



**BENEMÉRITA UNIVERSIDAD
AUTÓNOMA DE PUEBLA**

**Facultad de Ciencias de la
Computación**

**SISTEMA DE CONTROL DE ACCESO A
LA DIRECCIÓN DE INFRAESTRUCTURA
Y SOPORTE TÉCNICO DE LA
SECRETARÍA DE FINANZAS Y
ADMINISTRACIÓN DEL ESTADO DE
PUEBLA**

TESIS PROFESIONAL

**QUE PARA OBTENER EL GRADO DE:
LICENCIADA EN CIENCIAS DE LA
COMPUTACIÓN**

PRESENTA:

PATRICIA DOMÍNGUEZ MIRÓN

ASESOR:

DR. MARIO ROSSAINZ LÓPEZ

Puebla, México 2015

AGRADECIMIENTOS.

A mis maestros:

Dra. YOLANDA RAMIREZ ENCARNACIÓN, por enseñarme que con esfuerzo y dedicación se puede llegar al objetivo deseado, y por motivarme a explorar el maravilloso mundo de las matemáticas.

Especialmente al Dr. MARIO ROSSAINZ LÓPEZ, por el apoyo, la paciencia y dedicación para guiar el desarrollo de este trabajo de tesis y llegar a la culminación del mismo.

Al Ing. ALEJANDRO FLORES TORRES, jefe a cargo de la DIST en la Secretaria de Finanzas del Estado de Puebla, por las facilidades otorgadas en el diseño de este proyecto.

A todos mis amigos (a) s y compañeros:

Gracias por estar conmigo en las buenas y en las malas, pero sobre todo por sus consejos y apoyo incondicional.

Al Mtro. JOSE CASTILLEJOS LOPEZ, por la gran amistad y asesoría que siempre me ha brindado.

A JESUS DANIEL MARTINEZ ABUNDIS, gracias por el apoyo y la comprensión que siempre me tuviste para que lograra esta meta, por el amor y la paciencia con la que siempre me has apoyado.

“Las personas no son recordadas por el número de veces que fracasan, sino por el número de veces que tienen éxito”.

Thomas Alva Edison.

DEDICATORIAS.

A mis padres:

MIGUEL ANGEL y BLANCA ESTHER, gracias por guiar mi camino y estar junto a mí en los momentos más difíciles, por confiar en mí pero sobre todo gracias por darme la oportunidad de estudiar una licenciatura.

A mis hermanos:

Quienes no tuvieron las mismas oportunidades, pero gracias a su trabajo y esfuerzo logré concluir esta etapa. A OSCAR por enseñarme que siempre que se quiere algo se puede obtener con trabajo, persistencia, y la lucha diaria ante las adversidades de la vida.

A mis padrinos:

COSME y LOURDES, de quienes he aprendido que con esfuerzo y dedicación se pueden cumplir los objetivos deseados.

A todas aquellas personas que creyeron en mí, y que ahora comparten conmigo este triunfo no me queda más que decir: MUCHAS GRACIAS.

“Es mejor cojear por el camino que avanzar a grandes pasos fuera de él. Pues quien cojea en el camino, aunque avance poco, se acerca a la meta, mientras que quien va fuera de él, cuanto más corre, más se aleja”.

San Agustín

Contenido

CAPÍTULO 1: INTRODUCCIÓN.....	7
1.1 RESUMEN.....	8
1.2 PROYECTO DE ORIGEN.....	9
1.3 OBJETIVOS GENERALES Y ESPECÍFICOS.....	9
1.4 ORGANIZACIÓN DEL PROYECTO DE TESIS.....	10
CAPÍTULO 2: MARCO TEÓRICO.....	12
2.1 ANTECEDENTES.....	13
2.1.1 <i>Bitácora manual</i>	15
2.1.2 <i>Bitácora electrónica en Access</i>	16
2.1.3 <i>Bitácora electrónica Joomla, PHP, y MySQL</i>	19
2.2 ESTADO DEL ARTE.....	20
2.2.1 <i>Control de visitas de un centro educativo</i>	21
2.2.2 <i>Aplicación para registrar accesos al DIST</i>	24
2.2.3 <i>Gestor integral de una clínica de rehabilitación</i>	27
CAPÍTULO 3: PLANEACIÓN DEL PROYECTO.....	33
3.1 DEFINICIÓN DEL PROBLEMA.....	34
3.2 ESTRATEGIA DE SOLUCIÓN.....	35
3.3 ADECUACIÓN DEL PROYECTO EN LA PLANEACIÓN.....	36
3.3.1 <i>Infraestructura</i>	40
3.3.2 <i>Software</i>	40
CAPÍTULO 4: ANÁLISIS DE REQUERIMIENTOS.....	41
4.1 IDENTIFICACIÓN, LISTADO DE ACTORES Y CASOS DE USO.....	42
4.2 DIAGRAMAS DE CASOS DE USOS.....	46
4.3 REALIZACIÓN DE LOS CASOS DE USO.....	48
4.3.1 <i>Secuencias normales y excepciones de los CU</i>	51
4.3.2 <i>Diagramas de secuencia para cada CU</i>	56
CAPÍTULO 5: DISEÑO.....	67
5.1 DISEÑO CONCEPTUAL DEL PROYECTO.....	68
5.1.1 <i>Identificación de las clases u objetos del sistema</i>	70

5.2	DISEÑO DE NAVEGACIÓN DEL PROYECTO.....	72
5.2.1	<i>Diagramas de clases para modelar la estructura de navegación.</i>	74
5.2.2	<i>Uso de storyboard para modelar la presentación del proyecto.</i>	76
5.3	DISEÑO DE ESCENARIOS DEL PROYECTO.	81
CAPÍTULO 6: IMPLEMENTACIÓN.....		91
6.1	SISTEMA CLIENTE - SERVIDOR.....	92
6.1.1	<i>Implementación y pruebas del sistema.</i>	94
6.2	SISTEMA DE BASE DE DATOS.	112
6.2.1	<i>Sistema de administración de datos.</i>	114
6.3	CONCLUSIONES Y PERSPECTIVAS.....	115
APÉNDICE: A.		117
A.1	MODELO ENTIDAD-RELACIÓN.....	118
A.1.1	<i>Entidad.</i>	118
A.1.2	<i>Relación.</i>	118
A.1.3	<i>Atributo.</i>	119
A.1.4	<i>Identificador.</i>	121
A.1.5	<i>Jerarquía de generalización.</i>	121
A.2	MODELO RELACIONAL.	124
A.2.1	<i>Estructura de datos.</i>	125
A.2.2	<i>Propiedades de relaciones.</i>	126
A.2.3	Claves.	126
A.3	CÓDIGO DE LA BITÁCORA DIST.....	135
APÉNDICE: B.....		138
B.1	WAMP SERVER V2.5	139
B.1.1	<i>Apache.</i>	139
B.1.2	<i>MySQL.</i>	140
B.1.3	<i>PHP.</i>	141
B.2	JOOMLA V3.4.....	142
B.2.1	<i>Instalación de joomla.</i>	143
B.2.2	<i>HTML.</i>	144
B.2.3	<i>JavaScript.</i>	145

<i>B.2.4 CSS</i>	<i>146</i>
BIBLIOGRAFÍA	148

Capítulo 1: Introducción.

1.1 Resumen.

El siguiente proyecto de tesis propone mediante la implementación de un sistema computacional, un control de acceso a la zona restringida del departamento denominado: Dirección de Infraestructura y Soporte Técnico (DIST), de la Secretaría de Finanzas y Administración (SFA) del Estado de Puebla.

Dicho proyecto se lleva a cabo con la finalidad de contar con una aplicación computacional en la cual se evite la pérdida de información, así como también se tenga un mejor control de acceso y permita al administrador la consulta confiable y verídica de la información que requiera mediante la generación de informes o reportes digitales e impresos. Tal aplicación registrará la fecha, la hora de entrada y salida, asunto, departamento, empresa y demás datos, del personal autorizado al DIST.

Hay que tomar en cuenta que el personal puede ser interno o externo a la SFA, ya que hay proveedores que también ingresan a dicho departamento, así como en ocasiones pueden ingresar personas que pertenecen a cualquier otro departamento de la SFA.

1.2 Proyecto de origen.

Este proyecto tiene su origen en el Departamento de Infraestructura y Soporte Técnico (DIST), de la Secretaría de Finanzas y Administración (SFA) del estado de Puebla y surge como una necesidad de automatización de un procedimiento de control de entradas/salidas de personas al DIST, además de interesarse por la concientización de cuidado al medio ambiente. Es por eso que se decidió implementar un sistema computacional para tener un control eficiente de acceso que se realiza en el Centro de Cómputo.

1.3 Objetivos generales y específicos.

Objetivo General:

- ❖ Analizar, Diseñar, Implementar y poner en marcha un sistema computacional que permita llevar un control de acceso de entradas y salidas del personal de SFA al centro de datos.

Objetivos Específicos:

- ❖ Generar el sistema computacional de forma que permita tener un buen control del personal que accede al centro de cómputo.
- ❖ El Sistema permitirá al usuario saber cuántas veces accedió en el mes.

- ❖ El sistema permitirá al administrador estar informado de los datos del personal que accedió al centro de datos.
- ❖ El sistema computacional será una aplicación Web implementada en PHP usando como motor de base datos MySQL.

1.4 Organización del proyecto de tesis.

CAPITULO 1: Introducción.

En este capítulo se presenta un resumen del proyecto, el origen de dicho proyecto, así como los objetivos generales y específicos.

CAPITULO 2: Marco teórico.

Aquí se presentan los antecedentes que dan origen al proyecto, así como el estado del campo de arte.

CAPITULO 3: Planteamiento del proyecto.

Se establece la definición del Problema, la Estrategia de Solución así como la adecuación del Proyecto en la Planeación.

CAPITULO 4: Análisis de Requerimientos.

Se plantea la Identificación y listado de actores y casos de uso, Diagramas de Casos de uso, Realización de los casos de uso, Secuencias normales de ejecución de los casos de uso, Excepciones de ejecución de los casos de uso.

CAPITULO 5: Diseño del Proyecto.

Se definen los siguientes diseños: Diseño Conceptual, Diseño de Navegación, Diseño de Escenarios del proyecto.

CAPITULO 6: Implementación.

Se implementan los sistemas: Sistema Cliente, Sistema Servidor, Sistema de Base de Datos, Pruebas.

CAPITULO 7: Conclusiones y Perspectivas.

APÉNDICE A: Diseño de la Base de Datos del proyecto

APÉNDICE B: Uso de la arquitectura software utilizada para implementar el proyecto.

Capítulo 2: Marco teórico.

2.1 Antecedentes.

El grupo de trabajo de la Dirección de Infraestructura y Soporte Técnico (DIST); el cual administra el Centro de Datos (SITE) está integrado por una jerarquía de 7 elementos. Ver Figura 1.

Los cuales se encargan de la disponibilidad y mantenimiento de la infraestructura del Centro de Cómputo, así como de la ejecución de respaldos de toda la información que en él se maneja.

Y cuyo principal objetivo es:

Formar entre todos a la Dirección de Sistemas, para que actúe en la dependencia dotando a quien corresponda, con información sobre controles, dirección, supervisión, reuniones, retroalimentación, y actualizaciones en temas con prioridad, teniendo herramientas que faciliten su aplicación, con el fin de implementar y mantener el trabajo de la semana/mes/trimestre con la mejora continua.

Debido a que el área a la que se asignará este proyecto es una área restringida la cual es monitoreada las 24 horas del día, es urgente tener una herramienta computacional la cual sea específicamente para el control del acceso a dicho departamento.

Tal aplicación fue solicitada en el periodo de servicio social realizado por una servidora, por la Secretaria de Finanzas y Administración del Estado de Puebla; Para ser más específicos en la Dirección de Infraestructura y Soporte Técnico (DIST).

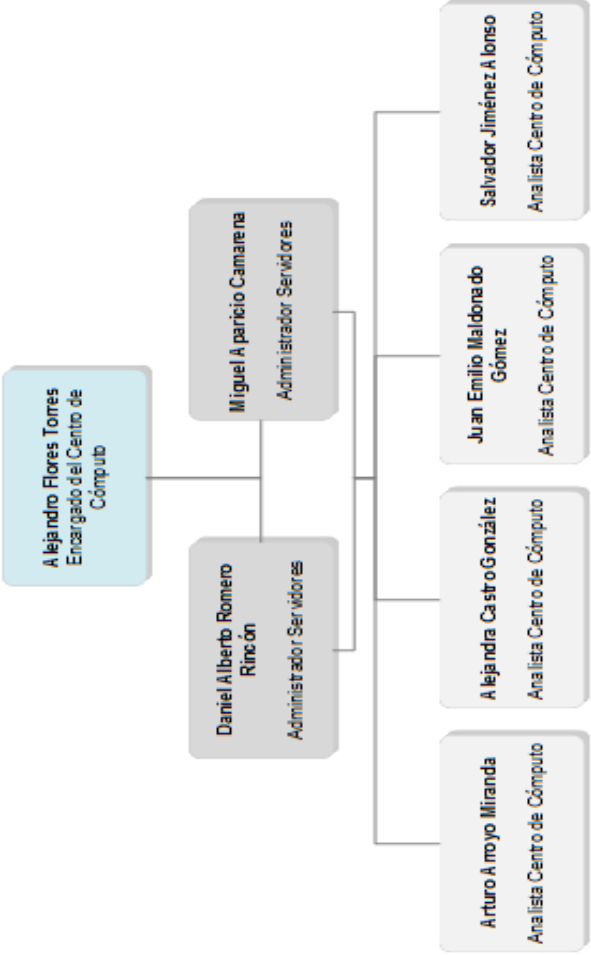


Figura 1: Jerarquía del grupo de trabajo.

2.1.1 Bitácora manual.

Cuando ingrese a desarrollar mi servicio social en DIST; en el departamento al cual fui asignada denominado SITE; contaba con una bitácora física y manual, en la cual se llevaba a cabo el registro del personal que accedía a tal departamento; la bitácora era un libro de registros en donde los usuarios se registraban con los siguientes campos:

- ❖ NOMBRE.
- ❖ APELLIDO.
- ❖ EMPRESA/DEPARTAMENTO.
- ❖ MOTIVO DE VISITA.
- ❖ FECHA.
- ❖ HORA DE ENTRADA.
- ❖ HORA DE SALIDA.
- ❖ FIRMA.

Debido a que era necesario escribir en un libro de registros; la bitácora no era 100% segura ya que se podía alterar la información con respecto a las horas de entrada/salida; así como la fecha.

Con esto se deriva la pérdida de información debido a que el usuario a veces solo registraba la entrada pero no la salida, y puede poner los datos que quiere aunque estos no sean verídicos.

Con motivo de eliminar la bitácora física que se tenía y así favorecer la ecología y un mejor manejo de los datos, se decidió implementar una herramienta electrónica o

sistema de cómputo para tener un control eficiente del acceso que se realiza en el Centro de datos.

2.1.2 Bitácora electrónica en Access.

Algunas semanas después; viendo que los registros que se llevaban a cabo no eran 100% seguros y que algunas veces los usuarios no completaban su registro; se procede a cambiar la bitácora manual por una bitácora electrónica.

Arturo Arroyo Miranda (Analista) comienza la búsqueda de una aplicación que permita llevar dicho control.

Esta aplicación fue descargada de internet y desarrollada en Access. Ver Figura 2.

La bitácora descargada contiene los siguientes campos:

- ❖ ID.
- ❖ NOMBRE.
- ❖ APELLIDOS.
- ❖ COMPAÑÍA.
- ❖ ASUNTO.
- ❖ VISITANTE.
- ❖ FECHA/HORA DE ENTRADA.
- ❖ FECHA/HORA DE SALIDA.
- ❖ EXPIRA.
- ❖ FOTO/IMAGEN.

Además cuenta con los siguientes botones:

- ❖ AGREGAR REGISTRO.
- ❖ DUPLICAR REGISTRO.
- ❖ IMPRIMIR PASE.
- ❖ EXPORTAR REGISTROS.
- ❖ ELIMINAR.
- ❖ SALIR.

The screenshot shows a web application interface for visitor registration. At the top, there is a navigation bar with 'Archivo' and 'frmVisitorsEntry'. Below this is a blue header with the title 'Registro de Visitantes'. The main area is divided into two columns. The left column contains a list of labels for visitor information: 'Badge ID', 'Nombre:', 'Apellidos', 'Compañía:', 'Asunto', 'Visitante', 'Fecha/hora Entrada', 'Expires:', 'Fecha/hora Salida', and 'Foto/Imagen'. The right column contains a large blue logo for 'DIST' (DIRECCIÓN DE INFRAESTRUCTURA Y SOPORTE TÉCNICO) and a vertical stack of buttons: 'Agregar Registro', 'Duplicar Registro', 'Imprimir pase', 'Exportar Registros', 'Eliminar', and 'Salir'. The 'Visitante' field is pre-filled with 'Centro de Cómputo' and the 'Fecha/hora Entrada' field is pre-filled with '08/08/2014 21:00:30'. The URL 'www.URTech.ca' is visible at the bottom left.

Figura 2: Bitácora electrónica en ACCESS.

Con esta aplicación aparentemente el problema de la pérdida de información se había solucionado, pero existían otros errores.

En esta aplicación el usuario ingresaba los datos antes ya mencionados, el sistema agrega el registro a una base de datos, para después poder importarlos a archivo PDF o Excel.

El problema que ahora presentaba la aplicación es que el usuario seleccionaba la hora de entrada y esa es la que se registraba en la BD, es decir por ejemplo si la hora de entrada real era 3:40 pm el usuario podía seleccionar como hora de entrada 12:20 pm, con esto la información continua siendo manipulable.

Otro de los problemas que presenta esta aplicación es que no contiene registros específicos para cada tipo de usuario, ya que como se mencionó en el **CAPITULO 1**, existe personal interno y externo a la SFA, por tal motivo es necesario que cada tipo de personal tenga campos específicos, debido a que; por ejemplo el personal interno pertenece a un **departamento** determinado dependiendo el tipo de **dirección** en la cual se desarrolla. En cambio en el personal externo se requiere saber la **empresa** en la cual trabaja.

También esta aplicación contiene los siguientes botones innecesarios:

- ❖ **DUPLICAR REGISTRO:** No es conveniente duplicar un registro; ya que con esto se contribuye a que la información almacenada en la BD sea redundante.

- ❖ **IMPRIMIR PASE:** Para el acceso al SITE no es necesario imprimir un pase de registro, ya que las puertas cuentan con seguridad magnética y únicamente los encargados de dicho departamento pueden autorizar abrir la puerta de acceso después de realizar el registro.

- ❖ **EXPORTAR REGISTROS:** En este caso es necesario contar con este botón pero en un submenú para que únicamente los administradores del sistema puedan consultar la información almacenada en la BD.

- ❖ **ELIMINAR:** Al igual que con el botón de exportar; únicamente el administrador podría tener acceso, ya que si cualquier usuario elimina información entonces provocaría la pérdida de registros.

2.1.3 Bitácora electrónica Joomla, PHP, y MySQL

Con base a toda la problemática que observe, me permite plantearle al personal del SITE la creación de una nueva bitácora; la respuesta fue favorable y se me encomendó realizar tal aplicación.

En la presente tesis se propone el desarrollo de una bitácora electrónica desarrollada en Joomla, usando como lenguaje de programación PHP y como motor de búsqueda MySQL.

Con esta aplicación el administrador podrá hacer las consultas necesarias para saber cuántas veces entro un usuario, el motivo de la visita, así como las fechas, horas de entrada y salida del mismo además de otros datos;

También podrá editar o eliminar información de los registros almacenados en la BD, previo registró como administrador para garantizar la estabilidad del sistema.

Los usuarios únicamente podrán registrarse mediante un formulario creado para cada tipo de personal; sin poder manipular las horas de entrada/salida; así como la fecha ya que estas serán asignadas automáticamente por el sistema.

En el **CAPITULO 3** se explica más a detalle la implementación y las herramientas necesarias para llevar a cabo este proyecto.

2.2 Estado del arte.

En esta sección se analizaran algunas aplicaciones realizadas en joomla así como aplicaciones similares que se presentan a continuación.

2.2.1 Control de visitas de un centro educativo.

Dentro del proceso de promoción institucional de la Universidad Politécnica de Madrid (UPM), se realizan visitas informativas a distintos centros educativos de la comunidad. El proyecto surge a partir de la necesidad de gestionar estas visitas.

Para la realización del mismo, se ha desarrollado una aplicación web utilizando diferentes tecnologías, como han sido Hypertext Preprocessor (PHP), AJAX y Hyper Text Markup Language (HTML).

El desarrollo de esta aplicación comprende una parte de diseño y otra de implementación. En la parte de diseño se ha utilizado una metodología orientada a objetos.

En la Figura 3 se muestra la página de inicio o login. Posteriormente por cada contexto navegacional o subsistema definido en el mapa de navegación se crea una página web.



Figura 3: Login del sistema.

Los subsistemas únicamente modelan una agrupación lógica de contextos. Por tanto, las páginas generadas a partir de ellos contendrán únicamente la implementación de un menú a partir del cual poder acceder a cada uno de los contextos de exploración contenidos en el subsistema.

Modelo entidad relación: En este caso, los datos se han modelado según el diagrama E/R (Figura 4). Este diagrama se compone de cuatro bloques básicos.

- ❖ Gestión de los usuarios del sistema, la cual está formada por la jerarquía USUARIO.
- ❖ Gestión de los grupos que se crean en el sistema, junto con las operaciones que se asignan a cada grupo.

- ❖ Gestión de colegios, la cual engloba las enseñanzas de cada colegio y la pertenencia con las escuelas o facultades; así como la gestión de visitas.

