de grupo. Serán los encargados de monitorear el estatus de los servicios registrados en el sistema, así como dar seguimiento a la solución de las solicitudes de los usuarios. Dentro del sistema deberá tener la facultad de monitorear y asignar los servicios asignados al personal del grupo al que corresponde para distribuir la carga de trabajo. Una de sus funciones principales dentro del sistema será la de monitorear constantemente el registro de los servicios para detectar alguna eventualidad en el funcionamiento del equipo de la infraestructura informática que se refleje en la solicitud de servicios de parte de los usuarios, para dar aviso a la gerencia y llevar a cabo el plan de contingencia adecuado de acuerdo al origen y naturaleza del problema.
Administrador	El administrador actualiza la información de los catálogos, por ejemplo: dar de alta nuevos usuarios, asignar el rol correspondiente a cada usuario del sistema de acuerdo a las actividades que desempeña, etc.
Gerencia	Dentro del sistema, tendrá la factibilidad de ejecutar cualquier función: monitorear los servicios registrados, registrar servicios, eliminar servicios registrados, etc. Aunque cabe señalar que con frecuencia ingresara al sistema únicamente para generar reportes.

Definición y Descripción de Casos de Uso

Caso de Uso	Ingresar al Sistema
Actores	Asesor, Supervisor, Gerencia y Administrador.
Propósito	Autenticarse en el sistema para obtener los permisos correspondientes de acuerdo al perfil del usuario.
Descripción	Al ingresar, al sistema, se le solicita al usuario su nombre de usuario y contraseña para validar que es un usuario activo y que pertenece al sistema. Una vez ingresados los datos de autenticación se carga la pantalla principal con el menú correspondiente de acuerdo al perfil del usuario.
Referencias cruzadas	R1, R2, R3, R4, R5

Curso de Eventos			
Caso de Uso: Ingresar al Sistema			
Acción del Actor		Respuesta del sistema	
1.	Ingresar al sistema.	2.	Despliega la pantalla para que se autentifique al usuario, con un campo para nombre de usuario y otro para la contraseña.
3.	Teclea el nombre de usuario y contraseña, finalmente hace clic en el botón Aceptar.	4.	Valida la contraseña
		5.	Si los datos son validos despliega la pantalla principal y el menú correspondiente, los desplegados son diferentes dependiendo del grupo al que pertenezca el actor. Cuando los datos no corresponden se le solicitan nuevamente.

Caso de Uso	Atención de Reportes
Actores	Asesor, Supervisor y Administrador.
Propósito	Registrar, modificar, cerrar un reporte.
Descripción	Los actores involucrados para cumplir con sus actividades deberán de registrar los reportes solicitados por los usuarios a la Unidad de Sistemas. A su vez durante la atención de los reportes puede darse el caso de requerir modificar la información de los reportes ya registrados, siempre y cuando sean asignados al actor que se encuentra en línea. Para finalizar el ciclo de una solicitud de servicio registrada es necesario cerrarla, cabe señalar que una vez cerrada, solo el actor "Supervisor" del grupo correspondiente, podrá realizar modificaciones o dicho de otra forma reabrir el reporte.
Referencias cruzadas	R1, R2, R3, R4, R5

Curso de Eventos			
Caso de Uso: Atención de Reportes			
Acción del Actor		Respuesta del sistema	
1.	Del menú, selecciona, de acuerdo al proceso que desee llevar a cabo: registrar, modificar o cerrar un reporte.	2.	En caso de seleccionar la opción para registrar un reporte, se despliega el formulario correspondiente con los campos vacíos para documentar los datos del nuevo reporte. En otro caso (modificar o cerrar) se despliega la lista de los reportes registrados, en primera instancia se muestran los reportes abiertos asignados al actor dueño de la sesión, esta pantalla tiene opciones para que el usuario seleccione ver también los reportes cerrados, incluso si tiene los permisos correspondientes, podrá ver los reportes de actores del mismo grupo o incluso de grupos diferentes.
3.	El usuario selecciona las opciones de la pantalla para visualizar los reportes de su interés, para seleccionar alguno.	4.	Se modifica la lista de los reportes registrados de acuerdo a los criterios seleccionados por el usuario.
5.	El actor selecciona el reporte deseado.	6.	La ventana con la lista y los criterios de selección se cierra.

7.		8.	Se despliegan los datos del reporte seleccionado por el usuario. Podrá modificar información siempre y cuando el reporte este asignado a el y no este cerrado, para lo cual las opciones de guardar cambios, asignar y cerrar estarán activadas, en caso contrario se mostraran inactivas.
9.	El usuario realizara los cambios en la información o llevara a cabo alguna acción con el reporte (asignar, cerrar o eliminar).	10.	El sistema registra la opción seleccionada por el usuario.
		11.	Se valida la ejecución de la acción seleccionada, de acuerdo al status actual del registro y a los permisos del actor frente al registro seleccionado.
		12.	Si la acción procede se actualizan los datos en la base. En caso contrario se envía un mensaje acerca del error o el por que no se pudo llevar a cabo la acción solicitada.
		13.	En la pantalla se desactivaran los controles que ya no tengan que estar disponibles de acuerdo al nuevo status del reporte.

Caso de Uso	Monitor de Servicios
Actores	Asesor, Supervisor, Gerencia y Administrador.
Propósito	Saber el estado actual de los servicios registrados.
Descripción	Los actores involucrados, sin lugar a duda por el objetivo del sistema, estarán interesados en conocer los servicios pendientes por atender en ese momento o tener la posibilidad de visualizar un reporte registrado ya sea que continúe abierto o en espera de ser atendido o que ya se encuentre cerrado lo que indicaría que fue atendido. Se desplegaran en primer momento los servicios abiertos asignados al asesor que este en línea, ahí mismo tendrá la posibilidad de seleccionar cuales registros quiere ver: los servicios cerrados, ambos (abiertos y cerrados), los correspondientes a su grupo (abiertos y/o cerrados), y si tiene permiso los de otros grupos como es el caso del actor Gerencia o Administrador.
Referencias cruzadas	R6, R7, R8, R9, R10, R11, R12, R13, R14

Curso de Eventos			
Caso de Uso: Monitor de Servicios			
	Acción del Actor		Respuesta del sistema
1.	Seleccionar del menú la opción: Monitor de Servicios.	2.	Se desplegara la lista de los servicios pendientes por atender asignados al actor dueño de la sesión, así como opciones para desplegar los servicios cerrados o que fueron atendidos por el mismo (histórico). También se mostraran opciones para desplegar en la lista, los servicios asignados a otro actor miembro de su grupo o incluso los pertenecientes a otro grupo siempre y cuando se tengan los permisos correspondientes.
3.	El usuario selecciona las opciones deseadas, para cambiar los criterios de la lista de servicios, desplegada en ese momento.	4.	Actualiza la lista de los servicios de acuerdo a los criterios seleccionados por el usuario, siempre y cuando el perfil del actor en curso tenga los permisos correspondientes, de lo contrario la lista aparecerá vacía.
5.	El usuario selecciona un servicio específico.	6.	Se despliega la información completa del servicio seleccionado.

Caso de Uso	Generar Reportes
Actores	Supervisor de Grupo, Gerencia o Administrador.
Propósito	Generar reportes de los servicios registrados.
Descripción	Una vez seleccionado el reporte a generar de los disponibles, se deberá proporcionar la información solicitada por el formulario correspondiente, puede ser el nombre de un asesor, grupo, un rango de fechas, etc. Finalmente se generara el reporte, de acuerdo a los datos proporcionados, desde donde se puede realizar una impresión del mismo.
Referencias cruzadas	R19

Curso de Eventos			
Caso de Uso: Generar Reportes			
	Acción del Actor		Respuesta del sistema
1.	El actor selecciona del menú la opción para "Generar Reportes"	2.	Se muestra la lista de los reportes disponibles.
3.	El actor deberá seleccionar el reporte que desea generar.	4.	Se despliega un formulario solicitando los datos que se requieren para generar el reporte seleccionado.
5.	El actor llenará el formulario con los rangos de datos de acuerdo a los criterios que seleccione para generar el reporte. Cuando estén completos deberá hacer clic en el botón con la leyenda <i>Aceptar</i> .	6.	El sistema validara si los datos están completos y si son validos de acuerdo a la información que se solicita.
		7.	Si la validación es aprobada, se genera el reporte, de lo contrario se le notifica al usuario para que documente la información solicitada nuevamente.
8.	El usuario visualiza el reporte generado. Puede seleccionar la opción para realizar una impresión.	9.	El sistema muestra la ventana con las opciones de impresión.
10.	El usuario selecciona las opciones de impresión.	11.	Se envía la impresión a la impresora seleccionada.

