

**BENEMERITA UNIVERSIDAD AUTONOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA COMPUTACION**



SIAC

SISTEMA INTEGRAL PARA LA ADMINISTRACION DE CURRICULOS



TESIS

Que para obtener el Título de:

INGENIERO EN CIENCIAS DE LA COMPUTACION

Presenta:

MISAEAL RODRÍGUEZ MORÓN



Asesora:

M.C. ALMA DELIA AMBROSIO VÁZQUEZ

Noviembre 2009

Aquél que obtiene una victoria sobre otro hombre es fuerte, pero quien obtiene una victoria sobre sí mismo es poderoso.

Lao-tse

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

Thomas Alva Edison

Con Amor y Respeto

A Mis Padres quienes supieron darme el mejor regalo que un hijo puede tener, la oportunidad de ser un Hombre de Bien, hoy y siempre les doy gracias por todas sus enseñanzas y gracias por darme su apoyo y confianza. Quiero que sepan que éste triunfo es también suyo.

Con Admiración

A Marisol Flores Hernández, pilar que se mantuvo en pie bajo cualquier circunstancia dándome la fuerza para alcanzar ésta meta.

Con Cariño

A mis Maestros y Amigos

*El sustentante
Misael Rodríguez Morón*

INDICE

Capítulo 1 Marco Teórico

- 1.1 ¿Qué es el Software?.....2
 - 1.1.1 Historia del Software.....2
 - 1.1.2 Aplicaciones del Software.....2
 - 1.1.3 Características del Software3
- 1.2 Ingeniería de Software3
- 1.3 Modelos de Desarrollo de Software4
- 1.4 Modelo a Utilizar.....5
- 1.5 Bases de Datos5
 - 1.5.1 Componentes de un Sistema de Bases de Datos.....5
 - 1.5.2 Tipos de Bases de Datos6
 - 1.5.2.1 Según la variabilidad de los datos almacenados.....6
 - 1.5.2.2 Según el contenido.....7
 - 1.5.2.3 De acuerdo al modelo que utiliza8
- 1.6 Normalización 11
 - 1.6.1 Normalización en las bases de Datos12
- 1.7 Arquitectura Cliente Servidor14
 - 1.7.1 Ventajas de la arquitectura cliente-servidor 15
- 1.8. Lenguajes a Usar 16
 - 1.8.1 MySQL.....16
 - 1.8.2 PHP16

Capítulo 2 Análisis de Requerimientos

- 2.1 Especificaciones 19
- 2.2 Requisitos del sistema de acuerdo a las especificaciones del cliente. 19
 - 2.2.1 Requisitos Funcionales 19
 - 2.2.2 Requisitos no funcionales 20
- 2.3 Análisis del sistema 20
 - 2.3.1 Gestionar Currículo 23
 - 2.3.2 Modificar Currículo..... 24
 - 2.3.3 Eliminar Currículo..... 24
 - 2.3.4 Buscar Oferta 25
 - 2.3.5 Buscar Egresado 25
 - 2.3.6 Gestionar Oferta..... 26
 - 2.3.7 Registrar Empresa 26