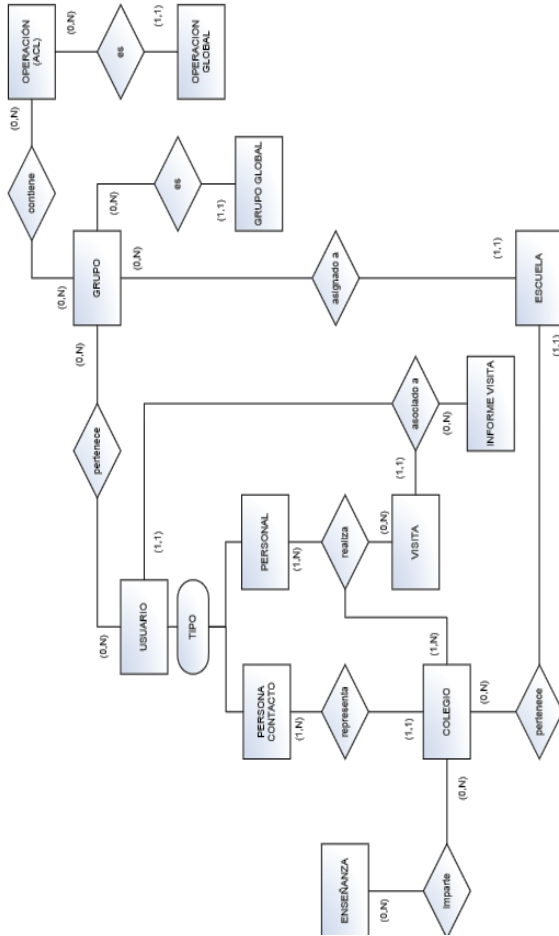


Figura 4: Modelo entidad relación.

2.2.2 Aplicación para registrar accesos al DIST.

El centro de datos SITE se vio en la necesidad de migrar de una bitácora física y manual a una bitácora electrónica; debido a que en la bitácora física no se llevaba un buen control de la información.

La herramienta utilizada para el registro del personal que accede al DIST, fue realizada en Microsoft Access; esta aplicación cuenta con un botón para exportar los registros, pero no cuenta con la seguridad suficiente ya que cualquier usuario puede exportar los datos y esa información es confidencial para el administrador.

Tal aplicación fue descargada de un portal de internet por Arturo Arroyo Miranda, quien ocupa el cargo de analista del centro de cómputo denominado SITE.

Por tal motivo no se tiene antecedentes o conocimientos previos sobre el uso de esta bitácora; y el uso de esta herramienta resulta complicado para el administrador.

Dicha herramienta tiene problemas al registrar la hora, ya que con frecuencia, la hora o fecha no coincidía con la de tiempo real.

Se puede apreciar cómo está conformada la pantalla principal, y los campos necesarios para proceder al registro de la visita en el sistema en cuestión (Figura 5)

De la misma manera se puede apreciar los pasos de instalación de la aplicación (Figura 6).



Figura 5: Pantalla principal del sistema.

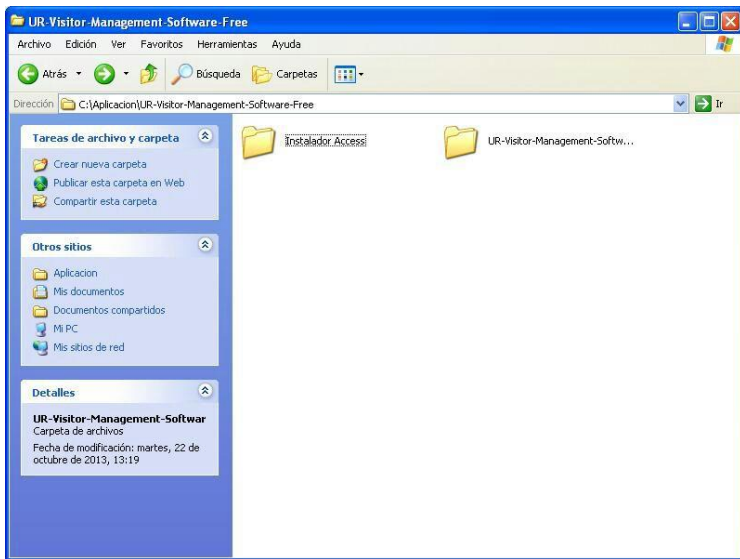


Figura 6: Instalación de la bitácora.

2.2.3 Gestor integral de una clínica de rehabilitación.

En este caso, se presenta la tesis profesional que presento Eder Osvaldo Espino López en diciembre de 2014 para obtener el título de licenciado en Ciencias de la computación de la Benemérita Universidad Autónoma de Puebla.

Esta aplicación ofrece crear lazos entre la medicina y la computación, para trabajar de la mano y facilitar a los pacientes mayor información acerca de los tratamientos que se les pueden ofrecer; en este trabajo de tesis se desarrolla una aplicación Web para la gestión integral de una clínica de rehabilitación.

Planteamiento del problema.

Es muy fácil encontrar uno de los problemas más comunes en los hospitales de México, la parte administrativa de ellos.

El problema de llevar a cabo una buena administración de todos sus datos como clínica, así como los datos de los pacientes, no solo consiste en usuarios y contraseñas sino de un minucioso registro para garantizar búsquedas confiables y eficientes.

El problema a resolver se propone solucionar con una aplicación Web, de manera que los accesos a la información por parte del médico así como la del paciente puede ser más amplia, en contextos previo y posteriores al tratamiento a recibir.

Aportaciones.

Se propone ayudar a la medicina actual con un sistema que les permita tener una administración eficiente de sus pacientes y que estos últimos puedan hacer de su conocimiento qué tipo de tratamiento se les podrá proporcionar.

Estado del Arte.

Este proyecto surge a raíz de las necesidades que conlleva tratar de tener una administración eficiente, que cualquier clínica debiese tener, para poder atender en tiempo y forma a sus pacientes, es decir evitar los largos tiempos de espera y cumplir con los horarios establecidos de los pacientes con cita, lo anterior para poder brindar una atención adecuada.

Esto hace más evidente la intervención de las tecnologías de la información para el apoyo que los médicos requieren para tener un mejor manejo de sus pacientes a través de una aplicación Web y tener un control de su información gracias a la implementación de base de datos para el almacenamiento de la información.

De esta forma se pretende realizar una relación más íntima entre la medicina y las nuevas tecnologías, lo cual proporcionaría una mayor certeza en lo que se realiza y sobre todo facilitar el trabajo del manejo de información de los usuarios y que se pueda brindar información a los usuarios de manera rápida y oportuna.

Modelo entidad relación.

El diagrama E-R del Sistema nos permite ver las entidades con sus atributos, así como las relaciones que existen entre ellas con la cardinalidad correspondiente.

La Figura 7 muestra el diseño conceptual de la base de datos, formada por 2 entidades, con atributos y 1 relaciones con cardinalidad de muchos a muchos. Contenido se refiere al contenido de la página Web.

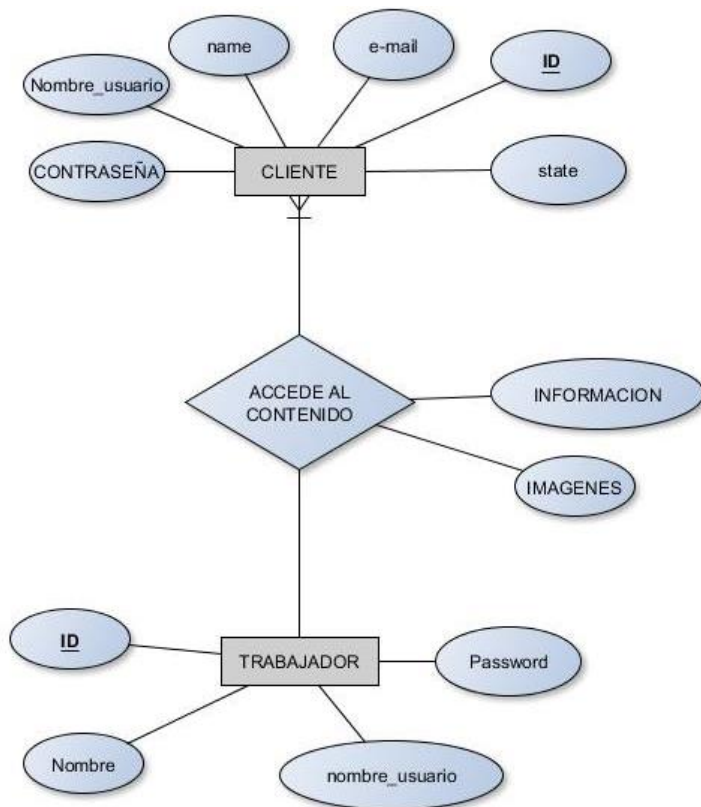


Figura 7: Modelo E/R de una clínica de rehabilitación.

Implementación y resultados.

La plataforma está compuesta de seis partes fundamentales: la cabecera y el cuerpo de la página. A continuación pasemos a explicar cada una de ellas. Mediante una tabla se explicarán los componentes que se utilizan en la cabecera de la plataforma *Gestor Integral en Línea de una Clínica de Rehabilitación*.

#	Elemento	Morfosintaxis	Semántica
1	Logo general	Se ubica en la parte superior izquierda de la página, su fondo es blanco como el de la página y el formato es PNG	Contiene un vínculo al de la página inicial, cuando se pasa el cursor por el espacio y se da clic sobre el regresa inmediatamente a la parte inicial de la página.
2	Menú principal	Con fondo blanco ligeramente degradable en Rojo.	Menú creado por medio de HTML que permite visualizar hipervínculos.
3	Buscador	Cuadro de Texto con contenido informativo sin logo	Función creada en HTML que busca contenido de la página. Cuando el usuario marca la casilla, desaparece el texto predictivo.
4	Carrusel imágenes	Espacio de 416x1170 pixeles que permite reflejar imágenes.	Componente de Script que cambia la imagen del carrusel cuando el usuario quiera apasara a observar la siguiente.

Conclusiones.

Se cumplieron todos los objetivos del proyecto, es decir, se desarrolló una Aplicación Web para la administración de una clínica de rehabilitación fácil de usar, visual, orientada al tipo de usuario de hoy en día acostumbrado a navegar en redes sociales.

Además se logró una aplicación enfocada en las necesidades de nuestros clientes, permitiendo tener un fácil acceso a todas las secciones de nuestra página Web, dando un servicio de alta calidad.

Las pruebas ejecutadas fueron satisfactorias, éstas se realizaron usando la Prueba Basada en Requisitos o Pruebas de Casos de Uso. De esta manera fue posible verificar que la aplicación Web cumpliera al 100% con los requerimientos establecidos.

Capítulo 3: Planeación del proyecto.

3.1 Definición del problema.

En la Secretaría de Finanzas y Administración del estado de Puebla (SFA), existe un área restringida denominada Dirección de Infraestructura y Soporte Técnico (DIST) donde se realizan las siguientes funciones:

- ❖ Arranque y reconfiguración de los servidores de: Base de Datos, aplicación, web.
- ❖ Arranque y reconfiguración del Directorio Activo.
- ❖ Respaldo de máquinas virtuales productivas.
- ❖ Reingeniería y configuración de los Proxy de contingencia.
- ❖ Atención de requerimientos.
- ❖ Impresión de la nómina del Gobierno del Estado de Puebla.
- ❖ Respaldo de cintas.

Entre algunas otras más.

Debido a que la información que se maneja en el DIST es de suma confidencialidad, se requiere poder tener una herramienta electrónica, en la cual se lleve un control de acceso del personal que ingresa.

Dicha herramienta es con el fin de tener bien identificado al personal que accede a esta área, debido a la importancia de la misma.

Existen 2 tipos de personal:

- ❖ **El personal interno de la SFA** (intendencia, administrativos, así como personal de algunas otras áreas asociadas a la DIST).
- ❖ **El personal externo a la SFA** (distribuidores o funcionarios de alguna otra institución o dependencia pública).

Anteriormente en la DIST se contaba con una bitácora manual en la cual a veces los usuarios olvidaban ingresar la información requerida. Dicha bitácora era fácil de manipular información por lo cual no era confiable.

Posteriormente se migro a una bitácora electrónica, pero esta no registra la hora de salida correctamente y no se puede acceder a una consulta o informe de accesos del personal en un determinado tiempo.

3.2 Estrategia de solución.

La estrategia de solución consta de los siguientes pasos:

- ❖ Analizar el problema.
- ❖ Diseñar la base de datos, así como la interfaz del sistema que el cliente solicito.

- ❖ Implementación del sistema.
- ❖ Pruebas
- ❖ Presentación y liberación del sistema.
- ❖ Redacción del documento de tesis.

Se va a crear un sistema web en PHP usando MySQL, el cual permitirá un buen control de la información del personal que accede al DIST.

Donde el sistema permitirá a un administrador hacer consultas sobre las veces que accedió un usuario, así como generar reportes de los usuarios que accedieron.

Cabe recalcar que existen 2 tipos de personal los cuales ya fueron explicados en la sección 3.1.

Si en algún momento hay que hacer alguna aclaración sobre los accesos, esta herramienta será de gran ayuda.

3.3 Adecuación del proyecto en la planeación.

La planeación del proyecto se lleva a cabo implementando un ciclo de vida en espiral (Figura 8) así como los 6 pasos de la estrategia de solución mencionados en la sección 3.2.

Ciclo de vida de espiral¹: Para cada ciclo habrá cuatro actividades:

1. Determinar Objetivos.
2. Análisis del riesgo.
3. Desarrollar y probar.
4. Planificación.

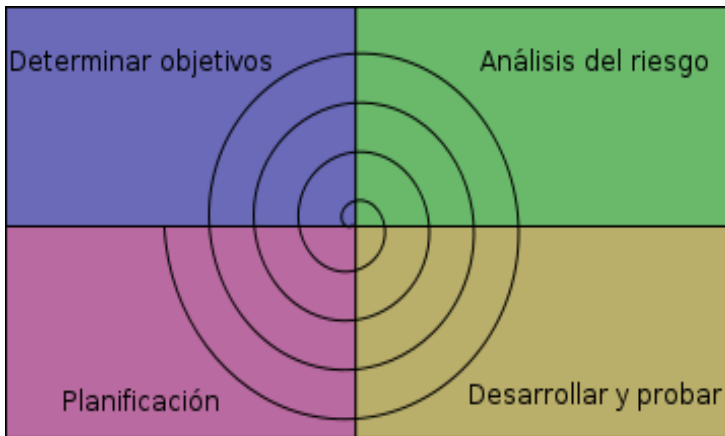


Figura 8: Modelo en espiral de Boehm.

❖ **Etapa de análisis:** Es el proceso de investigar, se define el sistema que se desea crear y se identifican los componentes principales del sistema.

¹ El desarrollo en espiral es definido por primera vez por Barry Boehm en 1986, Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a ninguna prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior.

- ❖ **Etapa de diseño:** Es el proceso de utilizar la información recolectada en la etapa del análisis del sistema para desarrollar un modelo con las especificaciones del sistema.
- ❖ **Etapa de implementación:** Consiste en crear el sistema, para ello se llevará a cabo la codificación de la aplicación web mediante la programación en HTML, PHP, MySQL y CSS.
- ❖ **Etapa de pruebas:** Consiste en asegurar que los componentes que integran el sistema cumplen con los requerimientos de la especificación creada en el diseño, para asegurar el funcionamiento correcto del mismo.
- ❖ **Etapa de presentación:** Consiste en presentarle el proyecto final al cliente, en este caso la presentación se realiza a Alejandro Flores Torres quien es el encargado del centro de datos, para obtener su aprobación.
- ❖ **Redacción de tesis:** Consiste en estructurar el documento de tesis que se presentara como antecedentes de lo que se realizó en este proyecto.

A continuación se muestra la Tabla 1 correspondiente a la agenda de actividades del presente proyecto.

ACTIVIDAD	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6
Análisis y Captura de Requisitos	X					
Diseño del Sistema y Modelado del diagrama entidad relación.	X	X				
Implementación del Sistema y Creación e implementación de la base de datos.			X	X	X	
Pruebas del Sistema					X	
Presentación y liberación del Sistema por Alejandro Flores Torres.						X
Escritura del documento de tesis		X	X	X	X	X

Tabla 1: Agenda de actividades.

3.3.1 Infraestructura.

Para el desarrollo de la aplicación se dispone de un equipo de cómputo con las siguientes características, ver Figura 9.

Id. del producto	00179-40447-19627-AAOEM
Procesador	Intel(R) Core(TM) i3-3217U CPU @ 1.80GHz 1.80 GHz
RAM instalado	4.00 GB (3.88 GB utilizable)
Tipo de sistema	Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil	La entrada táctil o manuscrita no está disponible para esta pantalla

Windows	
Edición	Windows 8.1 Single Language
Activación	Windows está activado

Figura 9: Características de la Infraestructura.

3.3.2 Software.

El software necesario para el desarrollo de la aplicación será:

- ❖ Wamp server 2.5, el cual incluye:
 - Apache 2.4.9
 - MySQL 5.6.17
 - PHP 5.5.12
 - PHPMyAdmin 4.1.14
 - SqlBuddy 1.3.3
 - XDebug 2.2.5
- ❖ Joomla 3.4

Capítulo 4: Análisis de requerimientos.

4.1 Identificación, listado de actores y casos de uso.

Para que un sistema sea de alta calidad, tiene que cumplir las necesidades de los usuarios. Hay que identificar los usuarios del sistema y las tareas que deben acometer con el sistema.

También hay que buscar información sobre que tareas son las más importantes, de manera que se pueda planificar el sistema de acuerdo con esto.

En UML se utilizan como términos técnicos: actores y casos de uso.

Donde un actor es un usuario del sistema que tiene un rol particular.

Y un caso de uso es una tarea que un actor necesita ejecutar con la ayuda del sistema. [1]

Identificación y listado de los actores.

A continuación se realiza la identificación de los actores que interactúan con el sistema; los cuales son los siguientes:

- ❖ Personal interno al DIST: este tipo de personal labora en la SFA y puede pertenecer a cualquier dirección o departamento de la misma. Este personal es identificado por un número de expediente único.
- ❖ Personal externo al DIST: El personal externo, como se ha mencionado anteriormente labora fuera de la

SFA, puede pertenecer a cualquier empresa que brinda sus servicios a esta institución.

- ❖ Administrador del sistema: El administrador del sistema labora en la SFA única y exclusivamente en DIST, ya que será el encargado de manipular la información así como de generar consultas y reportes.

Casos de uso.

Los casos de uso correspondientes para cada uno de los actores listados anteriormente son los siguientes:

Personal interno al DIST.

- ❖ Registro de entrada: Este actor únicamente podrá ingresar la información solicitada en un formulario, los cuales son:
 - Nombre.
 - Apellidos.
 - Tipo de personal.
 - Dirección a la que pertenece.
 - Departamento.
 - Número de expediente.
 - Control de cambios.
 - Folio de control de cambios.
 - Ubicación.
 - Área.
 - Y motivo de la visita.
- ❖ Registro de Salida: Una vez que este tipo de personal, concluye sus actividades correspondientes al motivo

de la visita, este podrá seleccionar su nombre en la lista de visitantes, para proceder a pulsar un botón de salir y así registrar la hora de salida.

Personal externo al DIST.

- ❖ Registro de entrada: El personal externo tendrá que ingresar los datos solicitados en el formulario de registro siguientes campos:
 - Nombre.
 - Apellido.
 - Tipo de personal
 - Empresa en la que labora.
 - Ubicación.
 - Área.
 - Motivo de la visita.

- ❖ Registrar Salida: Una vez que este concluyo sus actividades correspondientes, podrá seleccionar su nombre en la lista de visitas, para proceder a pulsar un botón de salir y así poder generar su registro de salida.

Cabe mencionar que para el personal interno como externo, el sistema asignara la hora y fecha correspondiente tanto de entrada como de salida.

Administrador del sistema.

- ❖ Registro: Se tendrá que registrarse como administrador, proporcionando la siguiente información; nombre completo, email, así como generar un nombre de usuario y contraseña.
- ❖ Autenticarse (previo registro): Para acceder al sistema, el administrador utilizara su nombre de usuario y contraseña establecidos en el registro.
- ❖ Consultar accesos: Con el hecho de ser administrador del sistema, este actor podrá realizar consultas sobre los registros realizados durante un periodo dado.
- ❖ Generar reportes impresos: El administrador podrá generar reportes impresos sobre los accesos correspondientes, tales reportes podrán ser diarios, o mediante una búsqueda personalizada.
- ❖ Cambiar password (previa autenticación): El administrador podrá cambiar su contraseña cuando lo crea conveniente, solo debe ingresar su contraseña actual y la contraseña nueva.
- ❖ Cerrar sesión: Una vez que este concluyo sus actividades correspondientes, para salir del sistema solo tendrá que cerrar sesión.

4.2 Diagramas de casos de usos.

Los diagramas de casos de uso son importantes para visualizar, especificar, y documentar el comportamiento de un elemento. Ellos hacen sistemas, subsistemas, y clases entendibles para presentar una vista exterior de cómo estos elementos pueden ser usados dentro del contexto.

Los diagramas de caso de uso son también importantes para probar sistemas ejecutables a través de ingeniería hacia adelante y para comprender sistemas ejecutables a través de ingeniería inversa.

Un diagrama de casos de uso es un diagrama que muestra un conjunto de casos de uso, actores y sus relaciones.

“un caso de uso capta un contrato que describe el comportamiento del sistema en distintas condiciones en las que el sistema responde a una petición de alguno de sus participantes”. [2]

Los diagramas de casos de uso comúnmente contienen:

- Casos de uso
- Actores
- Dependencias, generalización, y relaciones de asociación.

Se aplican los diagramas de casos de uso para modelar la vista estática de los casos de uso de un sistema. Esta primera vista soporta el comportamiento de un sistema, el

alejamiento de los servicios visibles que el sistema provee en el contexto del medio ambiente.

La notación de dichos diagramas se ilustra en la Figura 10

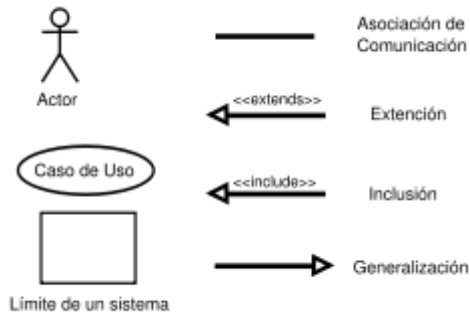


Figura 10: Notación de los casos de uso.