Caso de Uso	Administrar Catálogos
Actores	Administrador
Propósito	Mantener la información actualizada de los catálogos: usuarios y su adscripción (área), asesores y su grupo, y tipos de servicio.
Descripción	Modificar, agregar o eliminar, datos de los catálogos de acuerdo a las modificaciones solicitadas. Para lo cual, deberá seleccionar del menú el catalogo, generar un registro nuevo o seleccionar el registro correspondiente según sea el caso, hacer el cambio en el campo o campos; finalmente guardar los cambios realizados. Repetir la operación para modificar datos de otros registros.
Referencias cruzadas	R20, R21, R22

Curso de Eventos			
Caso de Uso: Administrar Catálogos			
	Acción del Actor		Respuesta del sistema
1.	El actor selecciona la opción "Administrar Catálogos" del menú.	2.	Se muestran los catálogos del sistema.
3.	El actor selecciona el catálogo al que desea ingresar.	4.	El sistema muestra un formulario con los campos vacíos correspondientes a un nuevo registro del catalogo seleccionado.
5.	Si el usuario requiere agregar un nuevo registro, llena los campos correspondientes y finalmente selecciona la opción para guardar los cambios realizados. Si desea realizar otro procedimiento como: modificar o eliminar un registro primero deberá seleccionar el registro deseado.	6.	El sistema mostrará una lista de los registros existentes en el catalogo, y las opciones para seleccionar uno de ellos en específico mediante algunos criterios.
7.	El usuario seleccionará el registro.	8.	Se mostrará un formulario con los datos completos del registro seleccionado, donde el actor podrá realizar cambios en la información o eliminar el registro.
9.	El usuario realizará los cambios necesarios o la opción para borrar el registro. Confirmando	10.	El sistema validará de acuerdo al estatus actual del registro y a al tipo de datos que contenga el

	<p>que requiere ejecutar la operación seleccionada.</p>	<p>formulario en todos sus campos. Si se acepta la validación, el procedimiento seleccionado por el usuario se llevara a cabo. De lo contrario se enviara un aviso al usuario, indicando el por que no se llevo a cabo la operación seleccionada.</p>
--	---	---

REFERENCIAS CRUZADAS

No.	Función
R1.	Registrar datos requeridos de un reporte de servicio
R2.	Asignar servicio
R3.	Registrar solución de un reporte de servicio
R4.	Cerrar solicitud de un reporte de servicio
R5.	Imprimir comprobante de una solicitud de servicio
R6.	Consultar reportes asignados pendientes por resolver
R7.	Consultar histórico de reportes por grupo
R8.	Consultar histórico de reportes por asesor
R9.	Consultar servicios pendientes por atender
R10.	Consultar histórico de servicios proporcionados a un usuario en específico
R11.	Consultar servicios pendientes por atender por asesor
R12.	Consultar servicios pendientes por atender por grupo
R13.	Consultar el detallado de un reporte abierto
R14.	Consultar el detallado de un reporte cerrado
R15.	Registrar alguna actividad a llevar a cabo en algún servidor
R16.	Modificar información de la actividad que se esta llevando a cabo en un servidor
R17.	Cerrar el registro de la información de alguna actividad llevada cabo en un servidor para registrar la fecha y hora de termino de la actividad
R18.	Imprimir la información de alguna actividad realizada en algún servidor
R19.	Generar un reporte
R20.	Añadir registros a un catalogo determinado
R21.	Modificar registros de un catalogo
R22.	Eliminar registros de un catalogo

3. 2 DIAGRAMA ENTIDAD-RELACIÓN

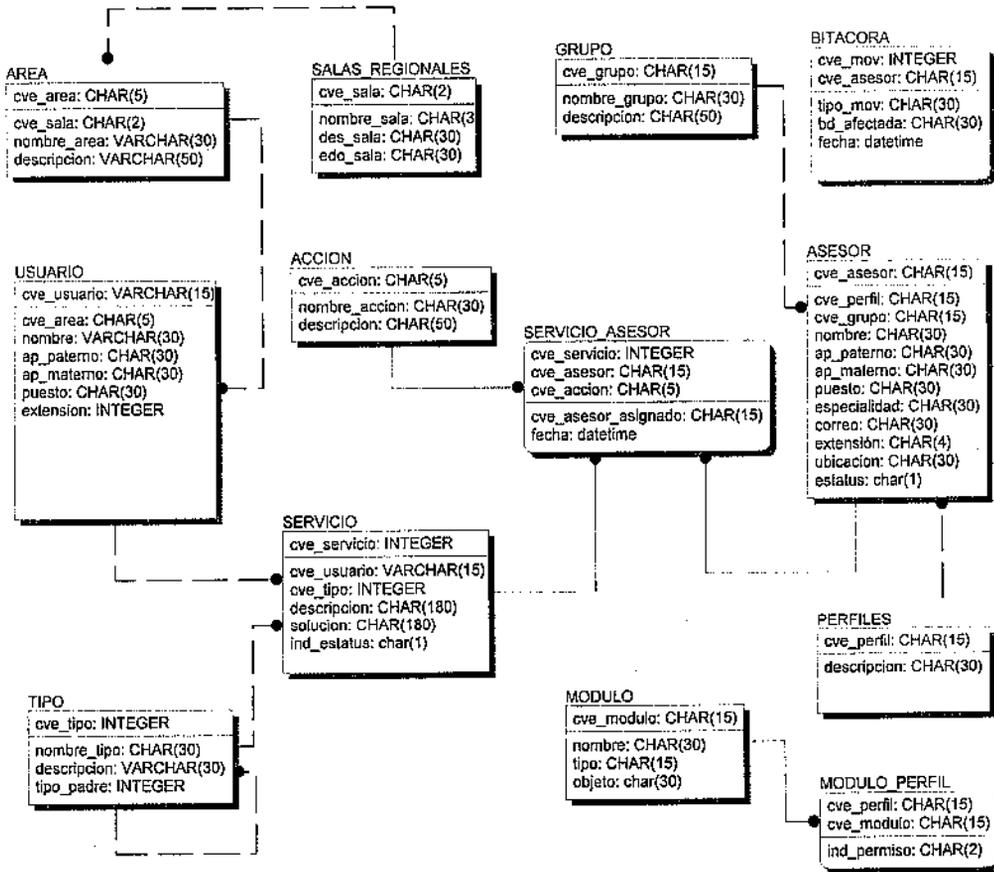


Figura 13. Diagrama Entidad-Relación.

3.3 DICCIONARIO DE DATOS

Entidad: USUARIO

Descripción: Almacena los datos personales de los usuarios del Tribunal Electoral. Los cuales solicitan servicios a la Unidad de Sistemas.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_usuario	varchar (15)	La clave de usuario será el identificador único para los usuarios. Está compuesto por la primera letra de su nombre y el apellido paterno.	PK
cve_area	numerico	Es una llave foránea, relaciona a esta entidad con la entidad AREA, mediante una relación muchos a uno.	FK
cve_sala	nimerico	Es una llave foránea, relaciona a esta entidad con la entidad SALASREGIONALES, mediante una relación uno a muchos.	FK
nombre	varchar (30)	Nombre del usuario.	
ap_paterno	varchar (30)	Apellido paterno del usuario.	
ap_materno	varchar (30)	Apellido materno del usuario.	
puesto	varchar (50)	Puesto que ocupa en la institución.	
extension	numerico	El número de la extensión telefónica a donde se le puede localizar dentro de la institución. (Se compone de cuatro dígitos)	

Entidad: AREA

Descripción: Será el catálogo de las área existentes en el Tribunal Electoral a las cuales pueden pertenecer los usuarios, cada usuario solo podrá estar asignado a una y solo una área. A su vez cada área estará vinculada a solo una sala regional. Estos datos se importaran del catalogo de áreas de Recursos Humanos.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_area	numérico	La clave de área será el identificador único para las áreas a las que pueden pertenecer los usuarios.	PK
cve_sala	numerico	Es una llave foránea, relaciona a esta entidad con la de SALAS REGIONALES, es la clave de la Sala Regional a la cual pertenece el área.	FK
nombre_area	carácter (30)	Es una llave foránea, relaciona a esta entidad con la entidad AREA, mediante una relación muchos a uno.	
descripción	carácter (60)	Breve descripción acerca del área, por ejemplo: ubicación, información más detallada, etc.	

Entidad: SALA_REGIONAL

Descripción: Será el catálogo de las Salas Regionales. Estos datos se importaran del catalogo de Salas Regionales de Recursos Humanos.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_sala	numérico	La clave de área será el identificador único para las áreas a las que pueden pertenecer los usuarios.	PK
nombre_sala	carácter (30)	Nombre descriptivo de la Sala Regional.	
des_sala	carácter (60)	Información particular de cada sala, por ejemplo: estados que componen la circunscripción, dirección, etc.	
edo_sala	carácter (30)	Nombre del estado donde se encuentra ubicada la Sala Regional.	