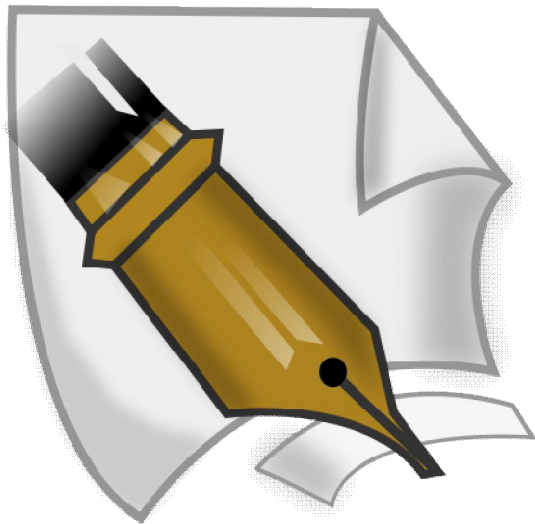
Capítulo 3 Diseño

- 3.1 Diseño Conceptual 29
 - 3.1.1 Identificar las entidades..... 30
 - 3.1.2 Identificar las relaciones 32
 - 3.1.3 Identificar los atributos y asociarlos a entidades y relaciones. 34
 - 3.1.3.1 Nombre y descripción del atributo..... 35
 - 3.1.4 Determinar los dominios de los atributos 38
 - 3.1.5 Determinar los identificadores 38
 - 3.1.6 Determinar las jerarquías de generalización 38
 - 3.1.7 Dibujar el diagrama entidad-relación..... 38
 - 3.1.8 Revisar el esquema conceptual local con el usuario..... 40
- 3.2 Diseño Lógico 40
- 3.3 Diseño Físico 42
- 3.4 Normalización 43
 - 3.4.1 Diagrama de dependencia Funcional 43
 - 3.4.2 Formas Normales 44
- 3.5 Diccionario de Datos 44
- 3.6 Tablas..... 47

Capítulo 4 Implementación

- 4.1 Ambiente de las herramientas de implementación 55
 - 4.1.1 MySQL y phpMyAdmin 55
- 4.2 Descripción general del sistema 55
 - 4.2.1 Pagina Principal 55
 - 4.2.2 Registro 56
 - 4.2.3 Ofertas 63
 - 4.2.4 Búsqueda 65
 - 4.2.5 Contacto..... 66
- 4.3 Pruebas y Resultados 68
 - 4.3.1 Registro de Un Egresado..... 68
 - 4.3.2 Registro de una Empresa 71
 - 4.3.3 Registro de una Oferta 72
 - 4.3.4 Busqueda 74
- CONCLUSIONES..... 77
- Trabajo a Futuro 77
- BIBLIOGRAFIA 78
- ANEXO: CODIGO FUENTE DEL SISTEMA..... 80

Capítulo



1

Marco Teórico

1.1 ¿Qué es el Software?

El Software es el conjunto de programas o instrucciones que se utilizan para realizar funciones determinadas.

1.1.1 Historia del Software

En la primera era de las computadoras, el software simplemente era algo extra, un producto de más, ya que este se realizaba sin ninguna planeación, bajo pedido y por tanto de distribución limitada, en la segunda era de las computadoras, el software se vuelve producto comercial, esto por que surgen las bases de datos, las operaciones en tiempo real y los multiusuarios, para la tercera generación debido al alto consumo del hardware por su bajo costo, el software se coloca ya como un producto de desarrollo, apareciendo así los sistemas distribuidos, En la cuarta generación el software es el factor más importante en la economía mundial, la tecnología es ahora orientada a objetos, la computación es en paralelo, y se crean las redes de computadora, el software tiene una gran demanda en una gran variedad de aplicaciones del mundo real tales como: redes neuronales artificiales, reconocimiento de patrones humanos, la programación de realidad virtual [6].

1.1.2 Aplicaciones del Software

Conforme más avanza la tecnología, el software va adquiriendo más demanda, sin embargo, de esta demanda se puede hacer una clasificación sobre qué tipos de software se requieren, aun que actualmente la lista es enorme, aquí sólo mencionaremos los más comunes [6].

- **Software de Sistema:** Es un conjunto de programas que sirven de base para otros programas, principalmente este tipo de software interactúa fuertemente con el hardware.
- **Software de Tiempo Real:** Son programas que miden, analizan, controlan sucesos del mundo real, conforme ocurren, es decir, los programas de tiempo real deben responder dentro de unas ligaduras estrictas de tiempo.
- **Software de Gestión:** Son programas que procesan información comercial accediendo a una o más bases para facilitar las operaciones comerciales y así gestionar la toma de decisiones, además de esto, también realizan cálculo interactivo.
- **Software Científico:** Estos programas se caracterizan por el manejo de números, sus aplicaciones las encontramos por ejemplo, en la astronomía, vulcanología, dinámica, biología molecular, etc.
- **Software Empotrado:** Este tipo de programas reside en memorias de sólo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo, este tipo de programas puede ejecutar funciones muy limitadas y curiosas.

- **Software de Computadoras Personales:** Este tipo de programas tienen muchas aplicaciones, tales como multimedia, aplicaciones financieras, aplicaciones personales, redes, acceso a bases de datos externas, por mencionar algunas de las muchas aplicaciones.
- **Software de Inteligencia Artificial:** Estos programas hacen uso de algoritmos numéricos para resolver problemas complejos para los que no son adecuados en cálculo o el análisis directo. El área más activa de la Inteligencia Artificial son los sistemas expertos, otras áreas de aplicación son el reconocimiento de patrones, las pruebas de teoremas y los juegos, una nueva rama son las redes neuronales artificiales.