A continuación una breve descripción de la notación UML para el diseño de diagramas de casos de uso.

- Extiende: Cuando un caso de uso especializa a otro extendiendo su funcionalidad.
- Usa: Cuando un caso de uso utiliza a otro.
- Los muñecos representan a los actores.
- Los óvalos representan a los casos de uso.

- Los rectángulos son el límite de un sistema.
- La línea es la asociación entre un actor y un caso de uso.

4.3 Realización de los casos de uso.

Para la gestión de los diagramas UML² se utilizara una herramienta CASE³ para hacerlos más fácilmente.

A continuación se presentan los diagramas de casos de uso para cada actor correspondiente realizados con dicha herramienta CASE.

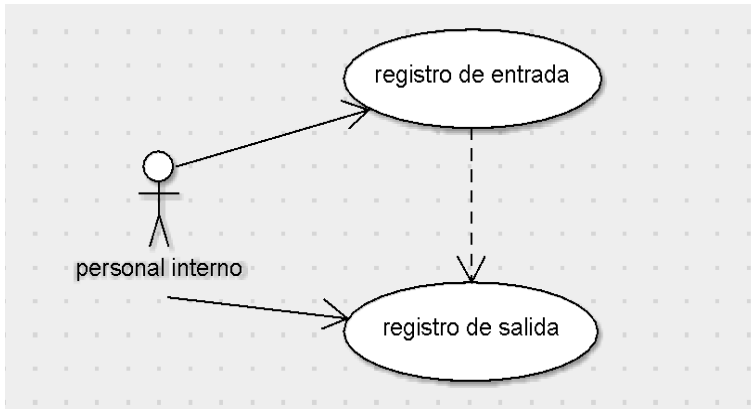


Figura 11: Caso de uso del personal interno.

² Lenguaje unificado de modelado.

³ Para el presente sistema se utiliza la herramienta CASE llamada ArgoUML.

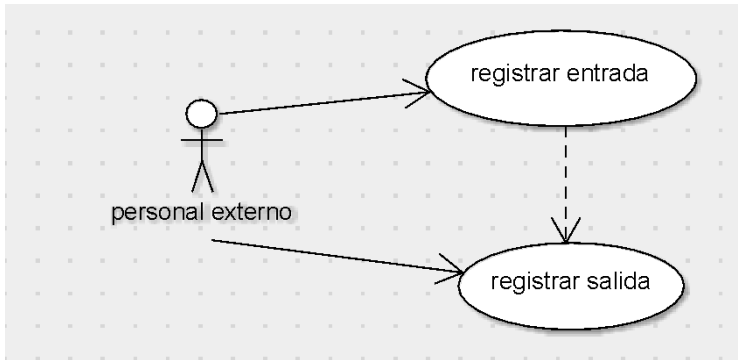


Figura 12: Caso de uso del personal externo.

Como se puede apreciar en la Figura 11 y la Figura 12, el caso de uso para el personal interno como externo es el mismo.

Puesto que no importando el tipo de personal, ambos deben realizar el correspondiente registro de entrada y salida.

La única variación que existirá entre estos 2 tipos de personal será que en el formulario de registro según sea su caso, solo se mostraran los campos diseñados para cada tipo de personal.

La Figura 13 muestra el diagrama de casos de uso correspondiente al administrador del sistema.

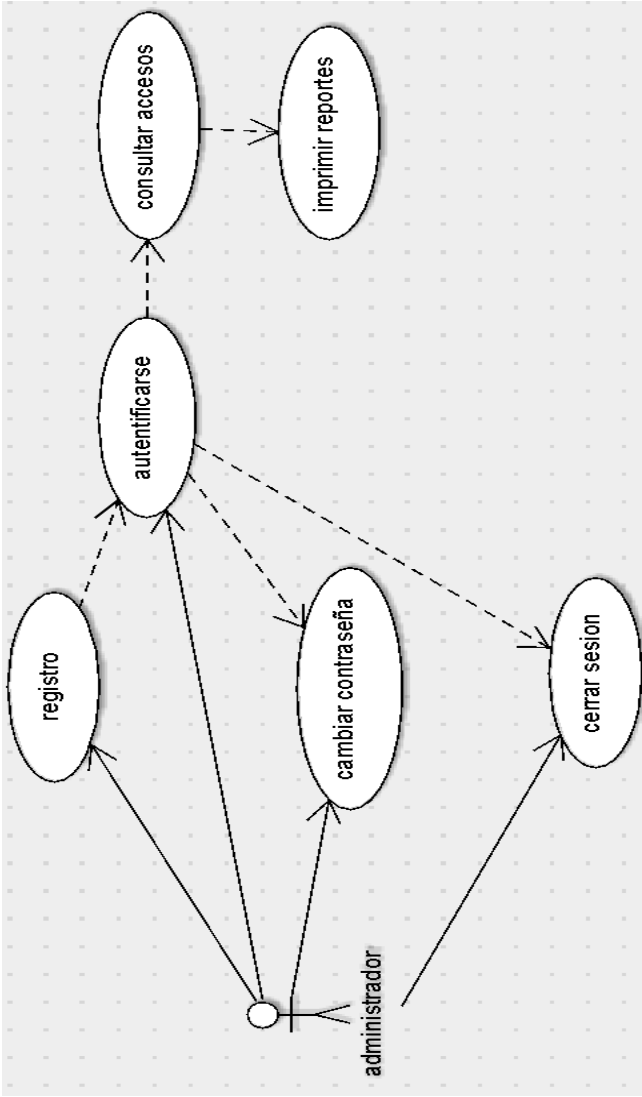


Figura 13: Caso de uso del administrador del sistema.

4.3.1 Secuencias normales y excepciones de los CU.

Durante la ejecución de un caso de uso, suelen aparecer errores o excepciones. Las desviaciones del curso normal del caso de uso se llaman alternativas, las cuales tienen las siguientes características:

- 1) Representan un error o excepción en el curso normal del caso de uso.
- 2) No tienen sentido por sí mismas, fuera del contexto del caso de uso en el que ocurren.

Resulta útil documentar los casos en tablas, mostrando el curso principal en la primera columna, y las alternativas en una segunda columna. [3]

A continuación se presenta mediante tablas las secuencias normales y excepciones de ejecución correspondiente a cada caso de uso.

Cabe recordar que el caso de uso del personal interno y externo es el mismo; puesto que solo necesitan registrar entrada y salida.

Es por ese motivo que a continuación se realiza una sola tabla de secuencias normales y excepciones para ambos casos de uso.

Caso de uso: Registrar entrada.			
Actor: Personal Interno y externo de la SFA.			
Descripción: Mediante el registro realizado se puede tener acceso a la zona restringida denominada SITE.			
Curso Normal:		Curso Alternativo:	
1	El usuario procede a llenar el formulario correspondiente a su tipo de personal.		
2	Al ingresar todos los datos; se procede a enviar la información.	2.1	Se pide al usuario llenar todos los campos requeridos.
3	Se muestra en una tabla de visitas le información que ingreso el usuario.	3.1	El registro no se ha completado.

Tabla 2: Registrar entrada.

Caso de uso: Registrar Salida.			
Actor: Personal Interno y externo de la SFA.			
Descripción: Mediante el registro de salida el sistema registra la hora de salida de dicho personal.			
Curso Normal:		Curso Alternativo:	
1	El usuario procede buscar su nombre y apellido en la lista de visitas.		
2	El usuario selecciona su información correspondiente y pulsa salir.	2.1	Si no encuentra su información en la lista, el usuario no se ha registrado.
3	Se reenvía a la página principal.	3.1	Error de conexión, no es posible acceder al sistema.

Tabla 3: Registrar salida.

Caso de uso: Registro como administrador del sistema.			
Actor: Administrador o encargado del sistema.			
Descripción: Se realiza el correspondiente registro para darse de alta como administrador del sistema y tener acceso a contenido exclusivo.			
Curso Normal:		Curso Alternativo:	
1	El usuario ingresa los datos solicitados en el formulario para crear un nuevo usuario, los cuales son: nombre, usuario, contraseña, confirmar contraseña y email.		
2	Al ingresar todos los datos; se procede a enviar la información.	2.1	Se pide al usuario llenar los campos requeridos: nombre, usuario, email.
3	Se envía el mensaje "El usuario ha sido creado correctamente" y se muestra en una tabla de usuarios el nuevo usuario registrado.	3.1	Se envía el mensaje "El usuario no ha sido creado".

Tabla 4: Registro como administrador.

Caso de uso: Autenticarse.			
Actor: Administrador registrado en el sistema.			
Descripción: Previo registro, el administrador procede a autenticarse para ingresar al contenido exclusivo.			
Curso Normal:		Curso Alternativo:	
1	El usuario accede a la pestaña diseñada para el administrador, donde ingresa: usuario y contraseña.	1.1	Se envía la advertencia, " El usuario y contraseña no coinciden, o usted no tiene una cuenta".
2	Se da la bienvenida y se muestra en las pestañas las opciones diseñadas para el administrador.	2.1	Error de conexión, no es posible acceder al sistema.

Tabla 5: Autenticarse.

Caso de uso: Consultar visitas.			
Actor: Administrador registrado en el sistema.			
Descripción: El administrador accede a realizar las consultas requeridas en el historial de visitas.			
Curso Normal:		Curso Alternativo:	
1	El administrador puede realizar la búsqueda de una visita en específico ingresando el nombre o apellido.	1.1	Se muestra el aviso, "No hay registros que coincidan con la búsqueda".
2	El administrador puede agrupar la información que se muestra en la tabla por: id, ingreso, nombre, apellidos, o motivo de visita.		

Tabla 6: Consultar visitas.

Caso de uso: Imprimir reportes de visitas.			
Actor: Administrador registrado en el sistema.			
Descripción: El administrador puede imprimir un reporte de las visitas conforme el criterio de búsqueda que se requiera.			
Curso Normal:		Curso Alternativo:	
1	El administrador puede presionar el botón de PDF, en donde se puede guardar el archivo con la información o imprimirla; también se puede importar o exportar información de la lista de visitas a un archivo en Excel presionando el botón CSV.	1.1	Se muestra el mensaje "Debe realizar una consulta"

Tabla 7: Imprimir reportes.

Caso de uso: Cambiar contraseña.			
Actor: Administrador registrado en el sistema.			
Descripción: El administrador puede cambiar su contraseña editando la información de su usuario.			
Curso Normal:		Curso Alternativo:	
1	El administrador ingresa al gestor de usuarios, presiona editar para crear y confirmar la nueva contraseña.	1.1	Se muestra el mensaje "Las contraseñas no coinciden"
2	Se envía el mensaje "El usuario ha sido modificado correctamente" y se muestra en una tabla de usuarios el usuario registrado.	2.1	Se envía el mensaje "El usuario no ha sido modificado".

Tabla 8: Cambiar contraseña.

Caso de uso: Cerrar Sesión.			
Actor: Administrador registrado en el sistema.			
Descripción: El administrador después de realizar las actividades requeridas, procede a cerrar sesión para evitar la manipulación de información por parte de los usuarios.			
Curso Normal:		Curso Alternativo:	
1	El administrador ingresa a la pestaña de "administrador" y en el módulo de acceso presiona el botón "DESCONECTAR".	1.1	Se muestra el mensaje "No es posible desconectar"

Tabla 9: Cerrar sesión.

4.3.2 Diagramas de secuencia para cada CU.

UML incluye los diagramas de secuencia como notación que puede representar las interacciones de los actores y las acciones que realiza.

El DSS⁴ muestra gráficamente los eventos que fluyen de los actores al sistema. [4].

A continuación se muestran los diagramas de secuencia correspondientes a los 2 tipos de personal, interno y externo. Así como también el DSS para el administrador.

⁴ Diagrama de Secuencia del Sistema.

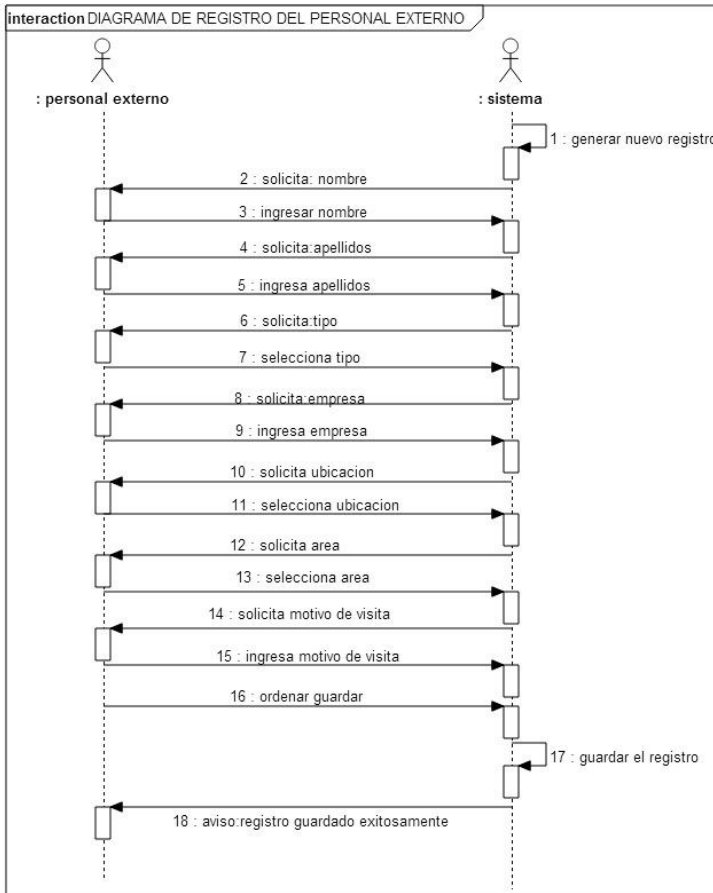


Figura 14: Diagrama de secuencia para el registro de visita del personal externo.

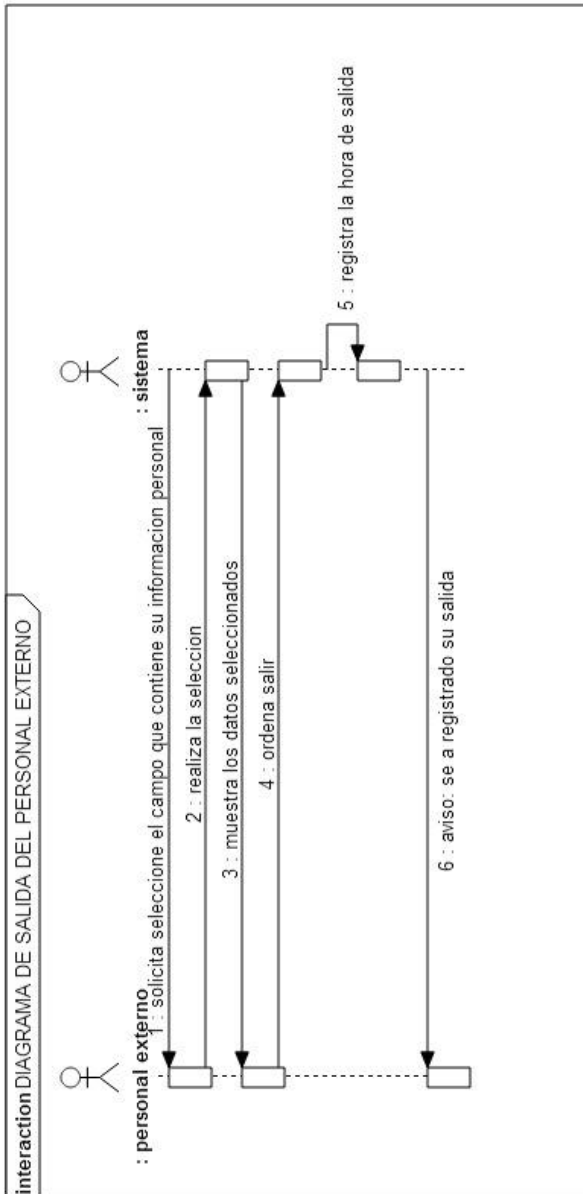


Figura 15: Diagrama de secuencia para la salida del personal externo.

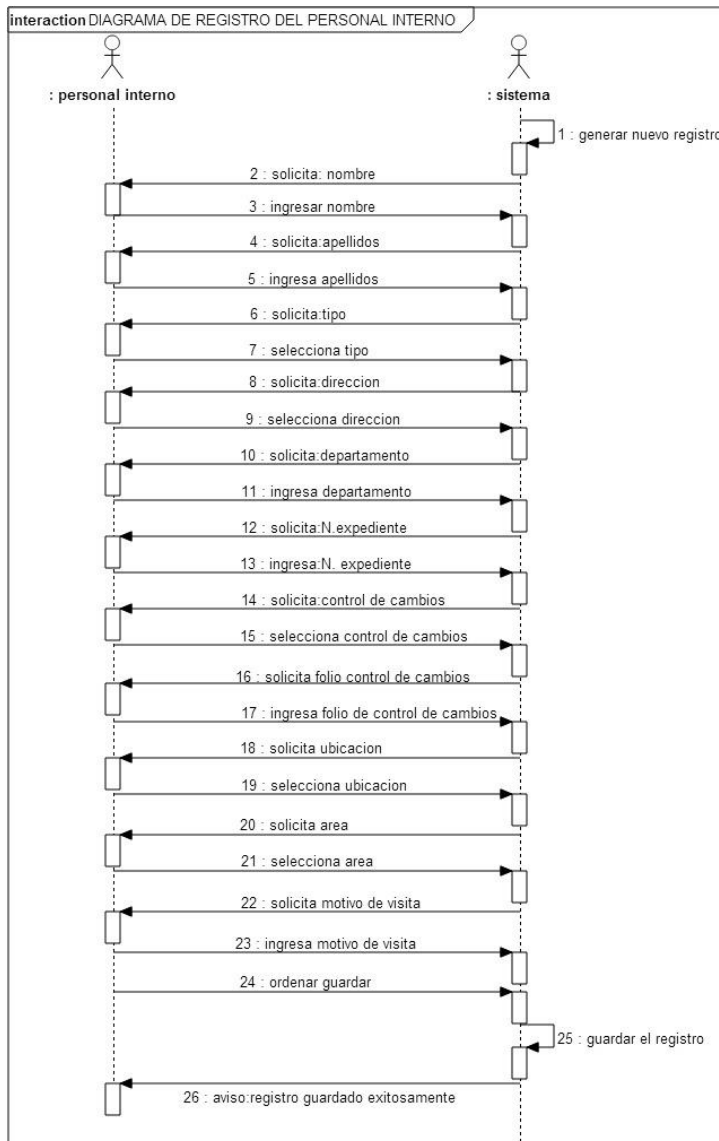


Figura 16: Diagrama de secuencia para el registro de visita del personal interno.

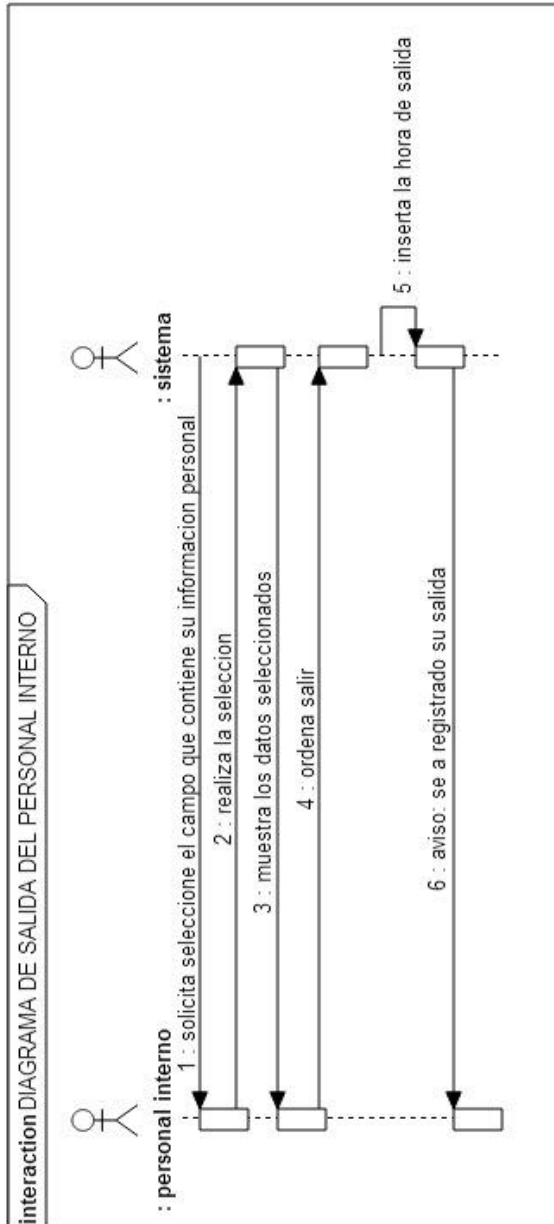


Figura 17: Diagrama de secuencia de salida del personal interno.