Entidad: SERVICIO

Descripción: En esta tabla se almacenara la información correspondiente a los servicios proporcionados por la Unidad de Sistemas, una vez que se registren los mismos.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_servicio	numérico	Será el identificador único para cada servicio registrado. Este número será generado automáticamente por el sistema.	PK
cve_usuario	carácter (15)	Es una llave foránea, relaciona a esta entidad con la entidad USUARIO, describe al usuario a quien se le proporciona el servicio.	FK
cve_tipo	numérico	Es una llave foránea, relaciona a esta entidad con la entidad TIPO, esta clave representa al tipo de servicio que se esta proporcionando.	FK
descripción	texto	Es la descripción del problema o servicio solicitado.	
solucion	texto	La solución al problema reportado, o la acción tomada en atención al servicio solicitado.	

Entidad: TIPO

Descripción: Es el catalogo que define los tipos de servicio, es decir la clasificación de los servicios proporcionados por la Unidad de Sistemas. La estructura de los servicios es jerárquica con lo que tendremos varios niveles para clasificarlos, por ejemplo: La clasificación Hardware es de la jerarquía mas alta y el siguiente nivel estará compuesto por los elementos CPU, monitor, teclado, mouse, impresoras, etc. A su vez la clasificación impresoras, tiene en un siguiente nivel con los siguientes elementos: Mantenimiento Preventivo, Atasco de Papel, Verificar Conexiones, Configuración.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_tipo	caracter (10)	Se compone de 8 caracteres numéricos, los cuales están agrupados en pares, y entre más a la izquierda se encuentren tienen más alta jerarquía. Para mayor referencia ver la tabla: "Clasificación de los tipos de Servicio localizada en el Anexo A"	PK
nombre_tipo	caracter (30)	Nombre del tipo de servicio.	
des_tipo	caracter (60)	Información descriptiva acerca del tipo de servicio.	
tipo_padre	caracter (10)	Representa su posición jerárquica, lo relaciona directamente con un tipo de servicio de jerarquía más alta lo que lo hace pertenecer a esa jerarquía. Si este valor es negativo significa que este tipo de servicio es del primer nivel en la clasificación de los servicios.	

Entidad: ASESOR

Descripción: Es el catalogo que contiene la información del personal de sistemas, encargados de registrar y atender los requerimientos solicitados por los usuarios.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_asesor	caracter (15)	Clave del Asesor. Se compone de la primera letra del nombre y apellido paterno.	PK
cve_grupo	caracter (15)	La clave del grupo es una llave foránea, para relacionar la entidad GRUPO con esta entidad. Para mayor referencia ver la información de la entidad GRUPO.	FK
nom_ase	caracter (30)	Nombre del asesor.	
ap_pat_ase	caracter (30)	Apellido paterno del asesor.	
ap_mat_ase	caracter (30)	Apellido materno del asesor.	
puesto_ase	caracter (40)	Puesto del asesor.	
esp_ase	texto	Especialidad. Se describirá la	

		especialidad del asesor de acuerdo a sus conocimientos.	
correo_ase	caracter (30)	Dirección de correo electrónico del asesor.	
ext_ase	numerico	Número de extensión telefónico del asesor.	
ubica_ase	caracter (30)	Se describe la ubicación física del asesor, por ejemplo: Centro de Cómputo, Soporte Técnico, y la Sala Regional a la que pertenece.	

Entidad: GRUPO

Descripción: Esta entidad define el catalogo de los grupos a los que puede pertenecer el personal de sistemas, los cuales se describen a continuación.

Nombre del Grupo	Clave	Descripción
Mantenimiento	M	El personal de este departamento, es encargado de llevar a cabo el mantenimiento preventivo y correctivo del equipo de cómputo así como la instalación del mismo.
Centro de Computo	C	El personal de este grupo lleva a cabo la administración de los servidores.
Soporte Técnico	S	Los integrantes de este departamento son los encargados de atender en sitio a los usuarios, así como configurar los equipos de cómputo que utilizan los usuarios.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_grupo	carácter (1)	Clave del grupo. Consta de un carácter. cve_grupo = [M C S] M = Mantenimiento. C = Centro de cómputo. S = Soporte Técnico.	PK
nom_grupo	carácter (30)	Nombre del grupo.	
des_grupo	carácter (60)	Descripción del grupo.	

Entidad: SERVICIO-ASESOR

Descripción: Esta entidad fue derivada de la relación del tipo muchos a muchos de las entidades: SERVICIO y ASESOR. Ya que cada servicio proporcionado podrá estar relacionado con uno o más asesores dependiendo de quién lleve a cabo las acciones durante el proceso de un servicio. En esta tabla se registran las acciones llevadas a cabo durante el proceso de la solicitud de un servicio especificando que asesor lleve a cabo la acción correspondiente junto con la fecha y hora. Por lo que de esta tabla dependerá el estatus del servicio, para mayor referencia ver el detalle de la entidad ACCION.

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_servicio	numérico	Clave del servicio.	FK*
cve_asesor	carácter (15)	Se refiere a la clave del asesor.	FK*
cve_accion	carácter (2)	Clave de la acción, que llevara a cabo el asesor al cual pertenece la clave con respecto al servicio especificado en la clave del servicio.	FK*
cve_asesor	carácter (15)	Este campo se llenara cuando el asesor que lleve a cabo la acción "Asignar", y representa la clave del asesor que se encargara de atender el servicio. Por lo tanto cuando el asesor, a quien se asigno el servicio, consulte los servicios pendientes por atender, este servicio aparecerá en la lista.	FK
fecha	fecha	Se refiere a la fecha y hora en la que se llevar a cabo la acción indicada. La información se registra con el siguiente formato: (dd/mm/aaaa) (hh:mm)	

Entidad: ACCION

Descripción: Se refiere al catalogo de las diferentes acciones que un asesor puede tener con respecto a la atención de un servicio en sus diferentes etapas, las cuales se describen a continuación.

ACCION		
Valor	Descripción	Estatus
Abrir (O)	La acción abrir corresponde a la acción de registrar una nueva solicitud de servicio.	Abierto
Asignar (D)	Se refiere a la acción de asignar el servicio al ingeniero de soporte encargado de atender el servicio al que se hace referencia. Nota: Al llevar a cabo esta acción se deberá documentar la clave del asesor a quien se asigna el servicio.	Abierto
Cerrar (C)	El asesor que atendió el servicio, será el encargado de cerrar el servicio, para lo cual deberá documentar las actividades realizadas en el atributo <i>solución</i> de la entidad <i>SERVICIO</i> . Cuando se lleva a cabo esta acción se da por entendido que el servicio se atendió y el usuario está conforme con las acciones llevadas a cabo.	Cerrado

Atributos:

Atributo	Tipo de Dato	Descripción	
cve_accion	caracter (1)	Clave de la Acción. Consta de un carácter. cve_accion = { O D C } O = Abrir o registrar servicio. D = Asignar servicio. C = Cerrar servicio.	
nom_accion	caracter (15)	Nombre de la acción.	
des_accion	caracter (60)	Descripción de la acción.	

PARTE IV DISEÑO E IMPLEMENTACIÓN DEL SISTEMA SIRESE

4.1 GENERALIDADES DEL SISTEMA PARA LA IMPLEMENTACIÓN

El Sistema para el Registro de los Servicios Proporcionados por la Unidad de Sistemas del Tribunal Electoral (SIRESE) es la propuesta de solución al planteamiento del problema expuesto en el primer Capítulo de este documento. A continuación se muestra el esquema general de la red para la implementación del sistema.

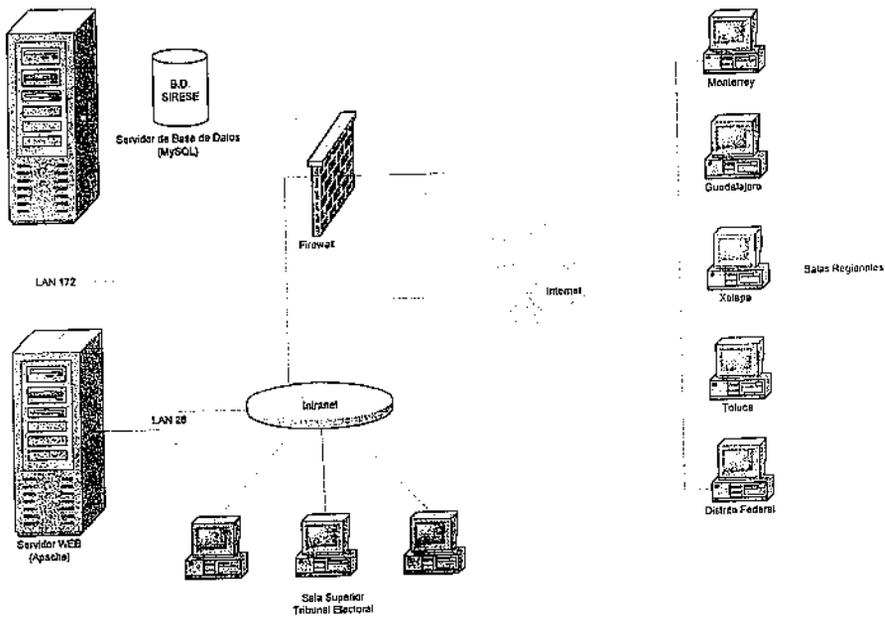


Figura 14. Esquema general de red para la implementación del SIRESE.