1.1.3 Características del Software

A diferencia de cualquier otro producto que el hombre produce, el software es un producto lógico, por lo que sus características difieren a los productos físicos [5].

- El software es un producto que se desarrolla, es decir no tiene piezas físicas que se unan y al final se genere un producto físico.
- El software no se echa a perder, es decir, no se va descomponiendo al mal uso o bien a las limitaciones físicas como son el polvo, etc.
- El software se desarrolla a la medida, en lugar de unir partes ya existentes.

1.2 Ingeniería de Software

Anteriormente ya hemos definido que es el software, veamos ahora que es la ingeniería; la Ingeniería es el estudio y la aplicación de procesos, métodos y herramientas, utilizado por especialistas de diversas ramas de la tecnología, y así pues, tenemos que Ingeniería de Software es la aplicación de procesos, métodos y herramientas utilizados para desarrollar programas con funciones determinadas.

Para que se pueda desarrollar software de diferentes aplicaciones, existen pasos comunes en todos los modelos del desarrollo de software:

- **Análisis de requerimientos:** Generalmente el cliente piensa que sabe lo que debe hacer el software, sin embargo se requiere de experiencia así como de habilidades innatas para encontrar los requisitos sin que estos sean ambiguos.
- **Especificación:** En esta tarea se especifica a manera de detalle el software que se va a desarrollar.
- **Diseño y Arquitectura:** Consiste en indicar de forma general las funciones del software así mismo se indican las características del hardware.
- **Programación:** Es la tarea de transferir el diseño a código de un lenguaje de

programación ya establecido.

- Prueba: Esta tarea consiste en comprobar que el software realice de manera correcta las funciones que se detallan en la especificación.
- Documentación: Esta tarea consiste en realizar el manual del usuario así como el manual técnico para futuros mantenimientos y actualizaciones del software.
- Mantenimiento: Agregar nuevos requisitos al software, así como corregir error encontrados, significa darle mantenimiento al software.

1.3 Modelos de Desarrollo de Software

- Modelo Lineal o Secuencial: Es un modelo sistemático o como su nombre lo indica, secuencial, es decir, a un paso, como secuencia, le sigue otro, las etapas de este modelo son: Análisis de requisitos, diseño, generación de código, pruebas y mantenimientos.
- Modelo de Prototipos: Dado los objetivos generales se hace un diseño rápido y se realiza un prototipo que muestran al usuario los aspectos visibles del software.
- Modelo DRA: El modelo Rápido de Aplicaciones (Rapid Application Development), es un modelo secuencial con un ciclo de desarrollo extremadamente corto.
- Modelos de Procesos Evolutivos: Los modelos evolutivos son iterativos, ya que se caracterizan por permitir a los desarrolladores hacer versiones más completas del software, existen diferentes tipos de estos modelos:
 - Modelo Incremental: Este modelo se caracteriza por la entrega de un producto operacional con cada incremento.
 - Modelo en Espiral: Es el modelo de prototipos sólo que esta integra procesos sistemáticos del modelo secuencial, este modelo se divide en un número de actividades estructurales: comunicación con el cliente, planificación, análisis de riesgos, ingeniería, construcción y adaptación y finalmente, evaluación del cliente.
 - Modelo de Ensamblaje de Componentes: Este modelo ensambla aplicaciones de componentes ya preparados (llamados clases) de software, estas clases fueron creadas en otros proyectos de desarrollo de software, las cuales se tienen almacenadas en una biblioteca de clases.
 - Modelo de Desarrollo Concurrente: Este modelo consiste en realizar en serie las actividades técnicas importantes del software, es decir, en vez de realizar de manera secuencial las tareas importantes del desarrollo de software, las realiza en una red simultánea de actividades.

- **Modelo de Métodos Formales:** Este modelo tiene tareas implícitas que llevan a especificar el software por medio de un lenguaje formal [6].

1.4 Modelo a Utilizar

El modelo que se utilizó para este proyecto es una combinación del espiral y del prototipo debido a que el modelo de prototipo permite que el cliente no pierda de vista las funciones visibles del software y el modelo espiral nos lleva paso a paso al desarrollo del mismo.