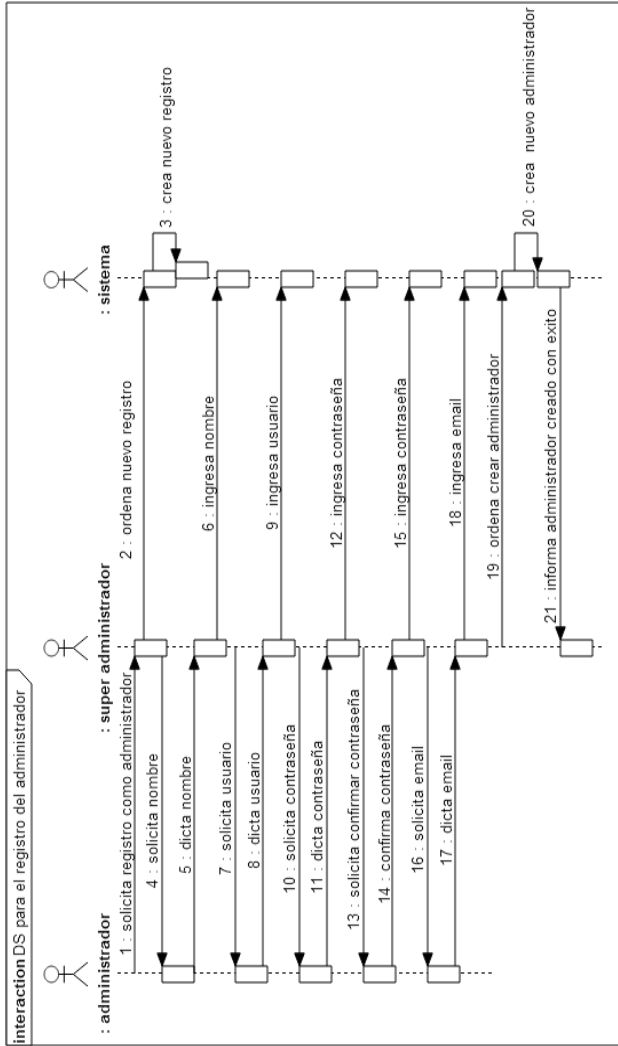


Figura 18: Diagrama de secuencia para el registro del administrador.

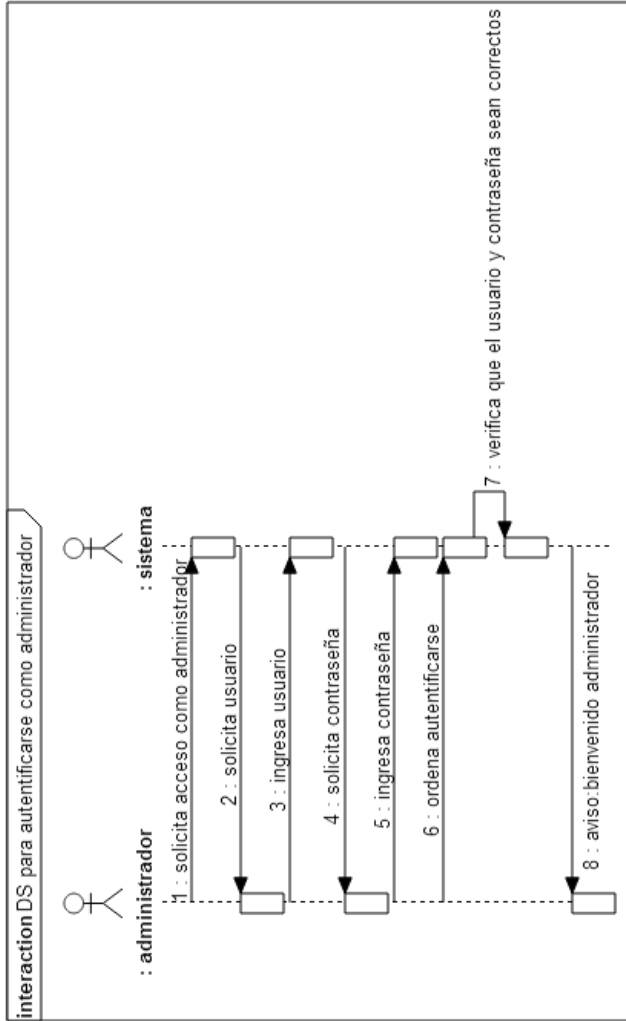


Figura 19: Diagrama de secuencia para autenticarse como administrador.

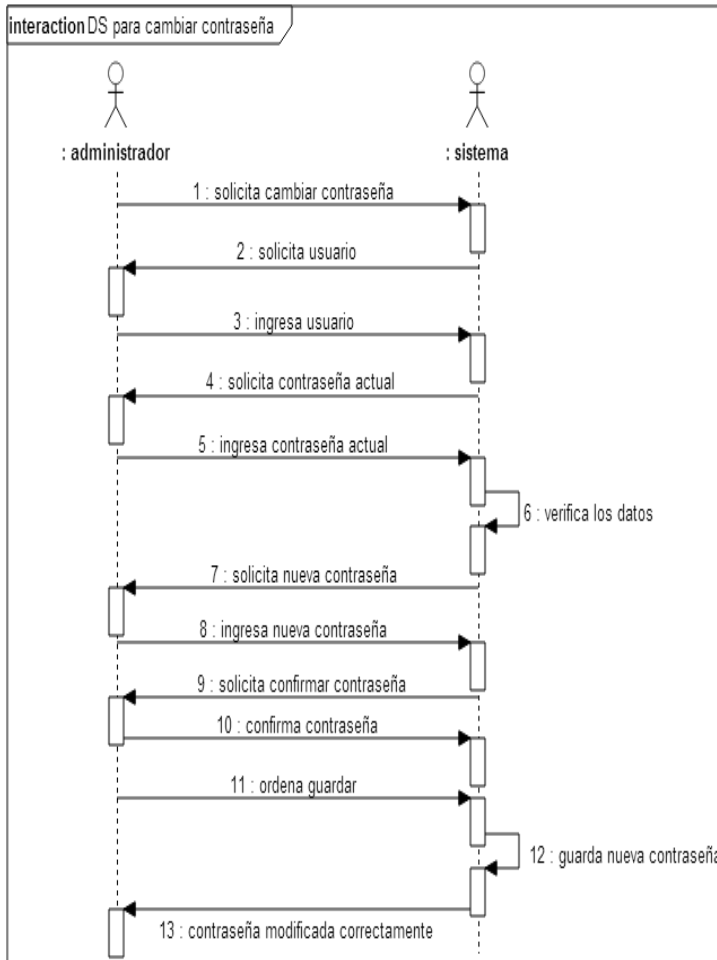


Figura 20: Diagrama de secuencia para cambiar contraseña.

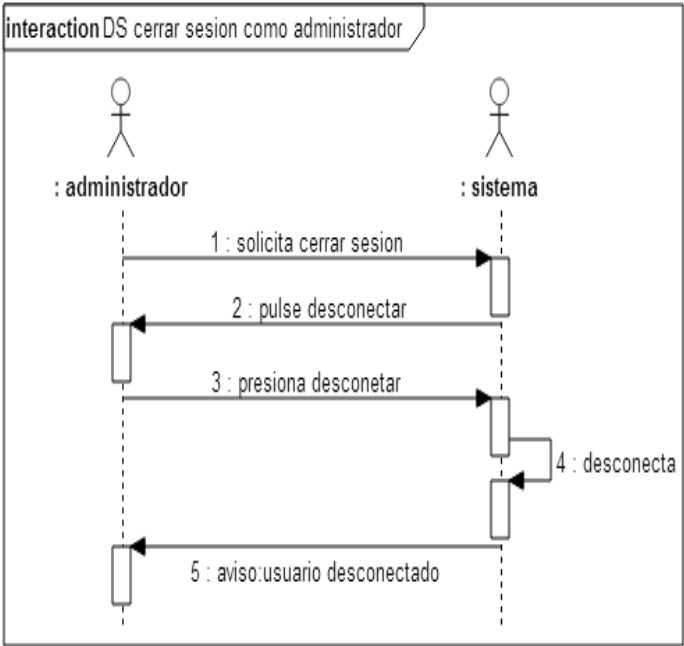


Figura 21: Diagrama de secuencia para cerrar sesión como administrador.

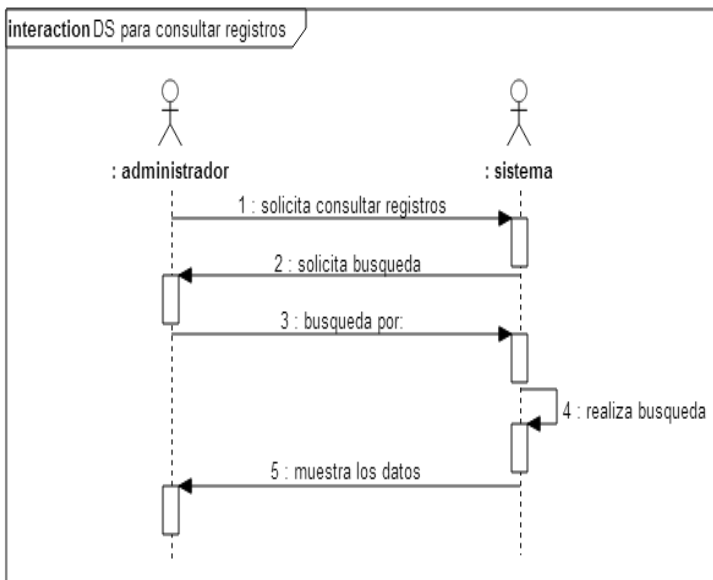


Figura 22: Diagrama de secuencia para consultar registros.

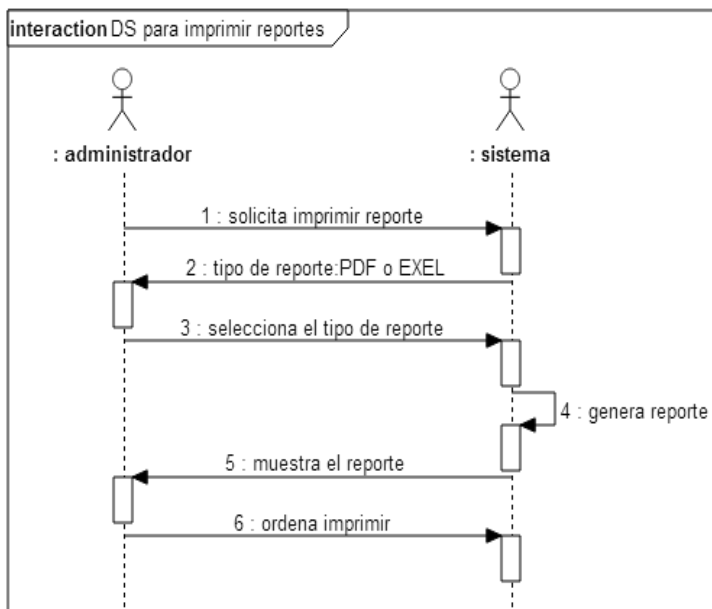


Figura 23: Diagrama de secuencia para imprimir reportes.

Capítulo 5: Diseño.

Para el diseño del proyecto, se propone el uso de la metodología OOHDM⁵.

La metodología identifica tres diferentes niveles de diseño en forma independiente de la implementación, los cuales son:

- ❖ El Conceptual.
- ❖ El Navegacional.
- ❖ Y el de interfaz.

Para ilustrar los diagramas correspondientes para cada uno de los niveles antes mencionados; se implementan con la metodología UWE.

Las imágenes que ilustran los diagramas que se presentan en el presente capítulo se realizaron con el software **MagicDraw** usando el plugin **UWE**⁶.

5.1 Diseño conceptual del proyecto.

El esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas

⁵ Metodología de diseño hipermedia orientado a objetos.

⁶ Metodología de ingeniería de software para el desarrollo de aplicaciones web, basado en UML.

entidades del mundo real. La Figura 24 muestra el diseño conceptual del proyecto.

Las clases son usadas durante el diseño navegacional para derivar nodos, y las relaciones que son usadas para construir enlaces. [5]

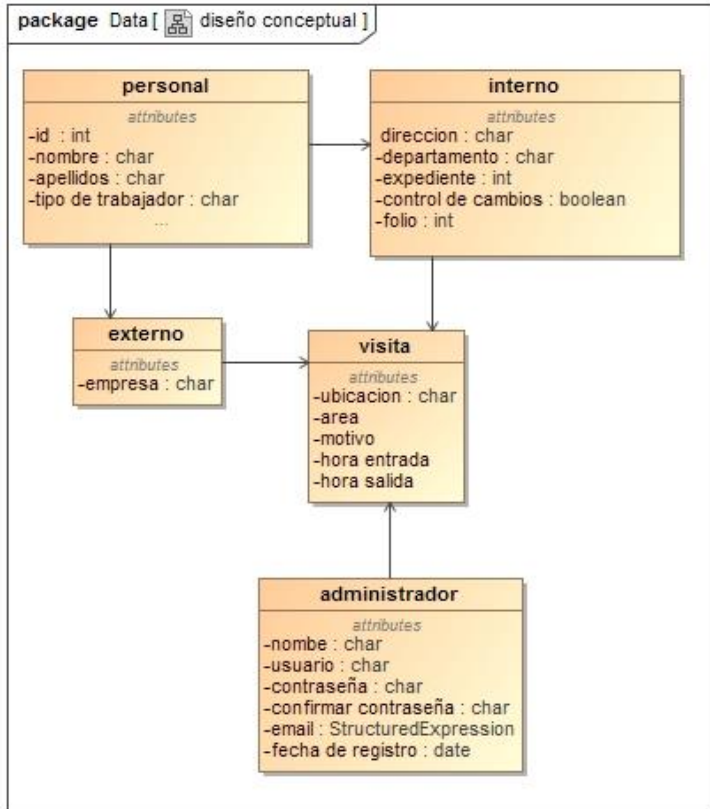


Figura 24: Diseño conceptual del proyecto.

5.1.1 Identificación de las clases u objetos del sistema.

Una clase engloba toda la información de un objeto compartiendo una estructura y comportamiento comunes. Un objeto es, por lo tanto, el “resultado” o instancia de una clase [6].

Una clase está conformada por dos elementos:

- **Atributos:** son los datos que definen al estado del objeto.
- **Métodos:** son las funciones que pueden aplicarse a los objetos.

A continuación se realiza la identificación y el listado de las clases del sistema:

- ❖ **PERSONAL:** Esta clase se utiliza para representar a los usuarios del sistema.

Atributos:

- Nombre
- Apellidos
- Tipo de personal

- ❖ **INTERNO:** Esta clase se utiliza para especificar los datos si el personal es interno.

Atributos:

- Departamento
- Número de expediente

- Control de cambios
- Folio de control de cambios

❖ **EXTERNO:** En esta clase se especifica el nombre de la empresa para la cual trabaja el personal externo.

Atributos:

- Empresa.

❖ **VISITA:** en esta clase el usuario registra su entrada/salida así como datos del área que visita.

Atributos:

- Hora de entrada
- Hora de salida
- Ubicación
- Área
- Motivo de visita

❖ **ADMINISTRADOR:** se almacenan los datos con los que se da de alta un administrador.

Atributos:

- Nombre
- Usuario
- Contraseña
- Confirmar contraseña

- Email
- Fecha de registro

5.2 Diseño de Navegación del proyecto.

Este modelo especifica qué objetos pueden ser visitados por la navegación sobre la aplicación. [5].

En la Figura 25 se muestran los iconos que son utilizados en la metodología UWE para el diseño de navegación del sistema.

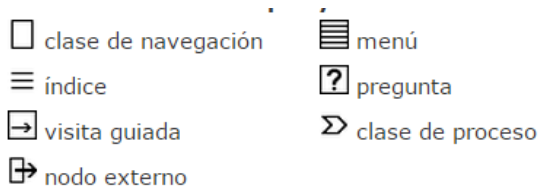


Figura 25: Iconos y estereotipos del diseño de navegación.

En la Figura 26 se ilustra el diseño de navegación del sistema, utilizando los iconos y estereotipos ilustrados anteriormente en la metodología UWE.

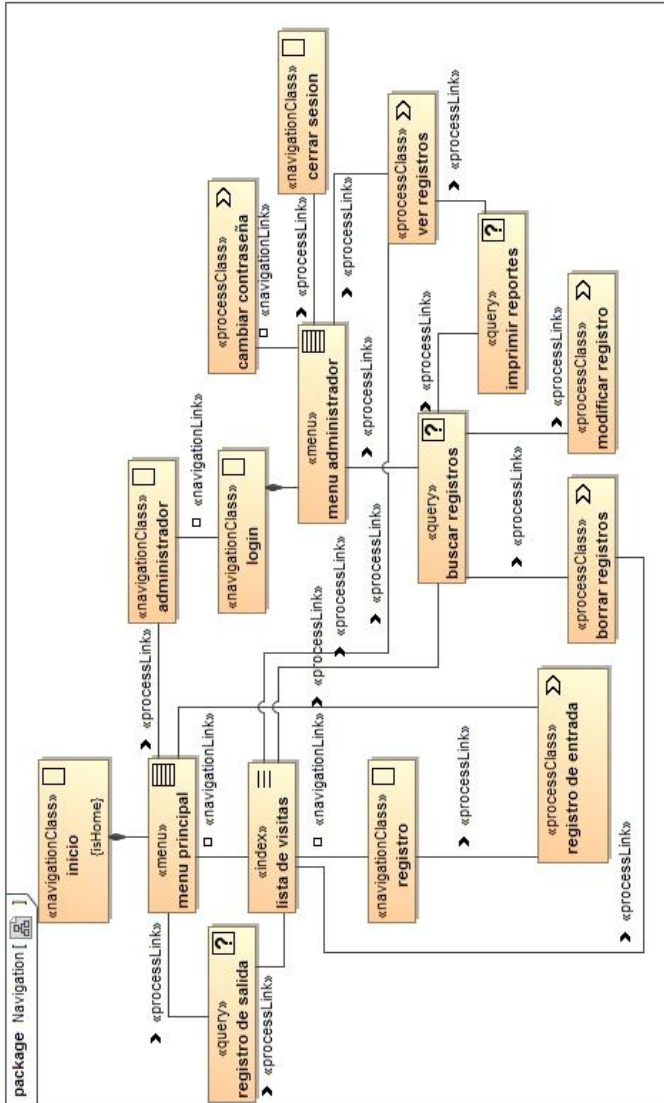


Figura 26: Diseño de navegación del sistema.

5.2.1 Diagramas de clases para modelar la estructura de navegación.

En la Figura 27 se puede observar el diagrama de clases de navegación, el cual está conformado por:

- ❖ **Inicio:** Es la página principal del sistema, está conformado por el objetivo, misión y visión de la empresa.
- ❖ **Registro de entrada:** Se encuentra conformado por un formulario en el cual el usuario registra los datos de su visita.
- ❖ **Registro de salida:** En este, se muestra una lista de las visitas registradas, el usuario realiza la búsqueda de sus datos para proceder a registrar la hora de su salida.
- ❖ **Administrador:** En esta sección el administrador tiene acceso a todos los datos de las visitas, permitiéndole modificar, borrar o crear el contenido de los registros, así como también imprimir reportes y modificar datos personales de su cuenta.

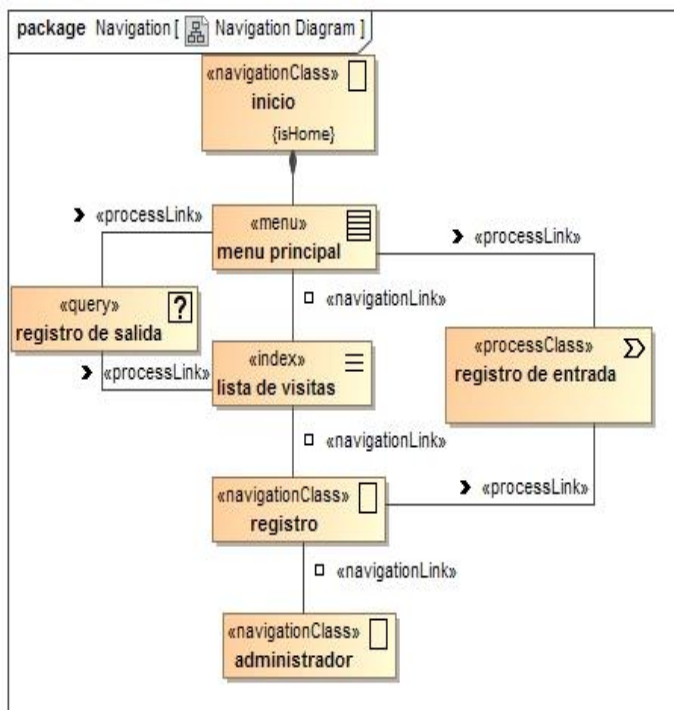


Figura 27: Diagrama de clases de navegación.

5.2.2 Uso de storyboard para modelar la presentación del proyecto.

En la Figura 28 se ilustran los iconos y estereotipos que son utilizados en los diagramas de presentación del sistema.

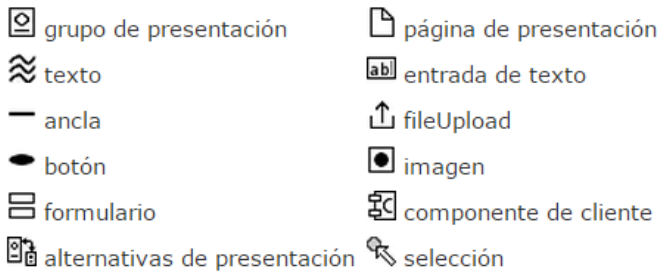


Figura 28: Iconos y estereotipos del diseño de presentación.

En la Figura 29 se muestra el diseño de la página principal, la cual consta de:

- ❖ Logotipo
- ❖ Datos de contacto
- ❖ Menú
- ❖ Texto: objetivo, misión, y visión.

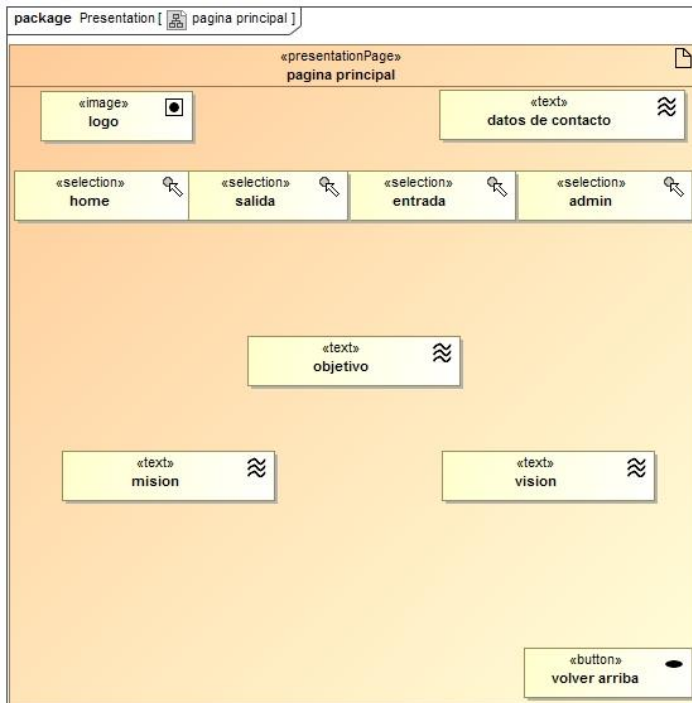


Figura 29: Página principal.