4.2 PROCEDIMIENTO PARA EL REGISTRO DE LOS SERVICIOS

El siguiente esquema muestra el procedimiento para el registro de los servicios, que será implementado en el sistema para tal fin.

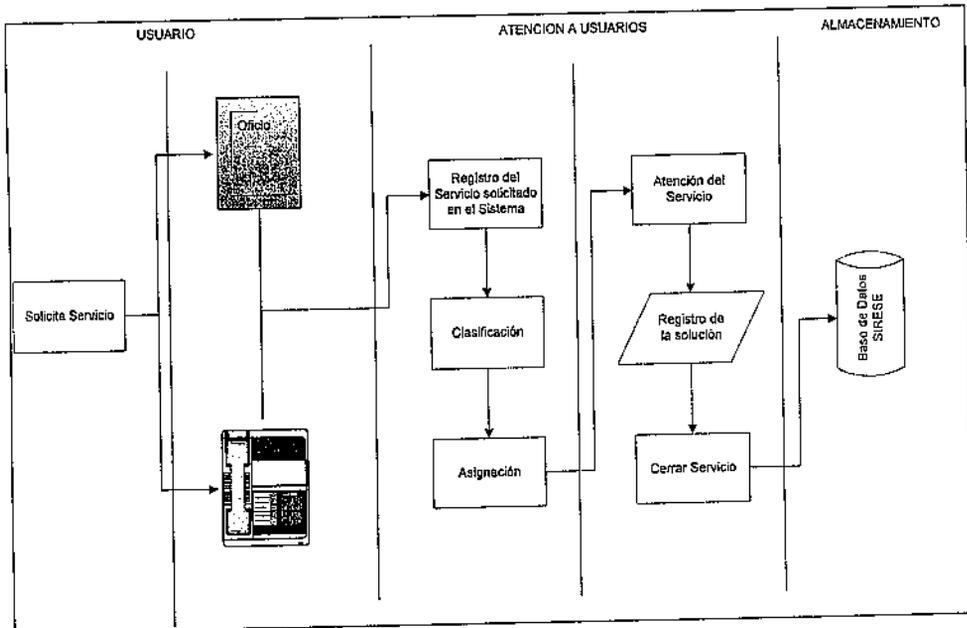


Figura 15. Diagrama de flujo del proceso de registro y atención de servicios.

A continuación se describe el procedimiento para el registro de los servicios.

1.- El usuario solicita un servicio, mediante una llamada telefónica, por escrito o directamente en el departamento de Atención a Usuarios.

2.- El servicio solicitado es registrado, clasificado y asignado por el personal de Atención a Usuarios encargado de recibir las solicitudes de parte de los usuarios.

El registrar un servicio consiste en documentar los datos del usuario (nombre y área a la que pertenece); la fecha y hora de recepción de la solicitud. Asignando un número de servicio.

De acuerdo a los requerimientos del usuario y a la naturaleza del problema, se clasifica el servicio de acuerdo a la definición de los *tipos de servicio*⁹. En este momento el personal encargado de registrar los servicios deberá documentar la descripción del servicio solicitado por el usuario.

Por último, como parte del registro del servicio, y de acuerdo a la clasificación del problema, el servicio se asignará a un Ingeniero de Soporte del departamento de Atención a Usuarios o se turnara al departamento correspondiente, ya sea Centro de Computo, Telefonía o Desarrollo de Sistemas.

3.- Desde el Monitor de Servicios el personal de Atención a Usuarios y de los demás departamentos, tienen conocimiento acerca de los servicios que tienen asignados. Desde donde pueden conocer el detallado de cada uno de ellos.

4.- Una vez que se resolvió el problema presentado por el usuario, se accesa desde el monitor de servicios al detallado del servicio, para documentar la solución del mismo y cerrar el servicio.

Si se requiere, una vez que el servicio esta cerrado, es posible realizar una impresión con la información del servicio para recabar la firma de conformidad del usuario.

Descripción del Estatus de los Servicios Registrados	
Abierto	El servicio esta en espera de ser atendido por el personal a quien haya sido asignado.
Cerrado	El servicio ya fue atendido. Deberá contener la descripción de la solución del problema, así como la fecha y hora del momento en el que el asesor a quien fue asignado, cerro el servicio.

Tabla 7. Descripción del estado de los servicios registrados.

⁹ Para mayor referencia consulte el apartado "*Clasificación de los Servicios*" de la sección 4.4 "*Reglas de Negocio*" de este Capítulo.

4.3 ESTRUCTURA Y DESCRIPCIÓN DE LOS MÓDULOS DEL SIRESE

De acuerdo al análisis llevado a cabo, acerca de los requerimientos del sistema, se plantea la siguiente estructura de los módulos y procesos de los que comprenderá el sistema. Posteriormente se presenta una descripción detallada de los mismos y una explicación detallada de aquellos procesos de importancia considerable para el funcionamiento integral del sistema.

DIAGRAMA DE PROCESOS

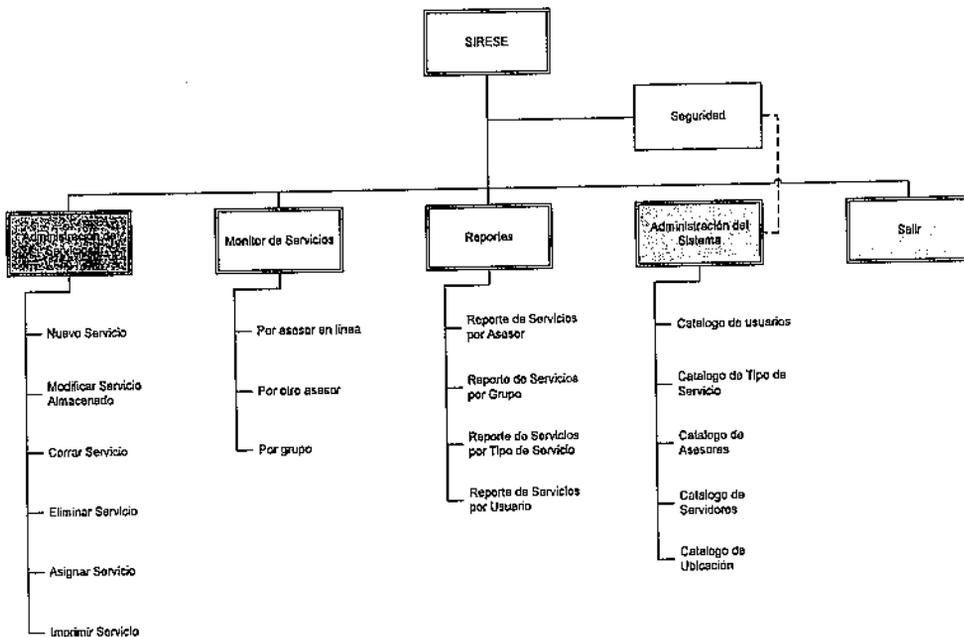


Figura 16. Diagrama de Procesos del SIRESE.

Módulo de Seguridad.

Este módulo será el encargado de permitir o denegar el acceso a los usuarios registrados en el sistema, mediante la solicitud de un nombre de usuario y contraseña, mismos que son almacenados en la base de datos "Seguridad".

Este módulo se compone de una pantalla para solicitar los datos del usuario, la cual se mostrara siempre que todo usuario intente acceder al sistema. Ver figura 17.

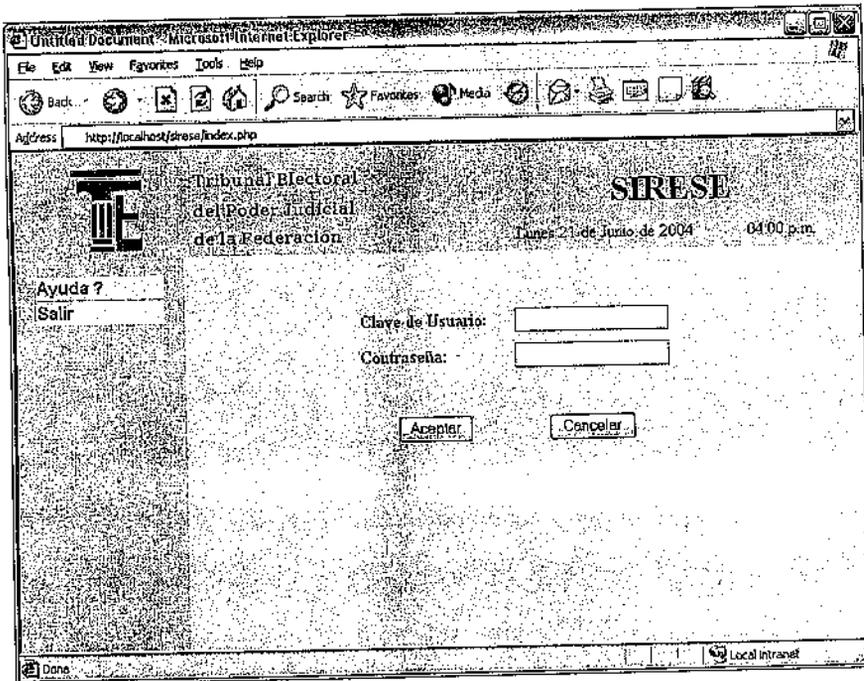


Figura 17. Pantalla de Inicio de Sesión, SIRESE.