1.5 Bases de Datos

Una **base de datos** o **banco de datos** es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos tienen formato electrónico, que ofrece un amplio rango de soluciones al problema de almacenar datos.

En informática existen los sistemas gestores de bases de datos (SGBD), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de los sistemas gestores de bases de datos se estudian en informática. Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

1.5.1 Componentes de un Sistema de Bases de Datos

- **Datos:** Un dato es un conjunto de texto, un número o un conjunto arbitrario de ambos.
- **Metadatos:** Es la información relevante sobre los datos almacenados.
- **Sistema Gestor de Bases de Datos (SGBD):** Es un programa que proporciona la interfaz a los usuarios, también se encarga de definir, crear y mantener la base de datos además proporciona un acceso controlado a dicha base de datos.
- **Elementos de Seguridad:** La seguridad en los elementos que se disponen es uno de los factores más importantes en una Base de datos, existen tres niveles de acceso. La primera, los registros a los que se tienen acceso, la segunda, las operaciones que se le pueden realizar a dichos registros y tercero el acceso a la estructura de la base de datos así como al diccionario de datos, esto, en función de los diferentes usuarios.
- **Lenguajes de Bases de datos:** En una Base de datos existen dos tipos de de

lenguaje, el lenguaje de definición de datos (LDD), que como su nombre lo indica sirve para definir los datos y el lenguaje de manipulación de datos (LMD), que sirve para manejar los datos [7].

1.5.2 Tipos de Bases de Datos

Existen diferentes tipos de bases de datos, las primeras fueron las jerárquicas y en red, posteriormente fueron sustituidas por las bases de datos relacionales y las bases de datos orientadas a objetos. En este proyecto, ocuparemos las bases de datos relacionales para modelar el problema y administrar los datos, debido a que son las más utilizadas en la actualidad.

Éste es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "duplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Edgar Frank Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que representarían las duplas, y campos (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos [1]. Que veremos en el siguiente apartado mientras tanto continuaremos con una breve descripción de los demás tipos de bases de Datos

1.5.2.1 Según la variabilidad de los datos almacenados

- Bases de datos estáticas

Son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

- Bases de datos dinámicas

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un videoclub, etc.

1.5.2.2 Según el contenido

Bases de datos bibliográficas

Solo contienen un su rogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque si no estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias - ver más abajo). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

Bases de datos de texto completo

Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

Directorios

Un ejemplo son las guías telefónicas en formato electrónico.

Bases de datos o "bibliotecas" de información Biológica

Son bases de datos que almacenan diferentes tipos de información proveniente de las ciencias de la vida o médicas. Se pueden considerar en varios subtipos:

- Aquellas que almacenan secuencias de nucleótidos o proteínas.
- Las bases de datos de rutas metabólicas
- Bases de datos de estructura, comprende los registros de datos experimentales sobre estructuras 3D de biomoléculas
- Bases de datos clínicas
- Bases de datos bibliográficas (biológicas)

1.5.2.3 De acuerdo al modelo que utilizan

Un modelo de datos es básicamente una "descripción" de algo conocido como contenedor de datos (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

- **Bases de datos jerárquicas**

Éstas son bases de datos que, como su nombre indica, almacenan su información en una estructura jerárquica. En este modelo los datos se organizan en una forma similar a un árbol (visto al revés), en donde un nodo padre de información puede tener varios hijos. El nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se los conoce como hojas.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

- **Base de datos de red**

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por

programadores más que por usuarios finales. Jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad

- **Base de datos relacional**

Éste es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Edgar Frank Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que representarían las tuplas, y campos (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años '80 (1980-1989) la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión.

- **Bases de datos multidimensionales**

Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como creación de Cubos OLAP. Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos multidimensional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.

- **Bases de datos orientadas a objetos**

Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los objetos completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

Se está trabajando en SQL3, que es el estándar de SQL92 ampliado, que soportará los nuevos conceptos orientados a objetos y mantendría compatibilidad con SQL92.

- **Bases de datos documentales**

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. Taurus es un sistema de índices optimizado para este tipo de bases de datos.

- **Base de datos deductivas**

Un sistema de base de datos deductivas, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. También las bases de datos deductivas son llamadas base de datos lógica, a raíz de que se basan en lógica matemática.