En la Figura 30 se muestra la página para realizar el registro de la visita correspondiente.

Mediante un formulario el usuario introduce la información correspondiente a la visita realizada.

Cuenta con un botón de enviar, para guardar la información en la base de datos del sistema.

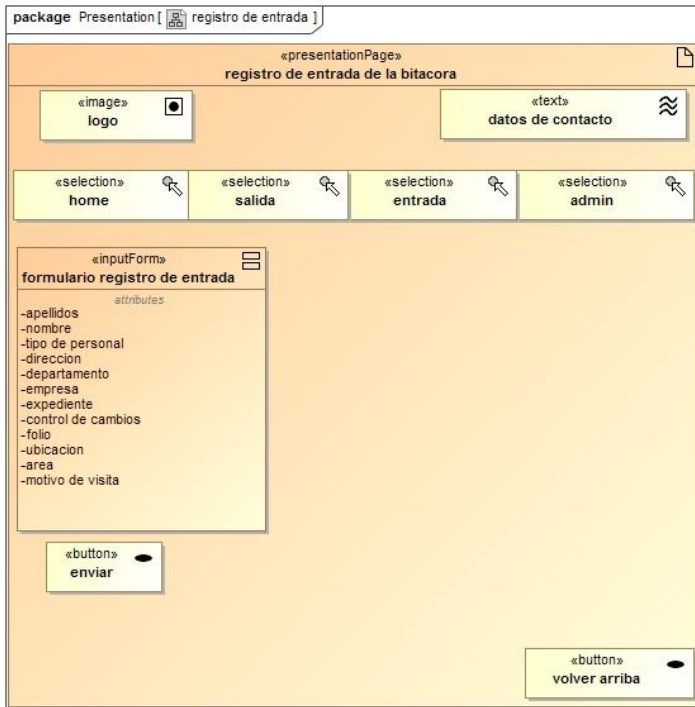


Figura 30: Registro de entrada.

La Figura 31 muestra la página para el registro de salida, esta consta de una lista de visitas registradas en la cual el usuario selecciona la información que corresponde a su visita, para proceder a registrar la salida.

Cuenta con botones para facilitar la búsqueda de información de la visita, los cuales son:

- ❖ Agrupar.
- ❖ Limpiar.
- ❖ Criterio de búsqueda.

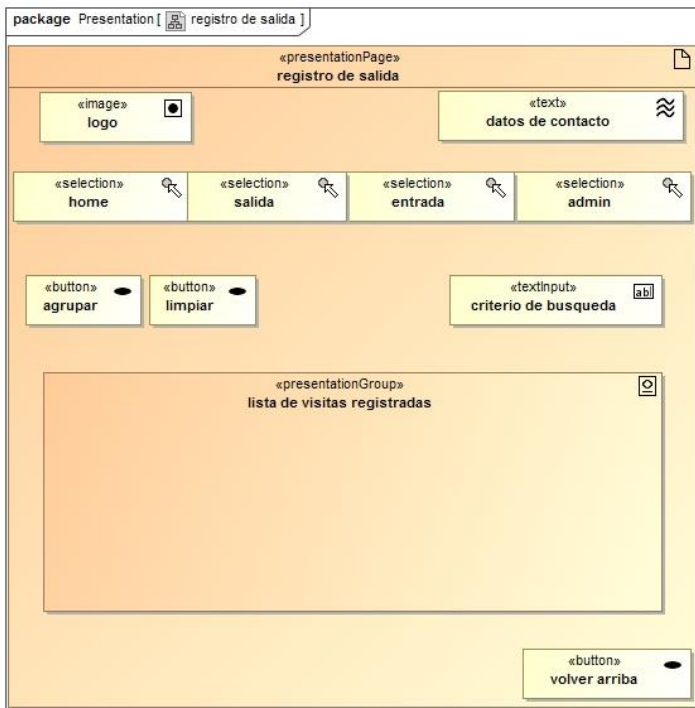


Figura 31: Registro de salida.

En la Figura 32 se presenta la sección dedicada al administrador del sistema, en la cual se presenta un aviso de acceso restringido así como una imagen para hacer alusión al administrador.

Para realizar el acceso consta de un módulo donde el administrador ingresa:

- ❖ Usuario.
- ❖ Contraseña.

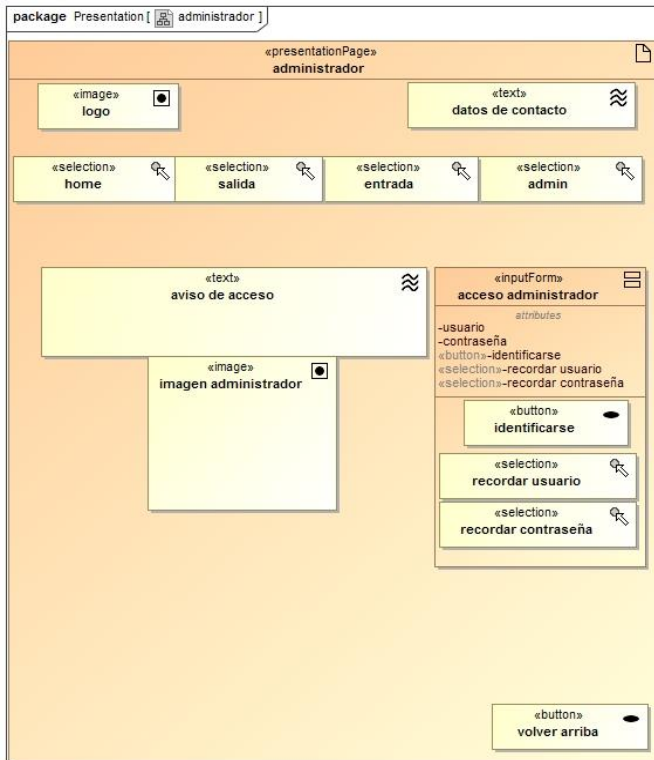


Figura 32: Acceso del administrador.

5.3 Diseño de Escenarios del proyecto.

En esta sección se ilustran algunos escenarios de actividades que utilizan los estereotipos e iconos que se muestran en la Figura 33.

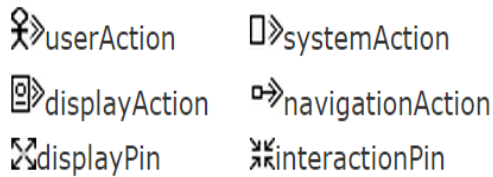


Figura 33: Iconos y estereotipos de los diagramas de actividades.

El estereotipo «user Action» es usado para indicar interacciones de usuario. Por lo contrario, «system Action» describe acciones que son ejecutados por el sistema.

Para modelar grupos de presentación en UWE son usados el estereotipo «display Action», mientras que los dos pines de acción estereotipados «interaction Pin» y «display Pin» son usados para modelar la entrada y la salida de datos.

Finalmente el estereotipo «navigationAction», puede ser usado para modelar opciones de navegación y los elementos asociados de presentación. [7]

En la Figura 34 y la Figura 35; se muestra el diagrama de actividades correspondiente al registro de entrada del personal interno y externo.

En el cual mediante un formulario se solicitan los datos correspondientes.

Una vez que se selecciona el tipo de personal, el formulario muestra los campos correspondientes para realizar el registro de entrada.

Mediante un botón diseñado para enviar los datos, se procede a validar que los campos obligatorios no se encuentren vacíos.

En caso que algunos campos se encuentren vacíos; entonces se envía una alerta para que el usuario ingrese todos los campos obligatorios.

Una vez que la validación es exitosa se guarda la información correspondiente en la base de datos, se envía un aviso de éxito en el registro y se muestra el registro de la visita en una lista.

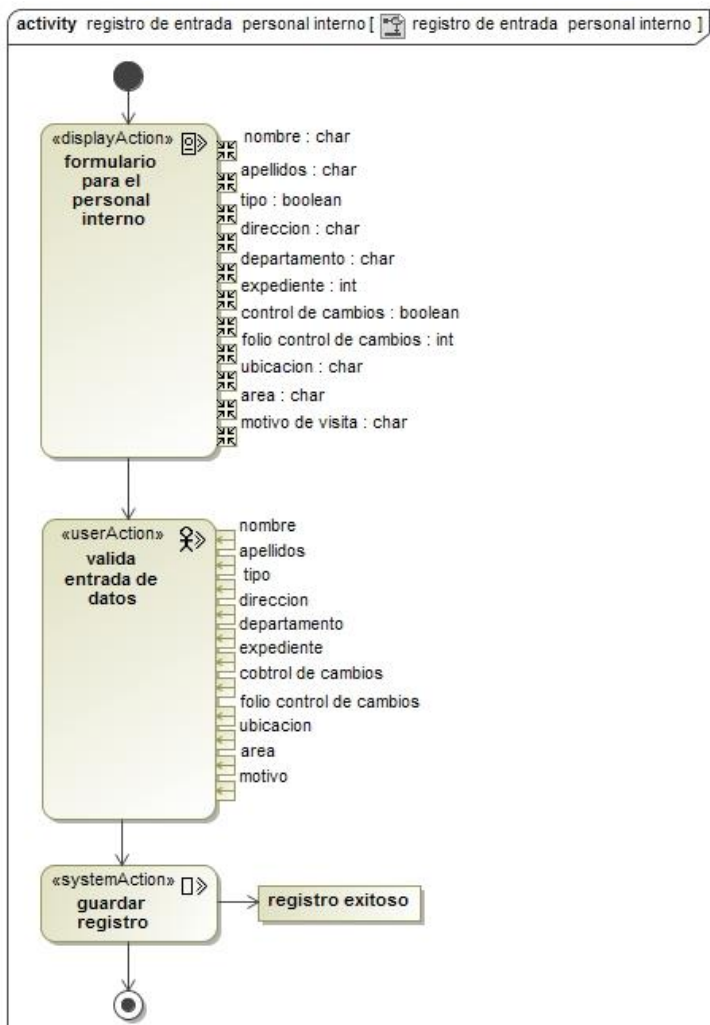


Figura 34: Registro de entrada del personal interno.

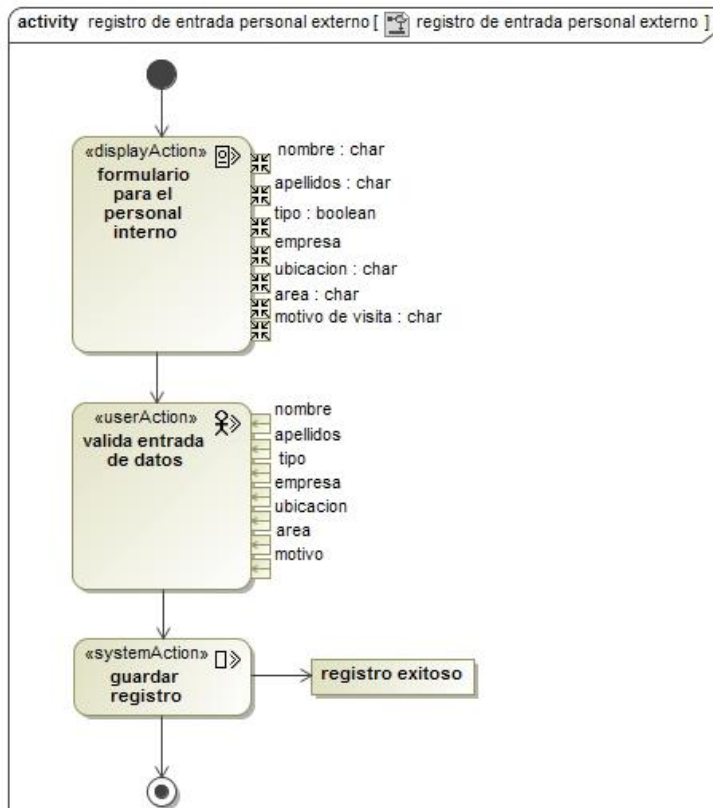


Figura 35: Registro de entrada del personal externo.

En la Figura 36, se muestra el diagrama de actividades correspondiente al egreso del personal interno y externo.

En donde el usuario selecciona de la lista de visitas los datos correspondientes a su información.

Una vez seleccionado, se pulsa el botón para generar la hora de salida emitida por el sistema y guardar los datos.

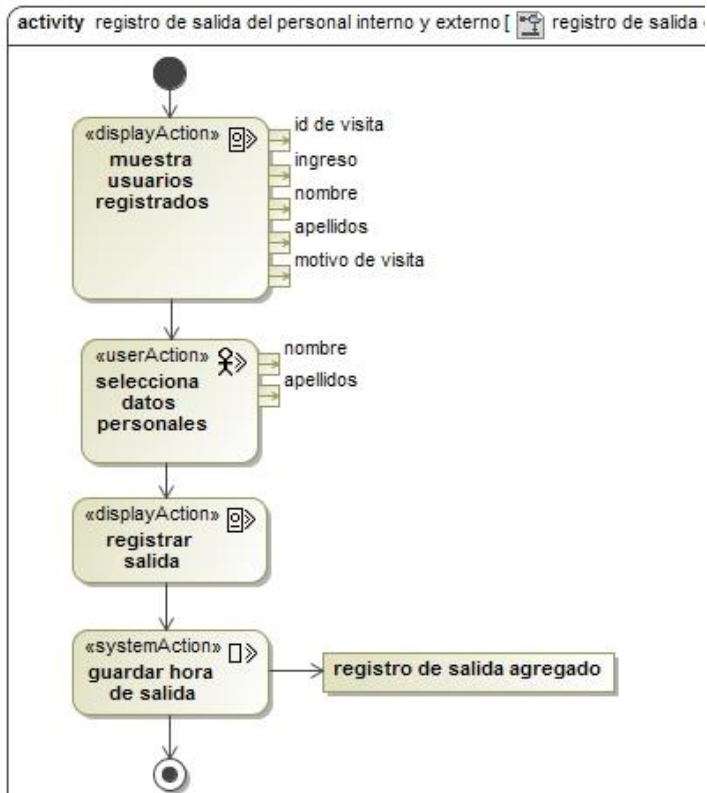


Figura 36: Registro de salida del personal interno y externo.

La Figura 37 muestra el diagrama general de las actividades realizadas por el administrador.

Donde mediante un formulario de acceso se ingresa el usuario y contraseña, una vez que los datos son verificados en la base de datos como válidos; el sistema permite el acceso.

En caso contrario el sistema realiza la redirección al módulo para ingresar nuevamente los datos solicitados o ponerse en contacto con un administrador para generar una nueva cuenta.

Cuando se ingresa al sistema como administrador, este da la opción de realizar búsquedas en la lista de registros de visitas, la búsqueda permitida puede ser por:

- ❖ Ingreso.
- ❖ Apellidos.
- ❖ Nombre.
- ❖ Motivo de visita.

Una vez realizada la búsqueda, al mostrar los datos correspondientes el sistema muestra opciones especiales para:

- ❖ Modificar.
- ❖ Borrar.
- ❖ Imprimir.

Después de que el administrador realice la actividad deseada, puede salir del sistema cerrando sesión.

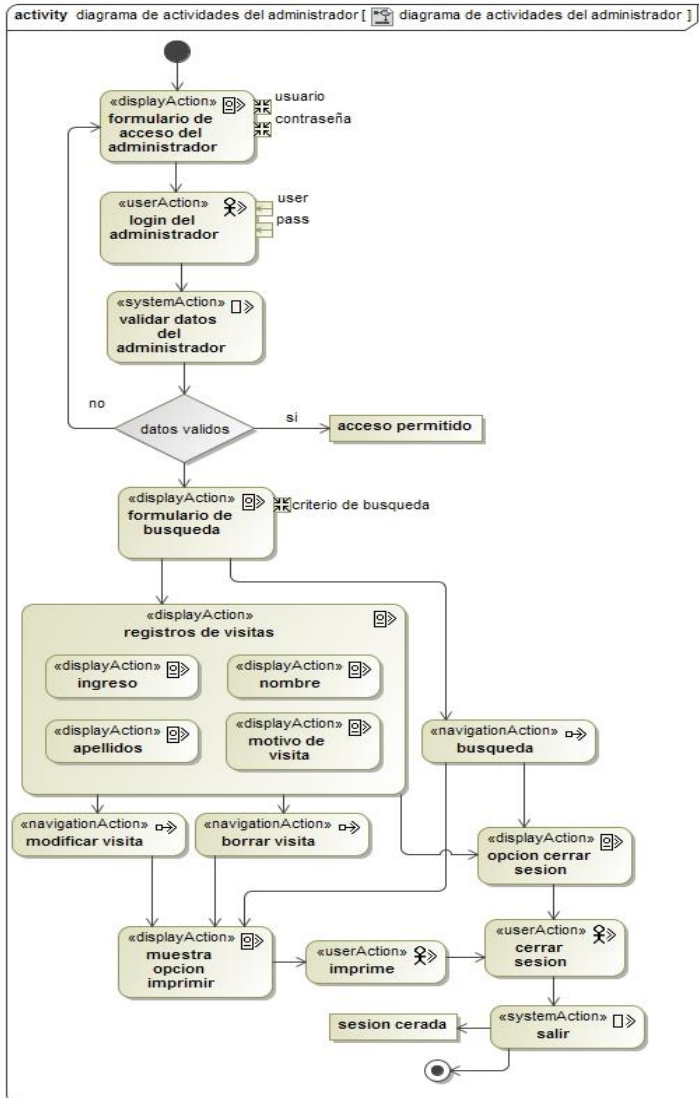


Figura 37: Diagrama de actividades del administrador.

En la Figura 38 se presenta el diagrama de actividades para realizar el login del administrador.

Se muestra un formulario de acceso, en el cual el administrador ingresa usuario y contraseña.

El sistema verifica que los datos se encuentren registrados, y si son válidos; es permitido el acceso.

En caso contrario se solicita verifique los datos y si no tiene una cuenta debe ponerse en contacto con un administrador del sistema.

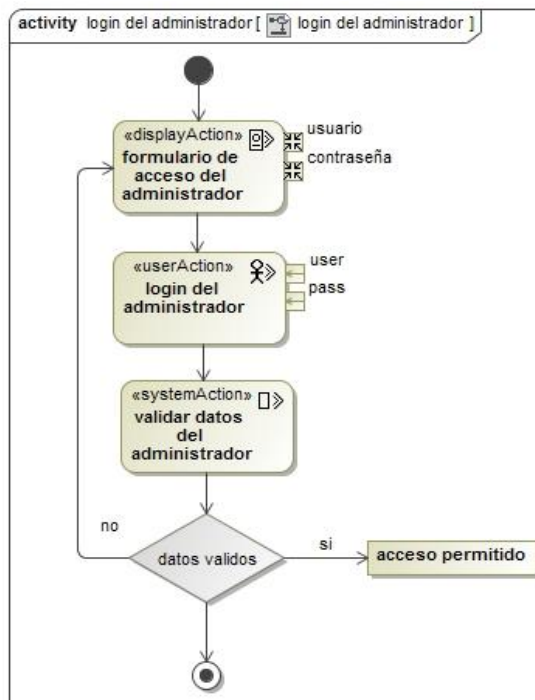


Figura 38: Acceso del administrador.

Para realizar el cambio de contraseña de una cuenta de administrador, primero debe realizarse el acceso como administrador.

El procedimiento a seguir ya fue explicado anteriormente en la Figura 38, así que partiremos desde ese punto.

Una vez que ya el sistema permitió el acceso como administrador, nos muestra un submenú con las opciones para el administrador.

Para cambiar la contraseña se muestra un formulario en el que se ingresan los siguientes datos:

- ❖ Usuario.
- ❖ Contraseña actual.
- ❖ Nueva contraseña.

Se procede a guardar los cambios y el sistema sustituye en la base de datos la contraseña actual por la nueva contraseña ingresada.

La Figura 39 muestra el diagrama de actividades correspondiente para cambiar la contraseña del administrador descrita anteriormente.

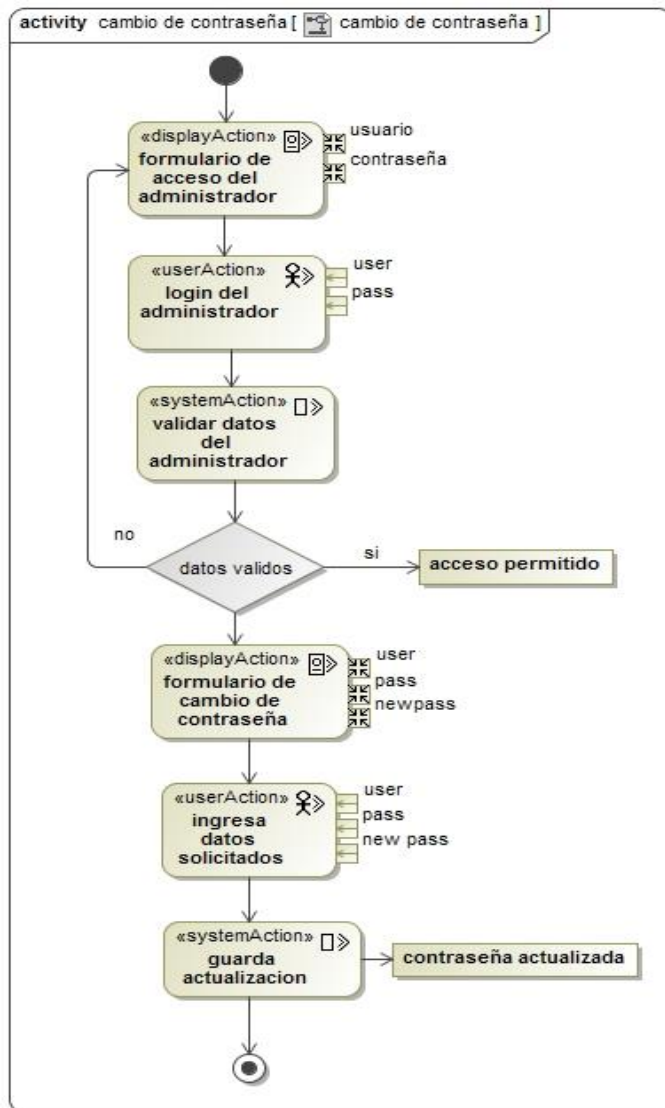


Figura 39: Cambio de contraseña del administrador.