Además, de una pantalla para la administración de usuarios, es decir dar de alta usuarios nuevos, modificar datos del usuario incluyendo la contraseña de acceso, así como en determinado momento bloquear el acceso a los usuarios, desactivando la cuenta del usuario. Para mayor referencia ver la descripción del "Módulo de Catálogos" que se presenta posteriormente.

Por otro lado y quizá lo más importante del módulo. Una vez que el usuario se autentifica o ingresa en el sistema satisfactoriamente, es decir es un usuario autorizado para ingresar al sistema, se despliega la pantalla principal (ver Fig.19), donde podemos apreciar del lado derecho de la pantalla, el menú con las opciones a las que tiene acceso el usuario que ingresó al sistema. Cada usuario del sistema estará asignado a

un *Grupo* así como a un *Rol*¹⁰. Las opciones de menú que se muestren se generan dinámicamente de acuerdo al rol del usuario que se registro. Si el perfil del usuario determina que no deberá tener acceso a la información de otros grupos, se restringirá, en la pantalla del módulo "Monitor de Servicios".

Los permisos asignados a cada usuario se almacenan en las tablas correspondientes a la base de datos seguridad. Debido a que la seguridad es algo muy importante para el sistema es conveniente analizar la forma en la que se registra la información que dará los privilegios de acceso a los usuarios, ya que estos determinan las opciones disponibles de forma individual para que interactúen con el sistema.¹¹

Módulo de Catálogos.

En este módulo se administrara la información de los catálogos que contienen la información sustancial del sistema. Es decir permite actualizar, añadir y eliminar la información de los catálogos.

Es conveniente señalar que el proceso para eliminar información de un catalogo, no borrar el registro de la base de datos, para no descuidar la integridad de los mismos, ya que probablemente la información se relacione con registros de otras tablas. Al consultar un catalogo se puede observar que no se visualiza la información de los elementos eliminados, de los cuales aun existe el registro pero tiene un status de inactivo o eliminado. Por ejemplo al eliminar un asesor, ya no aparece en pantalla, pero como existe un histórico de los servicios que atendió y requieren hacer referencia a sus datos como su nombre o clave de asesor el registro no se elimina físicamente, aunque tiene el estatus de inactivo, finalidad del campo Estatus de la tabla *ASESOR* de la base de datos del Sistema y por lo tanto no se muestra en el listado de asesores de la pantalla del "*Catálogo de Asesores*"¹².

¹⁰ y ¹¹ Para mayor referencia ver el apartado "*Grupos y Roles*" de la sección 4.4 "*Reglas de Negocio*" de este Capítulo.

¹² Para más información acerca de la definición de los catálogos ver como están definidos en el Diccionario de Datos, en la sección 3.3 "*Análisis del Sistema*" del Capítulo III de este documento.

Módulo Administración de Servicios

Registro de Servicios

Será utilizado por los operadores encargados de recibir las solicitudes de servicios de los usuarios del equipo de cómputo del Tribunal, para dar de alta o registrar un nuevo servicio, se deberá documentar la siguiente información:

- 1.- Seleccionar usuario.
- 2.- Clasificar el servicio.
- 3.- Descripción del problema.
- 4.- Asignar a un asesor.
- 5.- Guardar cambios.

Tabla 8. Información requerida para el registro de una solicitud de servicio.

Cerrar Servicio

Al cerrar un servicio, se deberá documentar la solución al problema. Una vez cerrado el servicio, se podrá cambiar su estado a abierto, únicamente por el supervisor de grupo, para efectuar alguna modificación en la información. El asesor que tenga asignado el servicio deberá nuevamente retraer el servicio para cerrarlo. Para mayor referencia acerca del "*Ciclo de Vida de un Servicio*" haga referencia a la sección "Procedimiento para el registro de los Servicios" al inicio de este Capítulo.

Consultar y Modificar un Servicio

Una vez registrado un servicio y a su vez asignado a un asesor, es posible consultarlo y modificarlo ya que se puede documentar parcialmente la solución al mismo.

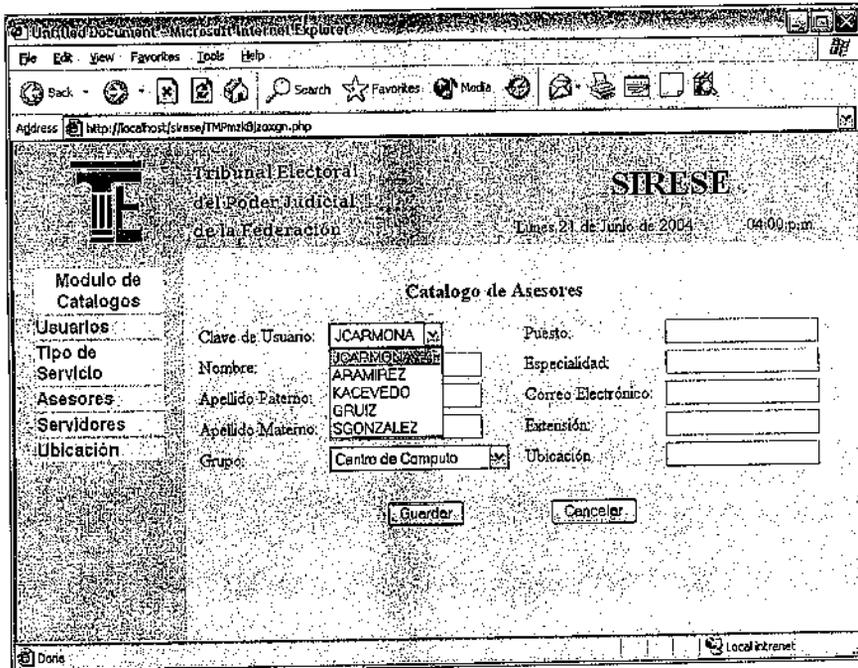


Figura 18. Pantalla del Catalogo de Asesores del Módulo Administración de Catálogos.

Módulo Monitor de Servicios

Este módulo será utilizado por los asesores, supervisores de grupo y operadores, para visualizar los servicios que se han registrado en el sistema. Los servicios se muestran clasificados, dado que lo más importante, en cuanto a funcionalidad se refiere, es saber si los servicios han sido atendidos o si hay muchos en espera de ser atendidos, lo cual indicara que probablemente está ocurriendo una contingencia.

Este módulo permite visualizar los servicios registrados, de acuerdo al estatus que presente "Abierto" o "Cerrado", con lo cual los Ingenieros de Soporte, conocen, en primera instancia, los servicios que tienen pendientes por atender.

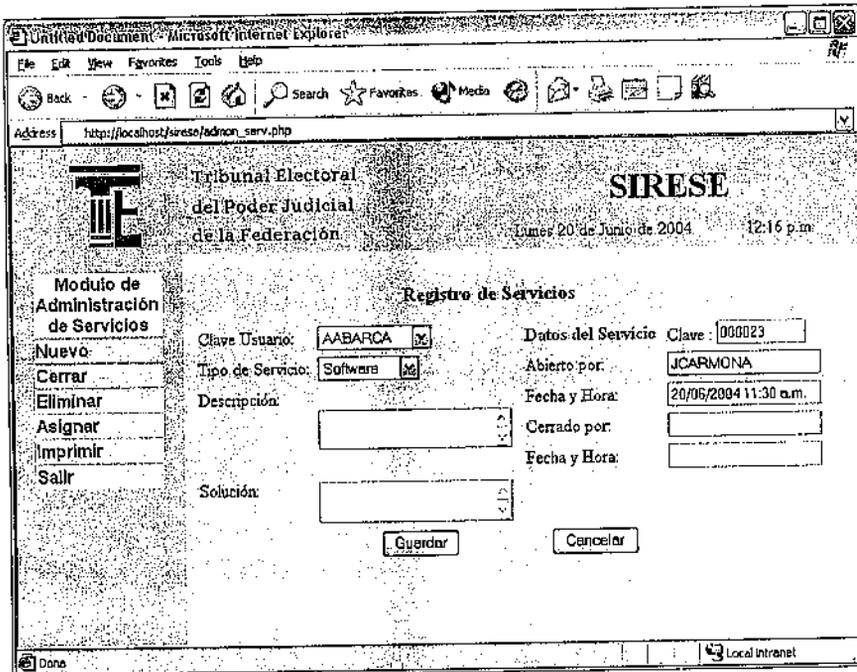


Figura 19. Pantalla para el Registro de los Servicios.