- **Gestión de bases de datos distribuida**

La base de datos está almacenada en varias computadoras conectadas en red. Surgen debido a la existencia física de organismos descentralizados. Esto les da la capacidad de unir las bases de datos de cada localidad y acceder así a distintas universidades, sucursales de tiendas, etcétera.

1.6 Normalización

La normalización o estandarización es la redacción y aprobación de normas que se establecen para garantizar el acoplamiento de elementos construidos independientemente, así como garantizar el repuesto en caso de ser necesario, garantizar la calidad de los elementos fabricados y la seguridad de funcionamiento.

La normalización es el proceso de elaboración, aplicación y mejora de las normas que se aplican a distintas actividades científicas, industriales o económicas con el fin de ordenarlas y mejorarlas. La asociación estadounidense para pruebas de materiales (ASTM), define la normalización como el proceso de formular y aplicar reglas para una aproximación ordenada a una actividad específica para el beneficio y con la cooperación de todos los involucrados.

Según la ISO (International Organization for Standardization) la Normalización es la actividad que tiene por objeto establecer, ante problemas reales o potenciales, disposiciones destinadas a usos comunes y repetidos, con el fin de obtener un nivel de ordenamiento óptimo en un contexto dado, que puede ser tecnológico, político o económico.

La normalización persigue fundamentalmente tres objetivos:

1. Simplificación: Se trata de reducir los modelos quedándose únicamente con los más necesarios.
2. Unificación: Para permitir la intercambiabilidad a nivel internacional.
3. Especificación: Se persigue evitar errores de identificación creando un lenguaje claro y preciso

Las elevadas sumas de dinero que los países desarrollados invierten en los organismos normalizadores, tanto nacionales como internacionales, es una prueba de la importancia que se da a la normalización.

Hay tres características importantes inherentes a los sistemas de bases de datos: la separación entre los programas de aplicación y los datos, el manejo de múltiples vistas por parte de los usuarios y el uso de un catálogo para almacenar el esquema de la base de datos. En 1975, el comité ANSI-SPARC (American National Standard Institute - Standards Planning and Requirements Committee) propuso una arquitectura de tres niveles para los sistemas de bases de datos. Las arquitecturas de bases de datos han evolucionado mucho desde sus comienzos, sin embargo hoy se considera un estándar la propuesta descrita por dicho comité: ANSI/X3/SPARC, la cual es una arquitectura general para DBMSs basada en tres niveles o esquemas: el nivel físico, o de máquina, el nivel externo, o de usuario, y el nivel conceptual. Así mismo describió las interacciones entre estos tres niveles y todos los elementos que conforman cada uno de ellos [7].

Nivel interno: Tiene un esquema interno que describe la estructura física de almacenamiento de base de datos. Emplea un modelo físico de datos y los únicos datos que existen están realmente en este nivel.

Nivel conceptual: tiene esquema conceptual. Describe la estructura de toda la base de datos para una comunidad de usuarios. Oculta los detalles físicos de almacenamiento y trabaja con elementos lógicos como entidades, atributos y relaciones.

Nivel externo o de vistas: tiene varios esquemas externos o vistas de usuario. Cada esquema describe la visión que tiene de la base de datos a un grupo de usuarios, ocultando el resto.

1.6.1 Normalización en las bases de Datos

El proceso de normalización de bases de datos consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional.

Las bases de datos relacionales se normalizan para:

- Evitar la redundancia de los datos.
- Evitar problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.

En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla sea considerada como una relación tiene que cumplir con algunas restricciones:

- Cada columna debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.

Formas Normales [editar] Las formas normales son aplicadas a las tablas de una base de datos, decir que una base de datos está en la forma normal N es decir que todas sus tablas están en la forma normal N.

En general, las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos. El creador de estas 3 primeras formas normales (o reglas) fue Edgar F. Codd.[1]

- Primera Forma Normal (1FN) [editar]Artículo principal: Primera forma normal

Una tabla está en Primera Forma Normal sólo si

Todos los atributos son atómicos. Un atributo es atómico si los elementos del dominio son indivisibles, mínimos.

1. La tabla contiene una clave primaria
2. La tabla no contiene atributos nulos
3. Eliminar grupos repetitivos y formar una nueva tabla

Una columna no puede tener múltiples valores. Los datos son atómicos. (Si a cada valor de X le pertenece un valor de Y, entonces a cada valor de Y le pertenece un valor de X)....