Capítulo 6: Implementación.

6.1 Sistema Cliente - servidor.

“El modelo cliente-servidor es un modelo de computación, en el que el procesamiento requerido para ejecutar una aplicación o conjunto de aplicaciones relacionadas se divide entre dos o más procesos que cooperan entre sí.” [8]

La mayoría de los sistemas cliente-servidor actuales se basan en redes locales.

El servidor es encargado de proporcionar un servicio al cliente y devolver los resultados. El cliente usualmente se ocupa de la interacción de manera gráfica con los usuarios.

En la Figura 40 se ilustra de manera general el modelo del sistema cliente-servidor.

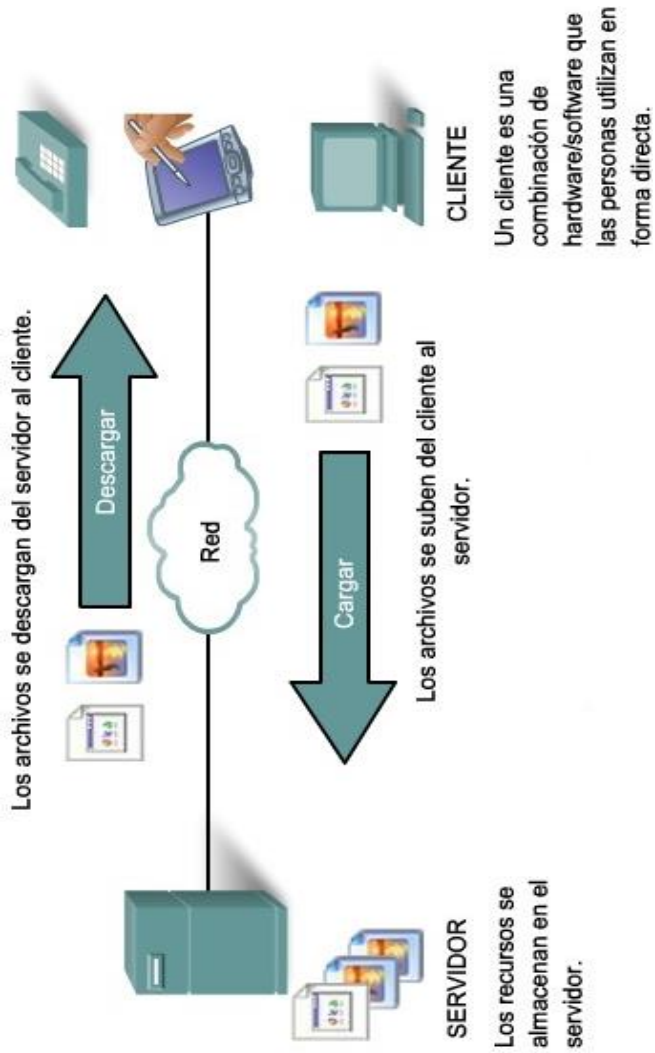


Figura 40: Modelo cliente-servidor.

6.1.1 Implementación y pruebas del sistema.

En esta sección se explica a fondo los pasos que debe seguir el usuario y el administrador del sistema para hacer un buen uso del sistema y poder alcanzar los objetivos principales que se plantearon para dicha aplicación.

Para ello, es necesario que el usuario cuente con los conocimientos básicos de navegación web.

Y en el caso del administrador, se requiere que cuente con conocimientos básicos de base de datos.

En la Figura 41, se ilustra un formulario para la creación de una cuenta como administrador del sistema.

Cabe mencionar que solo el súper administrador y los administradores registrados en el sistema pueden realizar tal operación, ya que por motivos de seguridad no es posible que un usuario se registre como administrador si no se encuentra autorizado.

Para darse de alta como administrador, únicamente se requieren los siguientes datos:

- ❖ Nombre.
- ❖ Usuario.
- ❖ Contraseña.
- ❖ Email.

La fecha de registro la asigna el sistema automáticamente al enviar los datos.

The image shows a web interface for user management. At the top, a dark blue header contains the text 'Gestor de usuarios: Añadir un nuevo usuario' with a user icon. Below this is a light gray navigation bar with three buttons: 'Guardar' (green), 'Guardar y cerrar' (white with a green checkmark), and 'Cancelar' (white with a red 'x'). The main content area is divided into two tabs: 'Detalles de la cuenta' (selected) and 'Configuración básica'. Under 'Detalles de la cuenta', there are several input fields: 'Nombre *', 'Usuario *', 'Contraseña', 'Confirmar contraseña', 'Correo electrónico *', and 'Fecha de registro'. The 'Fecha de registro' field is a grayed-out date picker. The 'Grupos de usuario asignados' section is currently empty.

Figura 41: Creación de la cuenta del administrador.

Una vez que se realizó la captura de los datos solicitados para crear una cuenta; se procede a seleccionar el tipo de usuario que será, esto con la finalidad de otorgarle permisos correspondientes al tipo de cuenta que deseamos crear.

En la Figura 42 se muestra que el tipo de usuario será administrador.

Cuando ya se han realizado los pasos anteriores; en la Figura 43 se puede observar que se ha creado un nuevo administrador registrado como:

- ❖ Nombre: administrador
- ❖ Usuario: admin
- ❖ Email: dirección.infraestructura.site@gmail.com
- ❖ Fecha de registro: 2015-03-20 13:36:55

En la Figura 44 se muestra la sección dedicada al login del administrador, en la cual únicamente se solicitan los siguientes datos:

- ❖ Usuario.
- ❖ Contraseña.

Una vez que se capturo el respectivo usuario y contraseña; el sistema verifica que la información ingresada se encuentre almacenada en la base de datos del sistema.



Figura 42: Selección del tipo de usuario.

Sistema - Usuarios - Menús - Contenido - Componentes - Extensiones - Ayuda - **dist**

Gestor de usuarios: Usuarios

[Nuevo](#) [Editar](#) [Activar](#) [Bloquear](#) [Destroquear](#) [Borrar](#) [Lote](#) [Ayuda](#) [Opciones](#)

Barra lateral

- Usuarios
- Grupos de usuario
- Niveles de acceso
- Notas del usuario
- Categorías de notas

Aviso
Could not instantiate mail function.

Mensaje
El usuario ha sido guardado correctamente.

Buscar

<input type="checkbox"/>	Nombre	Usuario	Habilitado	Activado	Grupos	Correo electrónico	Fecha de la última visita	Fecha de registro	ID
<input type="checkbox"/>	administrador Añadir nota	admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Administrator	direccion.infraestructura.site@gmail.com	2015-03-24 12:43:09	2015-03-20 13:36:55	601
<input type="checkbox"/>	daniel Añadir nota	cronoss	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Administrator	arandast19@hotmail.com	Nunca	2015-04-20 13:42:13	602
<input type="checkbox"/>	Super User Añadir nota	patydguez	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Super Users	patydguez.m@gmail.com	2015-04-20 13:26:03	2015-03-06 14:24:50	600

Ordenar tabla por:

Figura 43: Nuevo administrador registrado.



Inicio. Registro de Entrada. Registro de Salida. **Administrador.**

Contacto.
11 Oriente 2224, Col. Azcárate 72501 Puebla, Pue.
Tel: (222) 229 70 00 Ext. 1014, 1051, 3006, 7092.

Acceso.

Recuérdeme **Identifícase**
 ¿Recordar usuario?
 ¿Recordar contraseña?

Administrador.

Este espacio esta diseñado única y exclusivamente para personal autorizado, si no estas registrado favor de ponerte en contacto con un administrador.



Figura 44: Login del administrador.

En la Figura 45 se puede observar que el acceso es negado, y se muestra una advertencia la cual indica: “el usuario y contraseña no coincide o usted aún no tiene una cuenta”.

Y la Figura 46 muestra en el módulo de acceso el mensaje “Hola Administrador” así como un botón para desconectar o cerrar sesión.

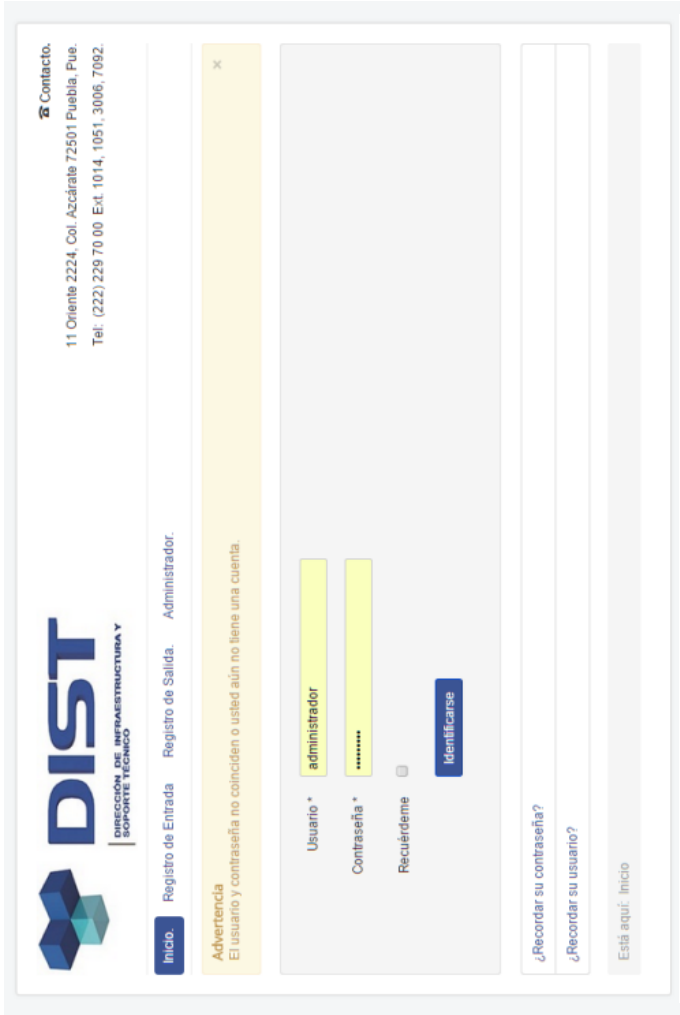


Figura 45: Acceso negado.

 **DIST**
INSTITUCIÓN DE INFRAESTRUCTURA Y
SOPORTE TÉCNICO

Inicio Registro de Entrada Registro de Salida **Administrador**

Contacto.
11 Oriente 2224, Col. Azcañal 72501 Puebla, Pue.
Tel: (222) 229 70 00 Ext. 1014, 1051, 3006, 7092.

Administrador.
Este espacio esta diseñado única y exclusivamente para personal autorizado si no estas registrado favor de ponerse en contacto con un administrador.




Acceso.
Hola administrador
Desconectar

Figura 46: Acceso permitido.


Ya que tenemos acceso al sistema como administrador; este nos permite realizar el cambio de contraseña de nuestra cuenta capturando los siguientes campos requeridos:


- ❖ Nombre.
- ❖ Usuario.
- ❖ Email.
- ❖ Nueva contraseña.
- ❖ Confirmar nueva contraseña.


En la Figura 47 se puede apreciar cómo se realiza dicha actualización y el mensaje de éxito.




Gestor de usuarios: Editar usuario

 Guardar

 Guardar y cerrar

 Guardar y nuevo

 Cerrar

Mensaje
El usuario ha sido guardado correctamente.

administrador

Detalles de la cuenta
Grupos de usuario asignados
Configuración básica

Nombre *	<input type="text" value="administrador"/>
Usuario *	<input type="text" value="admin"/>
Contraseña	<input type="password" value="*****"/>
Confirmar contraseña	<input type="password" value="*****"/>
Correo electrónico *	<input type="text" value="direccion.infraestructura.site@gme"/>

Figura 47: Cambio de contraseña.

En las Figura 48 se observa que al ingresar como administrador el sistema muestra la opción para editar el texto mostrado.

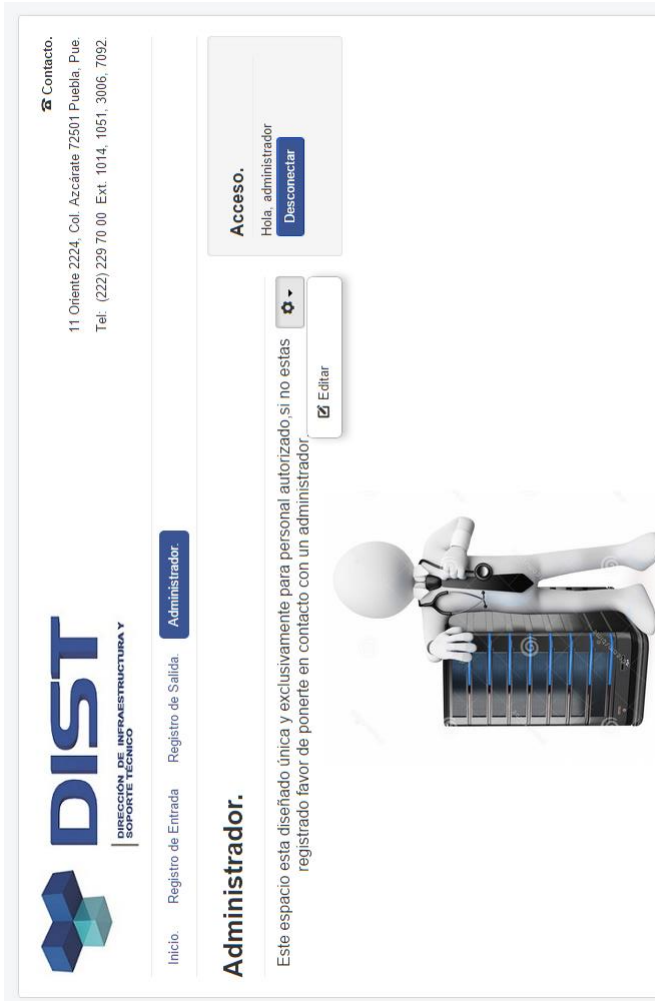



Figura 48: Opción del administrador editar texto.

Para facilitar la búsqueda en los registros; estos pueden ser agrupados. La Figura 49 muestra los diferentes tipos en los que se puede realizar.



**DIRECCIÓN DE INFRAESTRUCTURA Y
SOPORTE TÉCNICO**

11 Oriente 2224, Col. Azcaárate 72501 Puebla, Pue.
Tel: (222) 229 70 00 Ext. 1014, 1051, 3006, 7092.

Contacto.

Inicio
Registro de Entrada
Registro de Salida
Administrador

visitas

⌵ Agrupar por
🔄 Limpiar
⬇ CSV

Buscar

id	Nombre:	Apellidos:	Motivo de Visita:	
	oscar	dominguez	mantenimiento a servidores	▶
	jazmin	sanchez	administración	▶
	daniel	martinez	mantenimiento	▶
	fernando	huesca	personal	▶
9	05-08-2015 19:28	margarita	ventas	▶
10	05-08-2015 19:29	lizbeth	instalación	▶


Ninguno
id
Ingreso:
Nombre:
Apellidos:
Motivo de Visita:

Mostrar # 10

Está aquí: Inicio ▶ Registro de Salida

Figura 49: Opción del administrador agrupar.

La Figura 50 muestra las opciones para modificar, ver y borrar un registro de la lista de visitas.



DIST
DIRECCIÓN DE INFRAESTRUCTURA Y
SOPORTE TÉCNICO

11 Oriente 2224, Col. Azcárate 72501 Puebla, Pue
Tel: (222) 229 70 00 Ext: 1014, 1051, 3006, 7092.

Contacto.

Inicio
Registro de Entrada
Registro de Salida
Administrador.

visitas

Agrupar por Limpiar CSV
Buscar

id	Ingreso:	Nombre:	Apellidos:	Motivo de Visita:		
5	05-08-2015 19:15	oscar	dominguez	mantenimiento a servidores	<input type="checkbox"/>	
6	05-08-2015 19:23	jazmin	sanchez	administracion	<input type="checkbox"/>	
7	05-08-2015 19:25	daniel	martinez	mantenimiento	<input type="checkbox"/>	
8	05-08-2015 19:26	fernando	huesca	personal	<input type="checkbox"/>	
9	05-08-2015 19:28	margarita	fuentes	ventas	<input type="checkbox"/>	
10	05-08-2015 19:29	lizbeth	esteban	instalacion	<input type="checkbox"/>	


Mostrar # 10

Modificar
 Ver
 Borrar

Está aquí: Inicio Registro de Salida.

Figura 50 : Opción del administrador modificar, ver y borrar.

Los datos de los registros tambien pueden ser importados o exportados como se muestra en la Figura 51.



DIST
DIRECCIÓN DE INFRAESTRUCTURA Y
SOPORTE TÉCNICO

11 Oriente 2224, Col. Azcárate 72501 Puebla, Pue.
Tel: (222) 229 70 00 Ext. 1014, 1051, 3006, 7092.

Contacto.

Inicio. Registro de Entrada **Registro de Salida** Administrador.

Limpiar **CSV**

Buscar

visitas

Agrupar por

id	Ingreso:	Elitidos:	Motivo de Visita:
5	05-08-2015 19:15	luisfernando luisfernando	mantenimiento a servidores
6	05-08-2015 19:23	sanchez	administración
7	05-08-2015 19:25	martinez	mantenimiento
8	05-08-2015 19:26	fernando	personal
9	05-08-2015 19:28	margarita	ventas
10	05-08-2015 19:29	esteban	instalacion

Mostrar # 10

Está aquí: Inicio > Registro de Salida

Figura 51: Opción del administrador importar /exportar.

En la Figura 52 se muestra la pagina principal del sistema, en la cual se incluye el objetivo, mision y vision de la DIST.

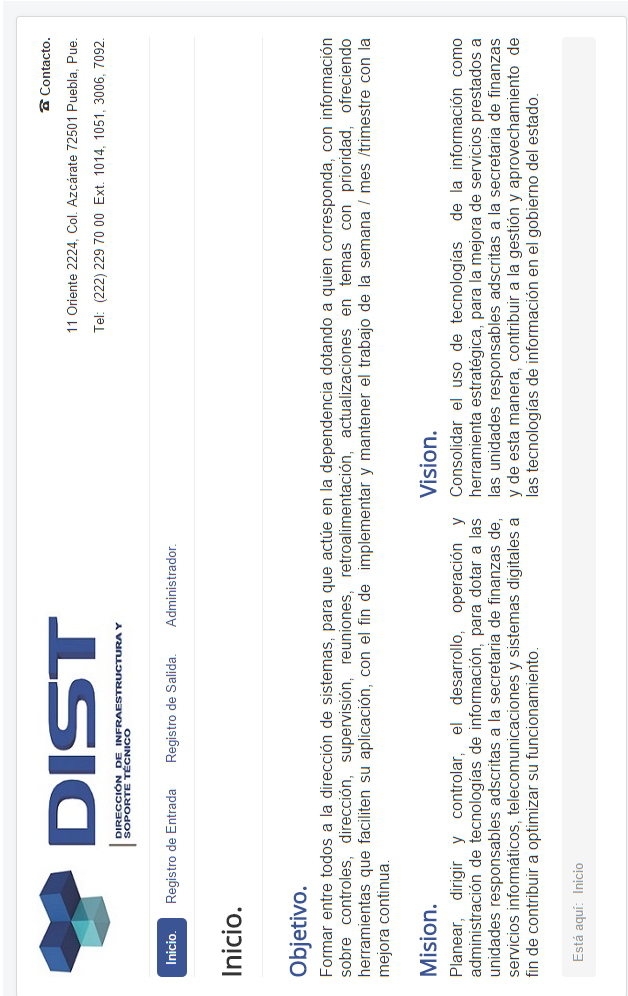


Figura 52 : Página principal del sistema.

En la Figura 53 se muestran los campos que contiene el formulario de registro; al capturar la información solicitada se valida que los campos requeridos no estén vacíos; para después almacenar en la base de datos del sistema.


Inicio. **Registro de Entrada** Registro de Salida. Administrador.

visitas

Nombre:
 Apellidos:
 Tipo de personal: Seleccione: ▼
 Dirección: Seleccione: ▼
 Departamento:
 Empresa:
 Expediente N°:
 Control de cambios: Seleccione: ▼
 Folio de control de cambios:
 Ubicación: Seleccione: ▼
 Area: Seleccione: ▼
 Motivo de Visita:

Figura 53: Formulario de registro de entrada.

En la pestaña "registro de salida" se muestra una lista con los usuarios que han registrado su visita; se selecciona el correspondiente y se procede a registrar la hora de salida. Ver Figura 54.



Contacto.
11 Oriente 2224, Col. Azcárate 72501 Puebla, Pue.
Tel: (222) 229 70 00 Ext. 1014, 1051, 3006, 7092.

Inicio Registro de Entrada **Registro de Salida** Administrador.

visitas

☰ Agrupar por 🔄 Limpiar Buscar

id	Ingreso:	Nombre:	Apellidos:	Motivo de Visita:	Salida:
5	05-08-2015 19:15	oscar	dominguez	mantenimiento a servidores	09-08-2015 21:42
6	05-08-2015 19:23	jazmin	sanchez	administracion	
7	05-08-2015 19:25	daniel	martinez	mantenimiento	
8	05-08-2015 19:26	fernando	huesca	personal	09-08-2015 21:39
9	05-08-2015 19:28	margarita	fuentes	ventas	
10	05-08-2015 19:29	lizbeth	esteban	instalacion	10-08-2015 01:07

Mostrar# 10 ▼

Está aquí: Inicio ▶ Registro de Salida.