A los operadores del sistema, les permitirá, conocer el número de servicios abiertos por asesor, con la finalidad de distribuir el trabajo de acuerdo al número de servicios por atender de cada asesor. Los supervisores tendrán un panorama acerca de los servicios pendientes por atender lo cual les puede dar la oportunidad de definir prioridades, así como monitorear en tiempo real el estatus de los servicios y conocer el número de servicios abiertos, así como el detallado de alguno de los servicios cerrados para conocer los términos en los que se dio solución al problema, así como el tiempo de respuesta.

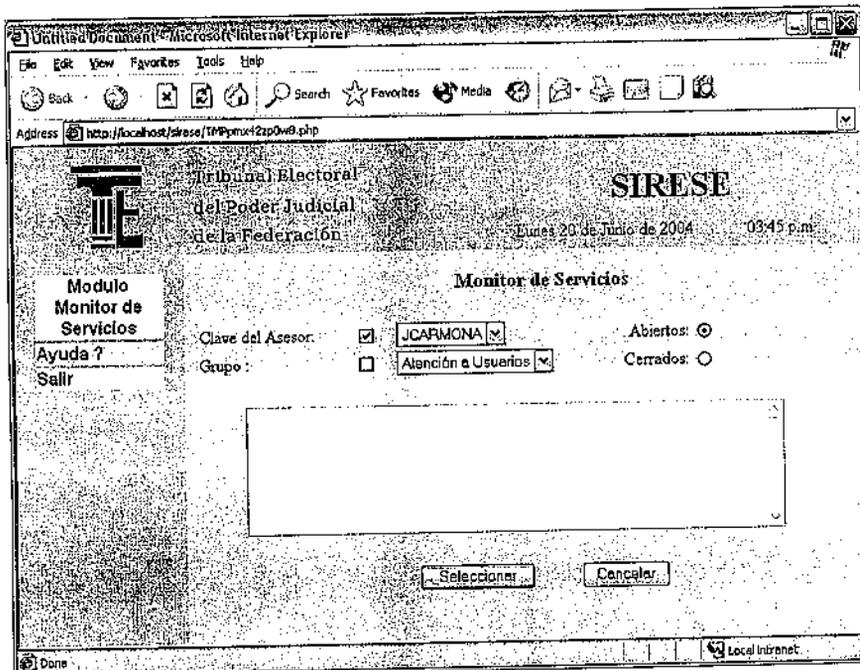


Figura 20. Pantalla del Módulo Monitor de Servicios.

Módulo de Reportes

Este módulo nos permite acceder a los reportes del sistema, los cuales son generados mediante código PHP. Los reportes son un producto muy importante del sistema, por lo general tiene más impacto en los niveles superiores de la jerarquía del medio donde se implementa un sistema ya que todo reporte sirve como fundamento para la toma de decisiones. Se debe ser muy cuidadoso al construir los reportes, en primera para satisfacer los requerimientos del usuario y en segunda para que los datos que se presenten sean los correctos sobre todo cuando se hacen cálculos con datos numéricos o de tipo fecha.

4.4 REGLAS DE NEGOCIO

Grupos y Roles

Grupos.

Los usuarios del sistema, estarán agrupados siguiendo la estructura de los departamentos de la Unidad de Sistemas, con el objetivo de que la información se clasifique de tal forma que sea posible filtrar la información a la hora de generar reportes, o durante la operación diaria, únicamente, y siempre y cuando así se indique, se tenga acceso a la información del departamento o grupo, al que el usuario que ingreso al sistema, pertenezca.

Para lo cual resulta importante ver la forma en la que estas restricciones se pueden lograr, para lo cual haremos referencia a las entidades GRUPO y ASESOR del diagrama Entidad-Relación.

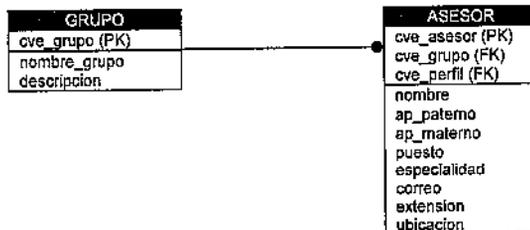


Figura 21. Extracto del diagrama E-R. Se presentan las entidades GRUPO y ASESOR y la relación existente entre ambas.

Lo que se quiere lograr es, además de tener a los usuarios del sistema agrupados de acuerdo al departamento al que pertenecen, mantener la información que se almacena en el sistema también agrupada de esta forma (para lograr que la información de cada grupo sea confidencial e independiente para cuestiones de generar reportes) se asocia a cada usuario del sistema, que en este caso se registran en la entidad ASESOR, un grupo (o departamento) mediante la relación uno a muchos que relaciona ambas entidades. De tal forma que desde el módulo "Monitor de Servicios" donde se muestran los servicios registrados ya sea por el asesor que se registro en el sistema, esto de

forma predeterminada, o por el grupo al que pertenece, pero sin poder visualizar la información de los servicios registrados por el personal de otro departamento.

A continuación se muestra la sentencia SQL para construir el *recordset* que contendrá los servicios abiertos del grupo al que pertenece el usuario firmado en ese momento en el sistema. Esta sentencia será utilizada por el módulo "Monitor de Servicios".

```
Select cve_servicio, cve_asesor, cve_usuario, cve_tipo,
descripcion from servicio_asesor
inner join servicio
on servicio.cve_servicio =
servicio_asesor.cve_servicio and servicio.estatus = 'o'
inner join asesor
on asesor.cve_asesor = servicio_asesor.cve_asesor
and asesor.cve_grupo = var_grupo
```

Figura 22. Sentencia SQL para consultar los servicios abiertos del Grupo del usuario actual del Sistema.

Donde *var_grupo* será la variable global que corresponde a la clave del grupo al que pertenece el usuario que actualmente se encuentra utilizando el sistema.

Roles.

Los roles se utilizan para definir los distintos perfiles de permisos para asociar a un módulo con los usuarios del sistema que como ya se mencionó, en este caso se registran en la entidad *ASESOR*. Dicho de otra forma los roles definen que módulos pueden ser ejecutados por los usuarios del sistema asociados a un rol en específico. Mediante los roles se asocian los permisos de acceso a los módulos y procesos del sistema.

A continuación analizaremos la forma de asociar los módulos y procesos del sistema con los roles, a su vez veremos como están definidos los roles y como se asocia un usuario con un rol.

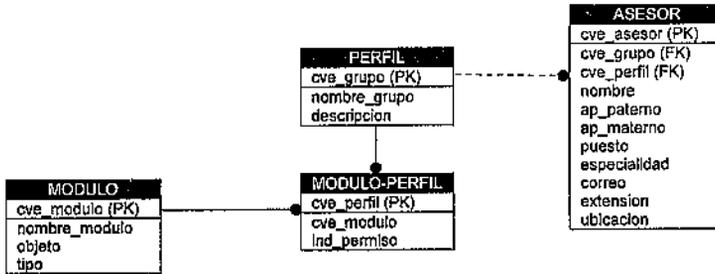


Figura 23. Extracto del diagrama E-R. Se presentan las entidades MODULO, MODULO-PERFIL, PERFIL y ASESOR y las relaciones entre las mismas.

En la entidad *MODULO* se registran los módulos del sistema, el campo destinado para el objeto almacenará el nombre del archivo .php, que estará relacionado con el proceso o módulo según se indique en el campo tipo.

La entidad *PERFIL* esta destinada para almacenar en sus registros los diferentes roles o perfiles que se pueden asociar a los usuarios del sistema mediante la relación uno a muchos que relaciona a las entidades *PERFIL* y *ASESOR*.

La entidad *MODULO-PERFIL* se utiliza para asociar los módulos a los perfiles existentes definidos en la entidad con el mismo nombre, para asignar los permisos correspondientes, el campo *ind_permiso* se utiliza para este fin, cuando el valor sea "1" indicara que se cuenta con permiso para la ejecución del proceso o módulo al que se asocia. Cuando el valor del campo *ind_permiso* sea "0" el permiso de ejecución estará negado.

A continuación se muestra una relación de los perfiles sugeridos para los usuarios del SIRESE.

Perfil	Descripción	Deberá tener acceso a
Asesor	Personal encargado de registrar servicios, así como también se le pueden asignar servicios, por lo que deberá dar solución y cerrar los mismos.	- Módulo de Seguridad - Módulo de Administración de Servicios - Módulo Monitor de Servicios
Supervisor	Es la persona encargada de dar seguimiento a los servicios	- Módulo de Seguridad - Módulo de Administración

	registrados de su grupo, por lo que se debe contar por un supervisor por grupo. Tiene la tarea de monitorear constantemente el número de registros pendientes de los asesores de su grupo, en caso de detectar alguna contingencia implementar el plan acertado para solucionar el problema. Deberá tener los privilegios para abrir y asignar un servicio, cuando su estado actual sea cerrado.	<ul style="list-style-type: none"> - de Servicios - Módulo Monitor de Servicios - Módulo de Reportes
Administrador	Será el administrador del sistema. Encargado de mantener la información de los catálogos actualizada. Así como dar de alta usuarios del sistema, asignarles permisos, etc.	<ul style="list-style-type: none"> - Módulo de Seguridad - Administración del Sistema
Gerencia	Este perfil se asignará a las personas de la Dirección de cada departamento y a la Jefatura de la Unidad de Sistemas. Este perfil está enfocado principalmente a la generación de reportes.	<ul style="list-style-type: none"> - Módulo de Seguridad - Módulo Monitor de Servicios - Módulo de Reportes

Tabla 9. Perfiles de los usuarios el sistema SIRESE.