Figura 54 : Registro de salida.

6.2 Sistema de Base de Datos.

“Un sistema de base de datos es un sistema computarizado para llevar registros”. [9]

Su finalidad general es almacenar información y permitir a los usuarios recuperar o actualizar la información en base a peticiones solicitadas.

En los archivos de las bases de datos se pueden realizar una variedad de operaciones, tales como:

- ❖ Agregar nuevos datos.
- ❖ Insertar datos.
- ❖ Recuperar datos.
- ❖ Modificar datos.
- ❖ Eliminar datos.

La Figura 55 se muestra que un sistema de base de datos comprende cuatro componentes principales los cuales son:

- ❖ Los Datos.
- ❖ El Hardware.
- ❖ El Software.
- ❖ Y finalmente los Usuarios.

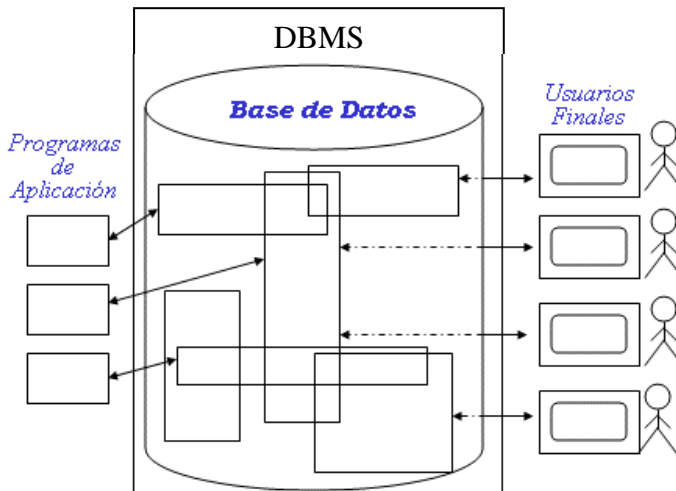


Figura 55: Sistema de base de datos.

La expresión base de datos, es utilizada con gran frecuencia; es por ello que empezaremos por definir su significado.

“Una base de datos es una colección de datos relacionados” [10].

Una base de datos tiene las siguientes propiedades implícitas:

- ❖ Representa algún aspecto del mundo real; lo que en ocasiones se denomina: mini mundo o universo de discurso.
- ❖ Es una colección de datos lógicamente coherente con algún tipo de significado inherente.

- ❖ Se diseña, construye y rellena con datos para un propósito en específico.

6.2.1 Sistema de administración de datos.

DBMS⁷ es el nombre bajo el cual se conoce a productos de fabricantes como Oracle, Sybase, Informix, Ingres, Borland, Microsoft, IBM, etc.

Es un software que controla la organización, almacenamiento, recuperación, seguridad e integridad de los datos en una base de datos.

Los DBMS pueden trabajar con lenguajes de programación tradicionales (COBOL, C, etc.) o pueden incluir su propio lenguaje de programación. [11]

En la Figura 56 se pueden observar los nombres y logotipos de algunos DBMS.



Figura 56: Sistema manejador de base de datos.

⁷ Sistema manejador de base de datos.

6.3 Conclusiones y perspectivas.

Debido a que el sistema ha sido creado implementando la metodología UWE y la ingeniería de software; podemos asegurar que se han realizado con éxito cada una de las etapas que éstas conllevan.

Este proyecto fue desarrollado, e implementado con base en **wampserver** y **joomla**, los cuales tienen un gran desempeño sin necesidad de sacrificar seguridad y estabilidad en la aplicación.

Se cumplieron todos los objetivos del proyecto, es decir, se desarrolló la aplicación web fácil de usar, agradable visualmente, y orientada a los diferentes tipos de usuarios que acceden a la DIST.

La aplicación se enfoca a cubrir las diferentes necesidades que tiene el administrador del sistema, permitiéndole tener un fácil acceso a la información que se solicite.

El sistema fue entregado en la Secretaria de Finanzas del estado de Puebla; específicamente al Ing. Alejandro Flores Torres quien está a cargo de la Dirección de Infraestructura y Soporte Técnico.

Se realizaron pruebas en un servidor local; y los resultados obtenidos fueron satisfactorios.

Cabe mencionar que la apariencia de la aplicación así como la información que se almacena en la base de datos fue plenamente satisfactoria para el cliente. De esta

manera fue posible verificar que nuestra aplicación Web cumpliera al 100% con los requerimientos y objetivos establecidos.

Este sistema garantiza la información oportuna y confiable de cada usuario, del mismo modo que las consultas sean aún más rápidas y sin redundancia de información.

Después de realizar las pruebas correspondientes y obtener satisfactorios resultados; bajo el mando del Ing. Alejandro Flores Torres, se procede a implementar el sistema en un servidor de pruebas de la SFA, esto con la finalidad que la aplicación solo sea usada y administrada únicamente por el personal de dicha secretaria.

Apéndice: A

A.1 Modelo Entidad-Relación

Es el modelo conceptual más utilizado para el diseño de bases de datos. Fue introducido por Peter Chen en 1976. El modelo entidad-relación está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

Originalmente, el modelo entidad-relación sólo incluía los conceptos de entidad, relación y atributo. Más tarde, se añadieron otros conceptos, como los atributos compuestos y las jerarquías de generalización, en lo que se ha denominado modelo entidad-relación extendido.

A.1.1 Entidad.

Es cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Existen dos tipos de entidades: fuertes y débiles. Una entidad débil es una entidad cuya existencia depende de la existencia de otra entidad. Una entidad fuerte es una entidad que no depende de otra.

A.1.2 Relación.

Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su

función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

Las entidades que están involucradas en una determinada relación se denominan entidades participantes. El número de participantes en una relación es lo que se denomina grado de la relación. Por lo tanto, una relación en la que participan dos entidades es una relación binaria; si son tres las entidades participantes, la relación es ternaria; etc.

Una relación recursiva es una relación donde la misma entidad participa más de una vez en la relación con distintos papeles. El nombre de estos papeles es importante para determinar la función de cada participación.

La cardinalidad con la que una entidad participa en una relación especifica el número mínimo y el número máximo de correspondencias en las que puede tomar parte cada ocurrencia de dicha entidad.

A.1.3 Atributo.

Es una característica de interés o un hecho sobre una entidad o sobre una relación. Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es portada por los atributos. Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen.

Cada atributo tiene un conjunto de valores asociados denominado dominio. El dominio define todos los valores posibles que puede tomar un atributo. Puede haber varios atributos definidos sobre un mismo dominio.

Los atributos pueden ser simples o compuestos. Un atributo simple es un atributo que tiene un solo componente, que no se puede dividir en partes más pequeñas que tengan un significado propio.

Un atributo compuesto es un atributo con varios componentes, cada uno con un significado por sí mismo. Un grupo de atributos se representa mediante un atributo compuesto cuando tienen afinidad en cuanto a su significado, o en cuanto a su uso. Un atributo compuesto se representa gráficamente mediante un óvalo.

Los atributos también pueden clasificarse en monovalentes o polivalentes. Un atributo monovalente es aquel que tiene un solo valor para cada ocurrencia de la entidad o relación a la que pertenece.

Un atributo polivalente es aquel que tiene varios valores para cada ocurrencia de la entidad o relación a la que pertenece.

A estos atributos también se les denomina multivaluados, y pueden tener un número máximo y un número mínimo de valores. La cardinalidad de un atributo indica el número mínimo y el número máximo de valores que puede tomar para cada ocurrencia de la entidad o relación a la que pertenece.

Por último, los atributos pueden ser derivados. Un atributo derivado es aquel que representa un valor que se puede obtener a partir del valor de uno o varios atributos, que no necesariamente deben pertenecer a la misma entidad o relación.

A.1.4 Identificador.

Un identificador de una entidad es un atributo o conjunto de atributos que determina de modo único cada ocurrencia de esa entidad. Un identificador de una entidad debe cumplir dos condiciones:

- No pueden existir dos ocurrencias de la entidad con el mismo valor del identificador.
- Si se omite cualquier atributo del identificador, la condición anterior deja de cumplirse.

Toda entidad tiene al menos un identificador y puede tener varios identificadores alternativos. Las relaciones no tienen identificadores.

A.1.5 Jerarquía de generalización.

Una entidad es una generalización de un grupo de entidades si cada ocurrencia de cada una de esas entidades es también una ocurrencia de entidad. Todas las propiedades de la entidad genérica son heredadas por las sub-entidades.

Cada jerarquía es total o parcial, y exclusiva o superpuesta.

Una jerarquía es total si cada ocurrencia de la entidad genérica corresponde al menos con una ocurrencia de alguna sub-entidad.

Es parcial si existe alguna ocurrencia de la entidad genérica que no corresponde con ninguna ocurrencia de ninguna sub-entidad.

Una jerarquía es exclusiva si cada ocurrencia de la entidad genérica corresponde, como mucho, con una ocurrencia de una sola de las subentidades. Es superpuesta si existe alguna ocurrencia de la entidad genérica que corresponde a ocurrencias de dos o más subentidades diferentes.

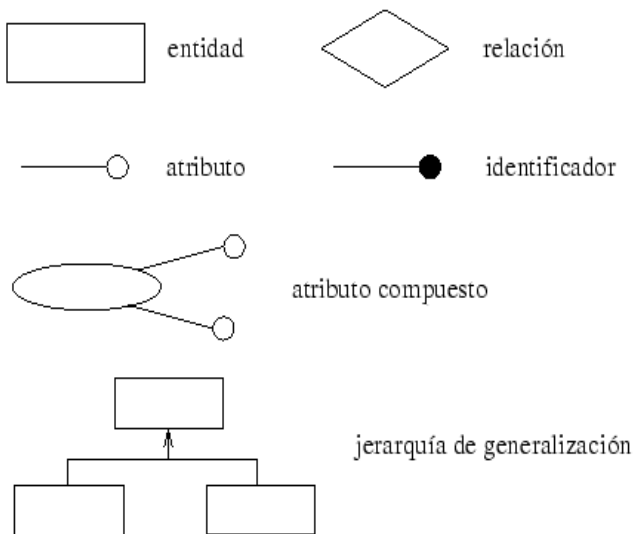


Figura 57: Componentes del modelo E-R.

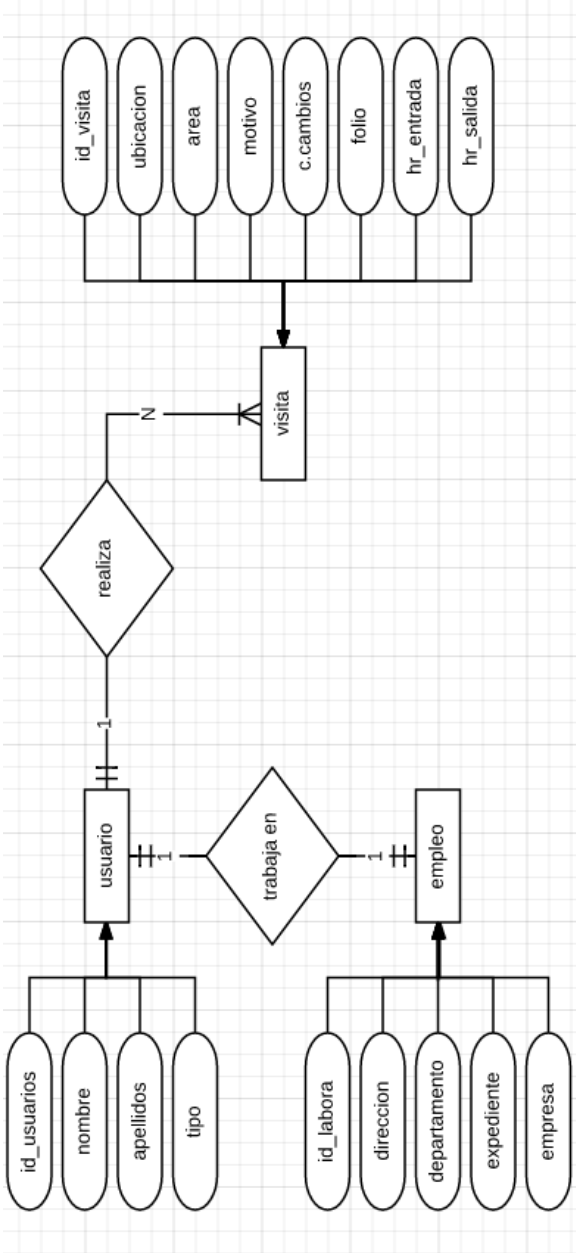


Figura 58: Diagrama entidad relación del sistema.

A.2 Modelo Relacional.

El modelo relacional representa la segunda generación de los SGBD. En él, todos los datos están estructurados a nivel lógico como tablas formadas por filas y columnas, aunque a nivel físico pueden tener una estructura completamente distinta. Un punto fuerte del modelo relacional es la sencillez de su estructura lógica.

Una entidad, representa una “cosa” u “objeto” del mundo real, se diferencia unívocamente de otro objeto o cosa, incluso siendo del mismo tipo, o una misma entidad. Los atributos, son las características que definen o identifican una entidad; pueden ser muchas, pero sólo se utilizan o implementan las más relevantes. Por otro lado, la relación describe cierta dependencia entre entidades permitiendo la asociación de las mismas [12].

El modelo relacional constituye una alternativa para la organización y representación de la información que se pretende almacenar en una base de datos. Se trata de un modelo basado en el concepto de relación. La información se representa en forma de “tablas” o relaciones, donde cada fila de la tabla se interpreta como una relación ordenada de valores. Una relación se define como un conjunto de n-tuplas, donde una n-tupla se define a su vez como un conjunto ordenado de valores atómicos; es decir, que no son divisibles en valores más pequeños [13].

Cada relación, vista como una tabla, consta de un conjunto de columnas; cada una de estas columnas recibe

el nombre de atributo. A cada atributo de una relación le corresponde un nombre, que debe ser único dentro de la relación, y un dominio (el conjunto de valores que cada tupla de la relación puede tomar para ese atributo).

Existen algunas restricciones de integridad que se deben cumplir, por ejemplo: de dominio: “los dominios de los atributos de una relación deben ser atómicos”; de clave: “en una relación no puede haber ninguna tupla repetida”; de integridad o identidad: “ninguna tupla de una relación puede tomar valores nulos en los atributos que forman parte de su llave primaria”; de integridad referencial: “si una tupla de una relación hace referencia a otra relación, debe referirse a una tupla que exista en ésta última”. Se llaman claves candidatas a aquellos atributos de los que ninguno puede ser eliminado sin provocar que el conjunto deje de ser una superclave de la relación; sólo una clave candidata podrá ser usada como clave o llave primaria de la relación cuando sea necesaria [14].

A.2.1 Estructura de datos.

Una relación es una tabla con columnas y filas. Un SGBD sólo necesita que el usuario pueda percibir la base de datos como un conjunto de tablas. Esta percepción sólo se aplica a la estructura lógica de la base de datos.

Un atributo es el nombre de una columna de una relación. En el modelo relacional, las relaciones se utilizan para almacenar información sobre los objetos que se

representan en la base de datos. Una relación se representa gráficamente como una tabla bidimensional en la que las filas corresponden a registros individuales y las columnas corresponden a los campos o atributos de esos registros. Los atributos pueden aparecer en la relación en cualquier orden.

A.2.2 Propiedades de relaciones.

Las relaciones tienen las siguientes características:

- Cada relación tiene un nombre y éste es distinto del nombre de todas las demás.
- Los valores de los atributos son atómicos: en cada tupla, cada atributo toma un solo valor. Se dice que las relaciones están normalizadas.
- No hay dos atributos que se llamen igual.
- El orden de los atributos no importa: los atributos no están ordenados.
- Cada tupla es distinta de las demás: no hay tuplas duplicadas.
- El orden de las tuplas no importa: las tuplas no están ordenadas.

A.2.3 Claves.

Ya que en una relación no hay tuplas repetidas, éstas se pueden distinguir unas de otras, es decir, se pueden

identificar de modo único. La forma de identificarlas es mediante los valores de sus atributos.

Una **superclave** es un atributo o un conjunto de atributos que identifican de modo único las tuplas de una relación.

Una **clave candidata** es una superclave en la que ninguno de sus subconjuntos es una superclave de la relación.

Cuando una clave candidata está formada por más de un atributo, se dice que es una **clave compuesta**. Una relación puede tener varias claves candidatas.

La **clave primaria** de una relación es aquella clave candidata que se escoge para identificar sus tuplas de modo único. Ya que una relación no tiene tuplas duplicadas, siempre hay una clave candidata y, por lo tanto, la relación siempre tiene clave primaria. En el peor caso, la clave primaria estará formada por todos los atributos de la relación, pero normalmente habrá un pequeño subconjunto de los atributos que haga esta función.

Las claves candidatas que no son escogidas como clave primaria son denominadas **claves alternativas**.

Una **clave ajena** es un atributo o un conjunto de atributos de una relación cuyos valores coinciden con los valores de la clave primaria de alguna otra relación. Las claves ajenas representan relaciones entre datos.

En la Figura 59 se muestra el diagrama relacional del sistema correspondiente a la bitácora DIST.

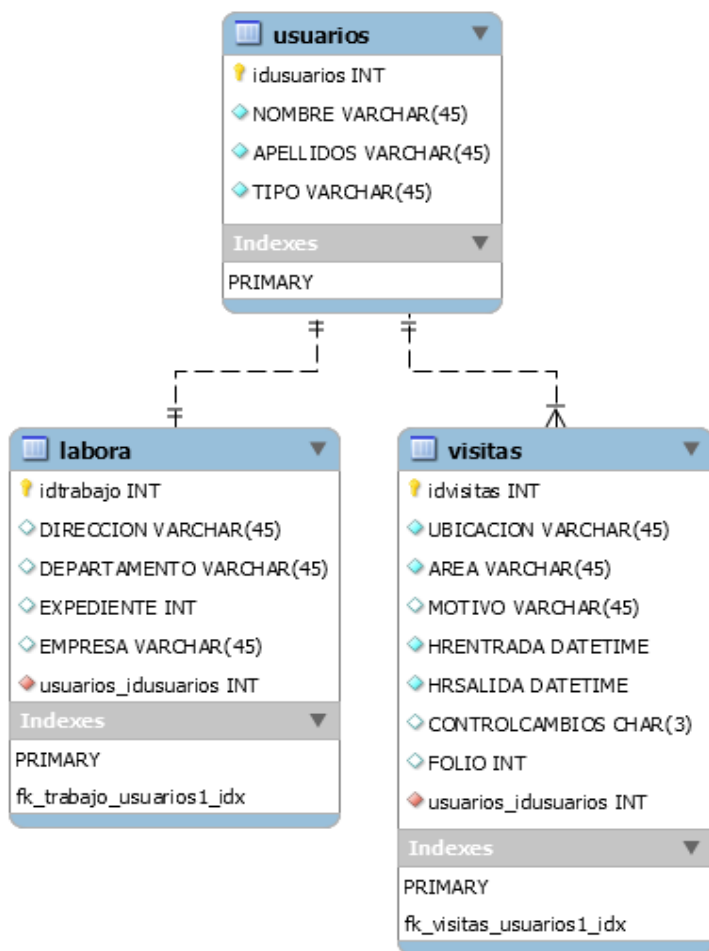


Figura 59: Modelo relacional de la bitácora DIST.

Después de realizar el modelo relacional mediante workbench 6.3; se procede a migrar el modelo realizado a phmyadmin.

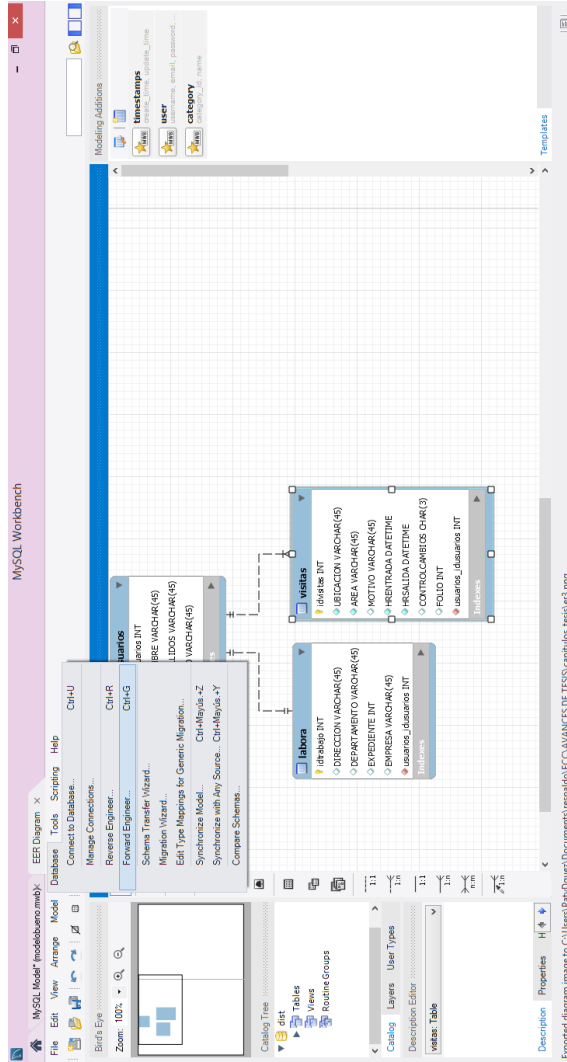


Figura 60: Seleccionar forward engineer.