Como se puede apreciar los roles son una representación de los actores que se modelan en el diagrama Casos de Uso del Capítulo III "Análisis del Sistema".

La generación del menú también está muy relacionado con las entidades que se analizaron anteriormente. El menú correspondiente al usuario que ingrese al sistema se generará dinámicamente de acuerdo al perfil que tenga asociado. A continuación se presenta la sentencia SQL utilizada para la generación del *recordset* que contendrá los elementos para la generación del menú.

```

select MODULO.nombre, MODULO.objeto, MODULO.tipo from
MODULO
  inner join MODULO-PERFIL
    on MODULO_PERFIL.cve_modulo = MODULO.cve_modulo and
MODULO_PERFIL.ind_permiso = '1'
  inner join ASESOR
    on ASESOR.cve_perfil =
MODULO_PERFIL.cve_perfil and ASESOR.cve_asesor =
varasesor

```

Figura 24. Código SQL para la generación dinámica del menú del usuario del sistema.

Donde *varAsesor* será la variable de sesión donde se almacena el nombre de usuario que ingreso al sistema, con lo que se tomaran los registros del record set para generar el menú dinámicamente, para lo cual *MODULO.nombre* será la etiqueta de la opción del menú que corresponde al proceso o módulo el cual será llamado por el dato que corresponde a *MODULO.objeto*.

Clasificación de los Servicios

La clasificación de los servicios a registrar, es la forma de clasificar los servicios proporcionados por la Unidad de Sistemas, esta clasificación esta basada en las actividades diarias de cada departamento de la Unidad de Sistemas por lo que cada uno de ellos generó una clasificación de los mismos. Esta clasificación permitirá elaborar reportes detallados de los servicios que se proporcionan y clasificarlos de acuerdo a su tipo, por lo que será un buen indicador para detectar posibles vulnerabilidades o recurrencia en ciertos aspectos del funcionamiento del equipo de cómputo, con lo que será posible tomar decisiones adecuadas a partir de la información generada como base en la clasificación de los servicios proporcionados.

A continuación se muestra un listado de la clasificación de los servicios que se registrarán en el sistema.

Clasificación de los Servicios a Registrar

1er. Nivel	2do. Nivel	3er. Nivel
Software		
	Office	
		Word
		Excel
		PowerPoint
		Outlook
	Sistema Operativo	
		Windows 95
		Windows 98
		Windows XP
		Windows 2000
	Aplicaciones	
		Antivirus
		Explorador de

1er. Nivel	2do. Nivel	3er. Nivel
		Internet
		Correo Electrónico
		Acrobat Reader
		Corel Draw
Hardware		
	CPU	
	Monitor	
	Teclado	
	Mouse	
	No-Break	
	Impresora	
	Terminal	
	Otros Dispositivos	
	Servicios Varios	
	Mantenimiento Correctivo	
	Mantenimiento Preventivo	
Conexión a Red	Software	
		Cliente Novell
		Proceso de Log On
	Hardware	
		Tarjeta de Red
		Cableado
Centro de Computo		
	Cuentas de Usuario	
		Mantenimiento
		Nueva
		Bloquear
		Asignar Password
	Transferencia de Archivos	
	Direcciones IP	
		Asignar
		Modificar
		Consultar
	Procesos	
		Iniciar
		Cancelar

Tabla 10. Clasificación de los Servicios a Registrar.

Como se puede observar la clasificación de los servicios esta basada en una estructura jerárquica, donde existen varios niveles. Cuando se registre un servicio este deberá ser clasificado de acuerdo a su tipo, de ser posible llegando al nivel más particular. A continuación se analiza la forma en la que los datos son almacenados para generar la estructura jerárquica de la clasificación de servicios.

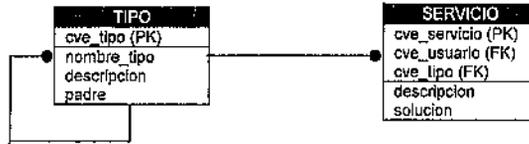


Figura 25. Extracto del diagrama E-R. Se muestran las tablas de las entidades TIPO y SERVICIO y sus relaciones.

Pongamos atención en el atributo *padre* que define a la entidad *TIPO* y a la relación del tipo uno a muchos que recursivamente apunta a la misma entidad, la cual se leería de la siguiente forma: Un *TIPO* (de servicio) puede pertenecer a uno o más *TIPO*. Lo cual puede ser lógico si lo vemos desde el punto de vista de que una clasificación padre o de primer nivel como por ejemplo *Software* (ver Tabla 10), tiene varios elementos los cuales deberán tener como valor en el campo *padre* la *clave_tipo* del registro correspondiente a *Software*. De esta forma se va definiendo la estructura jerárquica para la clasificación de los servicios.

De la relación uno a muchos que relaciona a las entidades *TIPO* y *SERVICIO* podemos deducir que un servicio solo puede estar clasificado con un tipo de servicio, como lógicamente debe ser.

Una de las ventajas de elegir este tipo de estructura para la clasificación de los servicios es la siguiente: los reportes que se generen con la información del sistema, están dirigidos al personal operativo, así como para los niveles directivos donde los reportes deben resumir la información en un aspecto más general, estos reportes se van a generar pasando como parámetro el primer nivel de la clasificación de los servicios, por ejemplo para el reporte trimestral de actividades, donde el detallado de que se realizó en cada servicio no es importante.

Por otro lado a la dirección de un departamento en específico le interesa conocer el número de reportes atendidos de acuerdo a la clasificación de los servicios del tercer nivel, con lo que podrá determinar, de acuerdo a las reincidencias, que medidas tomar para disminuir el número de problemas reportados en algún rubro en específico. Por ejemplo si se detecta que en el mes de febrero del año 2004, el número de reportes

atendidos, de acuerdo a la clasificación del tercer nivel "*Word*" es alta por arriba de las demás clasificaciones, se podrá considerar programar cursos para las áreas con mayor reincidencia. Si es necesario se pueden obtener reportes con el detallado de los servicios proporcionados, donde se puede detectar con mayor precisión las áreas involucradas y los problemas que se atienden en específico.

CONCLUSIONES

El análisis del sistema es la etapa en la que se realiza una investigación acerca del problema que se pretende resolver y cómo puede ser resuelto. En esta fase del desarrollo, la comunicación entre los analistas, los programadores y los usuarios es de gran importancia para que el sistema cumpla con las expectativas planteadas.

El primer paso en la etapa de análisis es definir el problema. Una vez hecho esto el problema debe de ser definido y delimitado en conjunto con los usuarios, los diseñadores del sistema, y los desarrolladores.

Una vez definido el problema, el siguiente paso es analizarlo y obtener la lista de requerimientos. Uno de los aspectos más importantes para que un sistema Web y cualquier tipo de sistema, sea exitoso, es que éste cubra por completo las necesidades para la que fue planeado.

La alternativa de la elaboración de un sistema en Web, en la actualidad es muy utilizada por su facilidad para ser distribuido. El desarrollo de aplicaciones Web es un campo relativamente nuevo para el cual existen diversas propuestas, una propuesta interesante de desarrollo de aplicaciones Web es utilizar software libre.

La integración de Linux, Apache, MySQL y PHP, es la tecnología más utilizada para la creación de servicios implementados en Web. Su libre distribución, robustez, soporte a gran cantidad de arquitecturas; seguridad y rapidez la convierten en una de las alternativas más factibles.

Otra característica muy interesante de esta arquitectura es el hecho de que estos cuatro productos pueden funcionar en una amplia gama de hardware, con requerimientos relativamente pequeños, sin que por ello sean menos estables que sistemas implementados en equipos de gran capacidad.

En un sistema al que "todo el mundo tiene acceso", es importante reforzar las medidas de seguridad para no permitir el acceso a usuarios no autorizados. Lo cual se puede lograr seleccionando las herramientas adecuadas. Por ejemplo seleccionar un servidor Web que cuente con extensiones del protocolo TCP-IP para conexiones seguras donde los datos viajaran encapsulados, por lo que no pueden ser interpretados.

La fase del diseño, o modelado de la implementación de un sistema, es una parte trascendental de preparación para pasar a la etapa de programación y ya propiamente a la fase de la implantación. Podemos decir que lo que se elabore en esta fase será el medio de comunicación con los programadores donde a diferencia de lo modelado en la etapa de análisis, aquí todo se estudia a un nivel más detallado, a tal grado que se definen las reglas de negocio del sistema, las cuales son la base para el funcionamiento de los módulos donde se utilicen y en algunos casos la integridad de la información, depende de las mismas.

La utilización de la arquitectura de tres capas es una buena opción para la implementación de un sistema de información en Web, debido a lo siguiente:

Capa de usuario: Si se requieren realizar cambios en la interfase, se puede asegurar que el código que define los procedimientos y transacciones no se verá afectado ya que es totalmente independiente del código que define la interfaz del usuario.

Capa de reglas de negocio: Quizá es donde más ventajas tenga el utilizar esta tecnología, ya que el código de los procedimientos del sistema, así como de las transacciones, puede ser reutilizado en cualquier módulo del sistema. Cuando alguno de los procedimientos cambia, los cambios se hacen únicamente en una sección del código, el cual define el procedimiento que cambio, ventaja invaluable a la hora de dar mantenimiento al sistema. Aunque, cabe señalar que el precio por esta ventaja fue pagado a la hora de programar el sistema ya que no siempre es separar la definición de los procedimientos o reglas de negocio, de la capa de datos y de la interfaz de usuario, es la vía más sencilla, en ocasiones los procedimientos se vuelven más complejos a la hora de programar.

Capa de Datos: Una ventaja, y sin lugar a dudas, con la ayuda de la capa de reglas e negocio, esta arquitectura permite interactuar con varias bases de datos aunque estas estén definidas en distintos SGBD (Sistema Gestor de Bases de Datos), con lo que el sistema podrá obtener información de sistemas ya implementados en la institución si así se requiere.

Ya que fueron sorteados todos los problemas técnicos de la implementación, podemos pensar que el procedimiento para la implantación del sistema será una tarea con un grado de dificultad menor, si se toma en cuenta que los usuarios directos del sistema son personas cuyo perfil profesional esta relacionado directamente con alguna rama de las ciencias de la computación. Pero la realidad fue otra, aunque estaban conscientes de la necesidad de la implementación del sistema y de los beneficios que traería consigo como por ejemplo: la elaboración de reportes de forma automática; no fue tarea sencilla que adoptaran la nueva forma para registrar los servicios, argumentando, que el tiempo que tardaban en registrar un servicio, muchas veces, era mayor que el tiempo que les llevaba resolver un problema sencillo. Lo cual probablemente sea cierto y no por que el sistema fuera deficiente ya que este aspecto se tomo en cuenta a la hora de diseñarlo y únicamente se solicitaba la información sustancial. El dedicar tiempo para registrar los servicios, recursos de Hardware y humanos para la implementación del sistema, se justifica por lo siguiente:

- Es importante registrar todas las actividades que llevan a cabo, por sencillas que sean, para que no pasen desapercibidas.
- Generar reportes con la información de forma automática.
- La toma de decisiones se refuerza con información obtenida del comportamiento de la infraestructura informática y de las necesidades de los usuarios.
- Es importante contar con una base de conocimientos para retroalimentarse y eficientar la solución de los problemas, así como para la elaboración de notas técnicas y ayudar a la integración del personal de nuevo ingreso.

Sistematizar cualquier tarea por sencilla e intrascendente que parezca, siempre traerá ventajas, sobre todo a los usuarios del sistema.

INDICE DE FIGURAS

Figura	Descripción	Página
1	Estructura de la Unidad de Sistemas.	10
2	Diagrama de la arquitectura de implementación de las tecnologías de Cliente y de Servidor para páginas Web dinámicas.	22
3	Elemento para representar un Actor en los diagramas de Casos de Uso.	25
4	Ejemplo del elemento para representar un Caso de Uso.	26
5	Representación gráfica de una entidad.	28
6	Representación de los atributos de una entidad en un diagrama E-R.	29
7	Contenido del archivo "ejemplo.html".	39
8	Vista del archivo "ejemplo.html" interpretado por un visor de páginas Web.	40
9	Arquitectura utilizada para la implementación del sistema SIRESE.	45
10	Porcentaje de los Servidores Web implementados en los dominios de Internet (1995 - 2005).	47
11	Total de Servidores Web activos (2000 – 2005).	48
12	Casos de Uso del Sistema.	54
13	Diagrama Entidad-Relación.	64
14	Esquema general de red para la implementación del SIRESE.	71
15	Diagrama de flujo del proceso de registro y atención de servicios.	72
16	Diagrama de Procesos del SIRESE.	74
17	Pantalla de Inicio de Sesión, SIRESE.	75
18	Pantalla del Catalogo de Asesores del Módulo Administración de Catálogos.	78
19	Pantalla para el Registro de los Servicios.	79
20	Pantalla del Módulo Monitor de Servicios.	80
21	Extracto del diagrama E-R. Se presentan las entidades GRUPO y ASESOR y la relación existente entre ambas.	81
22	Sentencia SQL para consultar los servicios abiertos del Grupo del usuario actual del Sistema.	82
23	Extracto del diagrama E-R. Se presentan las entidades MODULO, MODULO-PERFIL, PERFIL y ASESOR y las relaciones entre las mismas.	83
24	Código SQL para la generación dinámica del menú del usuario del sistema.	84
25	Extracto del diagrama E-R. Se muestran las tablas de las entidades TIPO y SERVICIO y sus relaciones.	87

INDICE DE TABLAS

Tabla	Descripción	Página
1	Datos de la cotización de la empresa Network Associates.	16
2	Tipos de datos para definir a un atributo.	28
3	Breve descripción de los niveles de normalización.	30
4	Notación empleada en el diccionario de datos.	32
5	Principios Generales de Usabilidad.	38
6	Módulos adicionales para el Servidor Web Apache.	46
7	Descripción del estado de los servicios registrados.	73
8	Información requerida para el registro de una solicitud de servicio.	77
9	Perfiles de los usuarios el sistema SIRESE.	84
10	Clasificación de los Servicios a Registrar.	86

BIBLIOGRAFÍA

"Análisis Estructurado Moderno"

Edgard Yourdon.
Prentice Hall, 1989.

"Apache Server"

Manuel Alberto Ricart.
Sams Net, 1996.

"Apache. The Definitive Guide"

Ben Laurie, Meter Laurie.
O'Reilly, 1997.

"Apuntes del Diplomado: Desarrollo de Sistemas en Web"

Marcelo Pérez Medel, Omar Mendoza González, Víctor Aguilar Ocampo, Rodolfo Vázquez Morales, Cesar Germán Rosas.
UNAM, ENEP Aragón, 2003-2004.

"Bases de Datos Relacionales"

Enrique Rivero Cornelio.
Paraninfo, 1998.

"Choosing a Client/Server Architecture. A Comparison of Two-Tier and Three-Tier Systems."

Gallaugh, J. & Ramanathan, S.
Information Systems Management, 1996.

"Creación de aplicaciones Web con PHP"

Ratschiller & Gemen.
Prentice Hall, 2001.

"Diseño de Bases de Datos Relacionales"

Adoración de Miguel Castaño, Mario Piattini Velthuis.
Alfaomega, 2000.

"Fundamentos de Bases de Datos"

Henry F. Kart, Abraham Silberschatz.
Mc Graw Hill, 1993.

"Runing a Perfect Web Site"

Stephen Wynkoop.
QUE, 1996.

“The Unified Modeling Language User Guide”

Grady Booch, James Rumbaugh, Ivar Jacobson.
Addison-Wesley, 1999.

“The Web Server Book: Tools & Techniques for building your own Internet Information Site”

Jonathan Magid.
Ventana, 1995.

“UML Gota a Gota”

Martin Fowler.
Addison-Wesley, 1999.

“Using the World Wide Web”

David Wall, Karen Cooper, William Eager.
QUE, 1996 .

“Web Site Construction”

Richard Schwartz.
Onward Press, 2001.

“Web Security”

Amrit Tiwana.
Boston, 1999.

BIBLIOGRAFÍA ELECTRÓNICA

"Arquitectura en tres capas (Diagrama y Definición)"

<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art140.asp>

"Aplicaciones en tres capas"

<http://www.lsi.us.es/~tdg/res/3-tier-SP/tsid001.htm>

"Client/Server Past, Present, and Future"

<http://news.dci.com/geos/dbsejava.htm>

George Schussel, 1995.

Documentación en línea de PHP

<http://www.php.net/docs.php>

Documentación en línea de MySQL

<http://www.mysql.com>

Documentación en línea de Apache

<http://httpd.apache.org/docs-project/>

"Estrategias de ecommerce y ebusiness"

<http://www.desarrolloweb.com/manuales/23>

José Ramón Pais Curto.

"Publicar en Internet"

<http://www.desarrolloweb.com/manuales/3>

Miguel Ángel Álvarez.

"Three Tier Software Architectures"

<http://www.sei.cmu.edu/str/descriptions/threetier.html>

"UML Tutorial"

http://www.sparxsystems.com.au/UML_Tutorial.htm

Sparx Systems, 2000-2005.