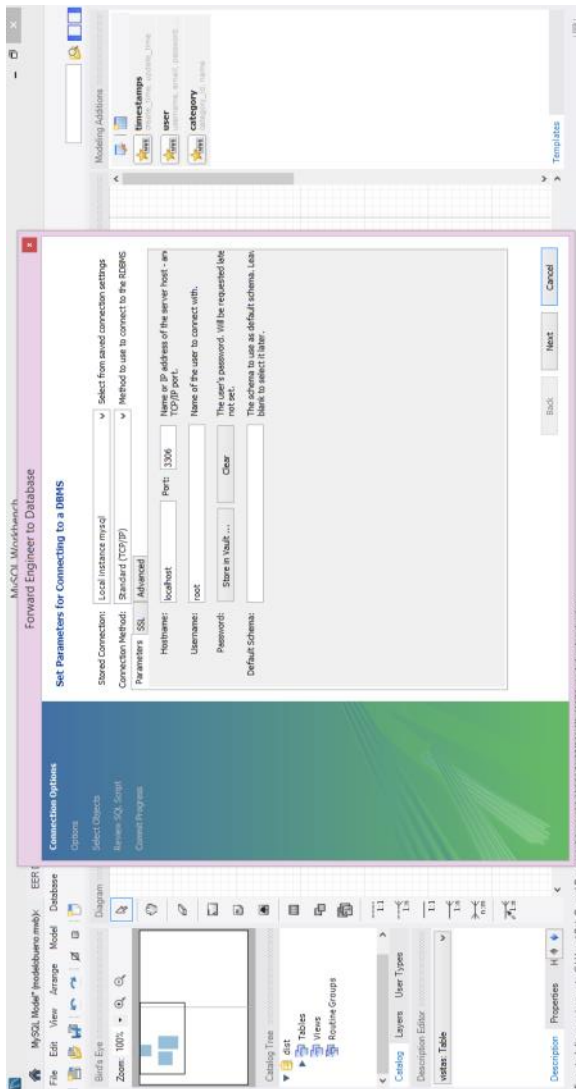


Figura 61: Parámetros establecidos para la conexión al DBMS.



Figura 62: Configuración de las opciones para crear la base de datos.

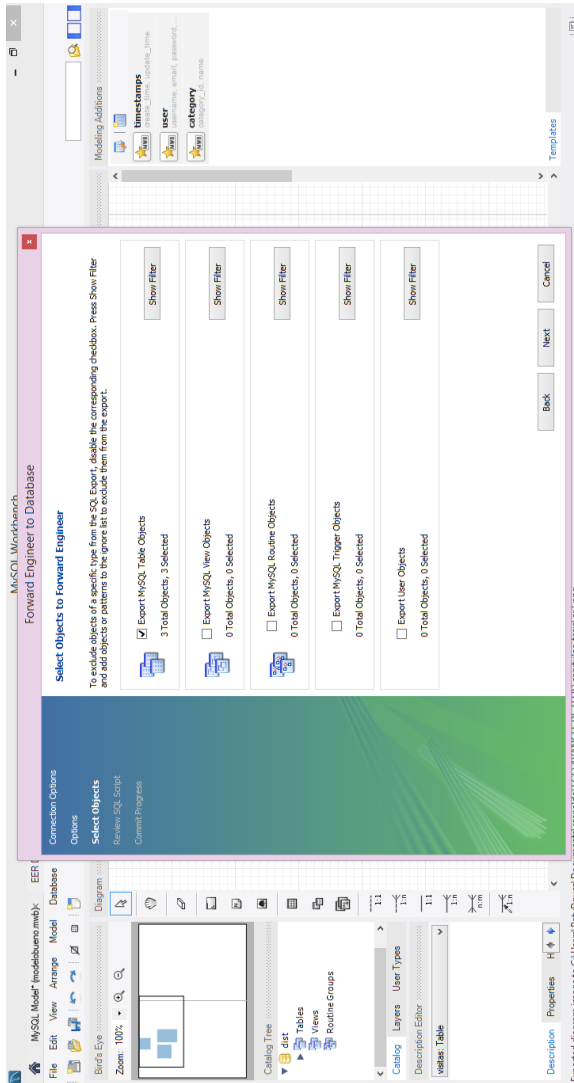
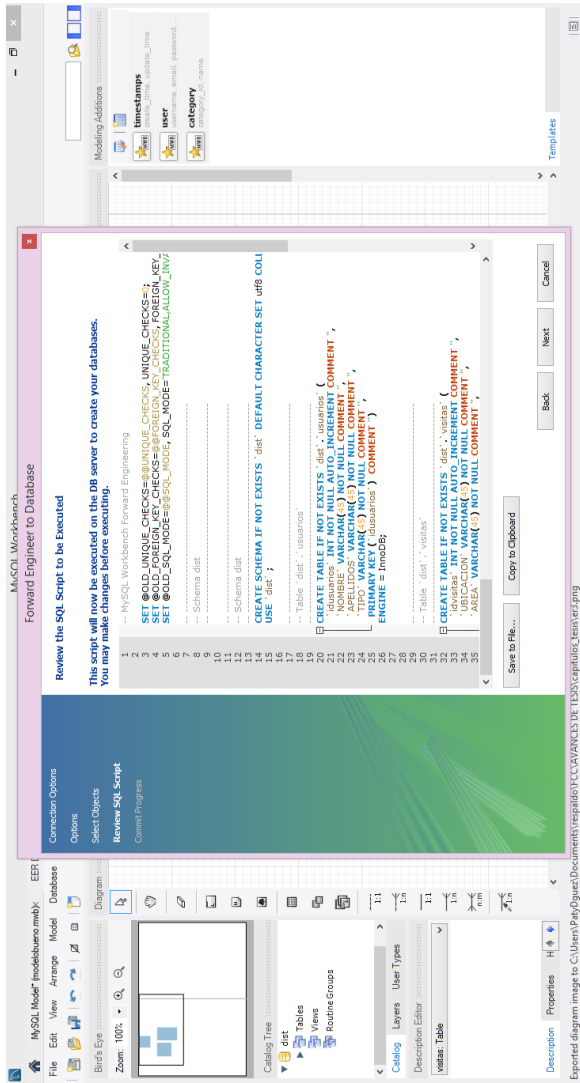


Figura 63: Selección de objetos.



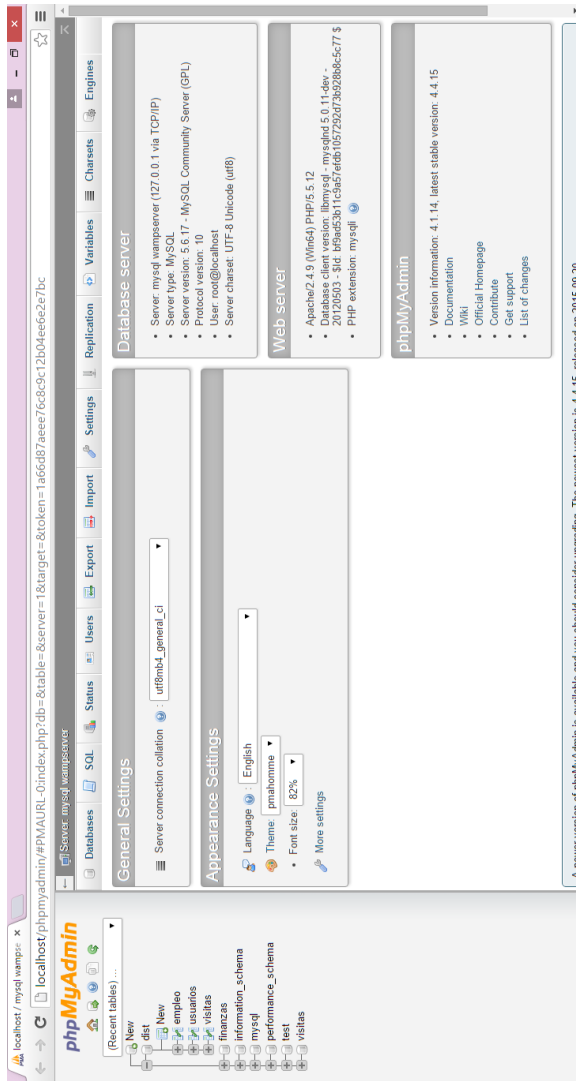


Figura 65: Tablas exportadas en phpMyAdmin.

A.3 Código de la bitácora DIST.

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;

SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;

SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

CREATE SCHEMA IF NOT EXISTS `dist` DEFAULT CHARACTER SET
utf8 COLLATE utf8_general_ci ;

USE `dist` ;

-- -----
-- Table `dist`.`usuarios`
-- -----
CREATE TABLE IF NOT EXISTS `dist`.`usuarios` (
  `idusuarios` INT NOT NULL AUTO_INCREMENT COMMENT '',
  `NOMBRE` VARCHAR(45) NOT NULL COMMENT '',
  `APELLIDOS` VARCHAR(45) NOT NULL COMMENT '',
  `TIPO` VARCHAR(45) NOT NULL COMMENT '',
  PRIMARY KEY (`idusuarios`) COMMENT '')
ENGINE = InnoDB;

-- -----
-- Table `dist`.`visitas`
-- -----
CREATE TABLE IF NOT EXISTS `dist`.`visitas` (
  `idvisitas` INT NOT NULL AUTO_INCREMENT COMMENT '',
  `UBICACION` VARCHAR(45) NOT NULL COMMENT '',
  `AREA` VARCHAR(45) NOT NULL COMMENT '',
```

```

`MOTIVO` VARCHAR(45) NULL COMMENT '',
`HRENTRADA` DATETIME NOT NULL COMMENT '',
`HR SALIDA` DATETIME NOT NULL COMMENT '',
`CONTROLCAMBIOS` CHAR(3) NULL COMMENT '',
`FOLIO` INT NULL COMMENT '',
`usuarios_idusuarios` INT NOT NULL COMMENT '',
PRIMARY KEY (`idvisitas`) COMMENT '',
INDEX `fk_visitas_usuarios1_idx` (`usuarios_idusuarios`
ASC) COMMENT '',
CONSTRAINT `fk_visitas_usuarios1`
FOREIGN KEY (`usuarios_idusuarios`)
REFERENCES `dist`.`usuarios` (`idusuarios`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

-----
-- Table `dist`.`labora`
-----
CREATE TABLE IF NOT EXISTS `dist`.`labora` (
`idtrabajo` INT NOT NULL AUTO_INCREMENT COMMENT '',
`DIRECCION` VARCHAR(45) NULL COMMENT '',
`DEPARTAMENTO` VARCHAR(45) NULL COMMENT '',
`EXPEDIENTE` INT NULL COMMENT '',
`EMPRESA` VARCHAR(45) NULL COMMENT '',
`usuarios_idusuarios` INT NOT NULL COMMENT '',

```

```
PRIMARY KEY (`idtrabajo`) COMMENT '',
INDEX `fk_trabajo_usuarios1_idx` (`usuarios_idusuarios`
ASC) COMMENT '',
CONSTRAINT `fk_trabajo_usuarios1`
FOREIGN KEY (`usuarios_idusuarios`)
REFERENCES `dist`.`usuarios` (`idusuarios`)
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB;

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Apéndice: B

B.1 Wamp server V2.5

Su nombre proviene de las siglas de Windows, Apache MySQL y Php. Es un programa todo en uno, que provee de un servidor local y sirve para instalar o desarrollar sobre Windows aplicaciones que usen estas tecnologías.

Un servidor local nos permite acceder a una aplicación situada en un algún directorio de nuestro disco duro a través del navegador como si se tratase de cualquier sitio web, pero al que solo vamos a poder acceder desde nuestra pc.

Es usado por desarrolladores y/o programadores para realizar prácticas de sus proyectos web antes de subirlos al servidor web en internet. [15]

Características:

- Manejo de Bases de datos con MySQL.
- Software para servidor web Apache.
- Software para poder programar script con PHP.
- Permite el manejo sencillo de Bases de Datos con PHPMyAdmin y SQLiteManager
- Es completamente gratuito

B.1.1 Apache.

El servidor Apache HTTP , también llamado Apache, es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, gratuito, muy robusto y que destaca por su seguridad y rendimiento.

Para entender mejor lo que es Apache, primeramente definiremos lo que es un servidor web. La definición más sencilla de servidor web, que es un programa especialmente diseñado para transferir datos de hipertexto, es decir, páginas web con todos sus elementos (textos, widgets, banners, etc.). Estos servidores web utilizan el protocolo http.

Los servidores web están alojados en un ordenador que cuenta con conexión a Internet. El web server, se encuentra a la espera de que algún navegador le haga alguna petición, como por ejemplo, acceder a una página web y responde a la petición, enviando código HTML mediante una transferencia de datos en red. [16]

B.1.2 MySQL.

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. Es muy utilizado en aplicaciones web, como Joomla, Wordpress, Drupal o phpBB. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. [17]

Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

B.1.3 PHP.

PHP es un lenguaje de scripting embebido en HTML, significa Hypertext Preprocesor (Preprocesador de Hipertexto). Mucha de su sintaxis es tomada de C, Java y Pearl con un par de características adicionales únicas y específicas de PHP.

Su propósito es permitir que los desarrolladores Web escriban páginas generadas dinámicamente con rapidez.

Diseñado para ser un lenguaje más seguro para escribir programas, a diferencia de Pearl o C, y con la correcta selección de las opciones de configuración del servidor y el servidor Web; es posible conseguir una combinación equilibrada de libertad y seguridad combinación

equilibrada de libertad y seguridad que requieren las aplicaciones que operan en un ambiente naturalmente inseguro como lo es la Web. [18]

B.2 Joomla V3.4.

Joomla es un Gestor de Contenidos para desarrollos web; es una especie de programa para gestionar páginas web. En inglés se denomina CMS (Content Management System).

Existen diversos CMS's de libre difusión (p.ej. Mambo, Drupal, Nucleus, etc.) y Joomla es uno de los más utilizados. Se basa en estándares propios de desarrollos web como XHTML, CSS, JavaScript y PHP.

Joomla es una aplicación web, es decir, no se ejecuta en nuestro equipo local sino en un servidor (aunque podemos hacer funcionar a nuestro equipo local como servidor).

Joomla es software libre: está desarrollado por una comunidad de usuarios que lo mejoran, actualizan y lo ponen a disposición para su uso gratuitamente. Permite publicar sitios web cuyos contenidos cambian continuamente, fáciles de administrar y ampliar sin necesidad de realizar una inversión económica en software.

Joomla se distribuye con unos contenidos básicos y a esta versión de base la denominamos "Joomla nativo".

Además se está continuamente ampliando a través de nuevos módulos y plantillas que se van publicando periódicamente.

Joomla permite tanto la publicación por parte del gestor de la página web como la interacción de los usuarios de la web a través de publicación de artículos, encuestas o colaboraciones en la gestión del sitio.

Joomla es una solución indicada para personas que quieren publicar una web centrándose en los contenidos y dejando que el backup se base en un desarrollo de software libre. [19]

Otra característica de Joomla es que pertenece a los proyectos llamados de código libre: se puede descargar e instalar de forma gratuita; además existen sitios en internet donde se pueden bajar plantillas, módulos, plugins (también llamadas extensiones), e instalarlas fácilmente en nuestro sitio para mejorar la funcionalidad del mismo.

B.2.1 Instalación de Joomla.

Para instalar Joomla solo se requerirá de un servidor web que tenga además soporte PHP y MySQL.

Una vez que se instaló el servidor, se procede a descargar Joomla de la web: <http://joomla.com.es/>

Al descargarlo obtendremos el archivo:

Joomla_3.4.0-Stable-Full_Package.zip (existen versiones anteriores y posteriores, podremos elegir la que deseemos)

Desmontaremos el archivo .Zip en la carpeta *C:\wamp\www* y le asignamos a la carpeta el nombre de nuestro sitio web.

Una vez extraídos los archivos, nos vamos con el navegador de nuestra elección al URL <http://localhost/sitioweb>

Y encontraremos la ventana de bienvenida de la instalación; en la cual se nos pedirá un nombre para el sitio, una cuenta con clave para administrar el sitio.

Joomla genera la base de datos del sitio, por ultimo hay que borrar el directorio de instalación. [20]

B.2.2 HTML.

Hypertext Markup Language por sus siglas en inglés (lenguaje de marcado hipertextual), hace referencia al lenguaje marcado para la elaboración de páginas web.

Es un estándar que define una estructura física y un código (denominado HTML) para la definición de un contenido de una página web, como texto, imágenes, etc.

HTML basa su filosofía de desarrollo en la Referenciación. Para añadir un elemento externo a la página (imagen, video, script, etc.), este no se incrusta

directamente en el código de la página, sino que se hace referencia a la ubicación de dicho elemento mediante texto.

De este modo, la página web contiene solo texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado. [21]

B.2.3 JavaScript.

JavaScript es un lenguaje de scripting multiplataforma y orientado a objetos. Es un lenguaje pequeño y liviano. Dentro de un ambiente de host, JavaScript puede conectarse a los objetos de su ambiente y proporcionar control programático sobre ellos.

JavaScript contiene una librería estándar de objetos, tales como Array, Date, y Math, y un conjunto central de elementos del lenguaje, tales como operadores, estructuras de control, y sentencias. El núcleo de JavaScript puede extenderse para varios propósitos, complementándolo con objetos adicionales, por ejemplo:

- Client-side JavaScript extiende el núcleo del lenguaje proporcionando objetos para controlar un navegador y su modelo de objetos (o DOM,

por las iniciales de Document Object Model). Por ejemplo, las extensiones del lado del cliente permiten que una aplicación coloque elementos en un formulario HTML y responda a eventos del usuario, tales como clicks del ratón, ingreso de datos al formulario y navegación de páginas.

- Server-side JavaScript extiende el núcleo del lenguaje proporcionando objetos relevantes a la ejecución de JavaScript en un servidor. Por ejemplo, las extensiones del lado del servidor permiten que una aplicación se comunique con una base de datos, proporcionar continuidad de la información de una invocación de la aplicación a otra, o efectuar manipulación de archivos en un servidor. [22]

B.2.4 CSS.

CSS es el acrónimo de Cascading Style Sheets (es decir, hojas de estilo en cascada).

CSS es un lenguaje de estilo que define la presentación de los documentos HTML. Por ejemplo, CSS abarca cuestiones relativas a fuentes, colores, márgenes, líneas, altura, anchura, imágenes de fondo, posicionamiento avanzado y muchos otros temas. [23]

CSS fue toda una revolución en el mundo del diseño web. Entre los beneficios concretos de CSS encontramos:

- Control de la presentación de muchos documentos desde una única hoja de estilo;
- Control más preciso de la presentación.
- Aplicación de diferentes presentaciones a diferentes tipos de medios (pantalla, impresión, etc.)
- Numerosas técnicas avanzadas y sofisticadas.

Bibliografía

- [1] R. P. P. Stevens, Utilizacion de uml en ingenieria del software con objetos y componentes, Madrid: Pearson Educacion, S.A., 2002.
- [2] A. Cockburn, «Escribir casos de uso efectivos.,» Pearson Education Corporate Sales Division., 2000.
- [3] S. Ceria, «Casos de Uso,» [En línea]. Available: http://www-2.dc.uba.ar/materias/isoft1/2001_2/apuntes/CasosDeUso.pdf.
- [4] C. Larman, UML Y PATRONES: Introduccion al analisis y diseño orientado a objetos., Mexico: PRENTICE HALL, 1999.
- [5] D. M. R. Lopez, «APLICACIONES WEB».
- [6] «Kioskea.net,» [En línea]. Available: <http://es.kioskea.net/contents/409-clases-de-objetos-e-instancias>.
- [7] «UWE – UML-based Web Engineering,» [En línea]. Available: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>. [Último acceso: agosto 2015].

- [8] R. de la Rosa Flores, B. Beltran Martinez, H. Castillo Zacatelco , L. Mendoza Alonzo y D. Vilariño Ayala, Sistemas Operativos Centralizados y Distribuidos, Puebla, Puebla: Direccion general de fomento editorial, 2009.
- [9] C. DATE, Introduccion a los sistemas de bases de datos., Mexico: PEARSON EDUCACION, 2001.
- [10] R. E. y. S. B. Navathe, Fundamentos de sistemas de bases de datos., Madrid: PEARSON EDUCACION, 2007.
- [11] «Mater Magazine,» [En línea]. Available: <http://www.mastermagazine.info/termino/4544.php>. [Último acceso: 29 agosto 2015].
- [12] P. M. M. E. De Miguel A., Diseño de Base de Datos Relacionales, RAMA. .
- [13] N. S. E. A. Elmasri R., Fundamentos de los sistemas de Bases de Datos, 2002: AddisonWesley.
- [14] K. H. S. S. Silberschatz A., Fundamentos de Bases de Datos, Mac-Graw Hill., 2002.
- [15] O. C. Elorz, «Qué es Wamp Server? - Aula Virtual SNE,» [En línea]. Available: http://aulasne.navarra.es/pluginfile.php/2477/mod_page/content/34/instalar_wamp.pdf. [Último

acceso: 25 septiembre 2015].

[16] E. F. Cases, «ibrugor.com,» [En línea]. Available: <http://www.ibrugor.com/blog/apache-http-server-que-es-como-funciona-y-para-que-sirve/>. [Último acceso: 25 septiembre 2015].

[17] «wikipedia,» [En línea]. Available: <https://es.wikipedia.org/wiki/MySQL>. [Último acceso: 25 septiembre 2015].

[18] G. C. Martín F., Introducción a PHP, Universidad de Sevilla..

[19] «aprenderaprogramar.com,» [En línea]. Available: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=119:ique-es-y-para-que-sirve-joomla-cms-potente-e-interesante-para-crear-paginas-web-de-todo-tipo-cu00403a&catid=38:curso-qcreacion-web-con-joomla-desde-ceroq&Itemid=152. [Último acceso: 25 septiembre 2015].

[20] A. M. Peñuela, Domine Joomla Manual Practico, Mexico: Alfaomega Ra-Ma, 2013.

[21] S. Luján, “Programación en Internet: clientes web”, Ed. Club Universitario., 2001.

[22] «MDN,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/JavaScri>

pt/Guide/Introducci%C3%B3n. [Último acceso: 25 septiembre 2015].

[23] «HTML.net,» [En línea]. Available: <http://es.html.net/tutorials/css/lesson1.php>. [Último acceso: 25 septiembre 2015].