- Segunda Forma Normal (2FN) [editar]Artículo principal: Segunda forma normal

Dependencia Funcional. Una relación está en 2FN si está en 1FN y si los atributos que no forman parte de ninguna clave dependen de forma completa de la clave principal. Es decir que no existen dependencias parciales. Los atributos dependen de la clave. Varía la clave y varían los atributos.

En otras palabras podríamos decir que la segunda forma normal está basada en el concepto de dependencia completamente funcional. Una dependencia funcional $X \rightarrow Y$ es completamente funcional si al eliminar los atributos A de X significa que la dependencia no es mantenida, esto es que $A \rightarrow X, (X \text{ ó } \{A\}) \rightarrow Y$. Una dependencia funcional $X \rightarrow Y$ es una dependencia parcial si hay algunos atributos A de X que pueden ser removidos de X y la dependencia todavía se mantiene, esto es $A \rightarrow X, (X \text{ ó } \{A\}) \rightarrow Y$. Por ejemplo $\{SSN, PNUMBER\} \rightarrow HOURS$ es completamente dependencia dado que ni $SSN \rightarrow HOURS$ ni $PNUMBER \rightarrow HOURS$ mantienen la dependencia. Sin embargo $\{SSN, PNUMBER\} \rightarrow ENAME$ es parcialmente dependiente dado que $SSN \rightarrow ENAME$ mantiene la dependencia

- Tercera Forma Normal (3FN) [editar]Artículo principal: Tercera forma normal

La tabla se encuentra en 3FN si es 2FN y cada atributo que no forma parte de ninguna clave, depende directamente y no transitivamente, de la clave primaria.

Un ejemplo de este concepto sería que, una dependencia funcional $X \rightarrow Y$ en un esquema de relación R es una dependencia transitiva si hay un conjunto de atributos Z que no es un subconjunto de alguna clave de R, donde se mantiene $X \rightarrow Z$ y $Z \rightarrow Y$. Por ejemplo, la dependencia $SSN \rightarrow DMGRSSN$ es una dependencia transitiva en EMP_DEPT de la siguiente figura. Decimos que la dependencia de DMGRSSN el atributo clave SSN es transitiva vía DNUMBER porque las dependencias $SSN \rightarrow DNUMBER$ y $DNUMBER \rightarrow DMGRSSN$ son mantenidas, y DNUMBER no es un subconjunto de la clave de EMP_DEPT. Intuitivamente, podemos ver que la dependencia de DMGRSSN sobre DNUMBER es indeseable en EMP_DEPT dado que DNUMBER no es una clave de EMP_DEPT.

- Forma Normal de Boyce-Codd (FNBC) [editar]Artículo principal: Forma normal de Boyce-Codd

La tabla se encuentra en BCNF si cada determinante, atributo que determina completamente a otro, es clave candidata.

- Cuarta Forma Normal (4FN) [editar]Artículo principal: Cuarta forma normal

Una tabla se encuentra en 4FN si, y sólo si, para cada una de sus dependencias múltiples no funcionales $X \twoheadrightarrow Y$, siendo X una super-clave que, X es o una clave candidata o un conjunto de claves primarias.

- Quinta Forma Normal (5FN) [editar]Artículo principal: Quinta forma normal

Una tabla se encuentra en 5FN si:

La tabla esta en 4FN

No existen relaciones de dependencias no triviales que no siguen los criterios de las claves. Una tabla que se encuentra en la 4FN se dice que está en la 5FN si, y sólo si, cada relación de dependencia se encuentra definida por las claves candidatas.

1.7 Arquitectura Cliente Servidor

La alta demanda de computadoras personales de bajo costo, los recursos de los sistemas de información existentes en cualquier organización se pueden distribuir entre ordenadores de diferentes tipos: computadoras personales, estaciones de trabajo, mini ordenadores o incluso grandes ordenadores.

La arquitectura cliente - servidor es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes. Los tipos específicos de servidores incluyen los servidores Web, los servidores del uso, los servidores de archivo, los servidores terminales, y los servidores del correo.

En este modelo las aplicaciones se dividen de forma que el servidor contiene la parte que debe ser compartida por varios usuarios, y en el cliente permanece sólo lo particular de cada usuario.

Los clientes realizan generalmente funciones como:

- Manejo de la interfaz de usuario.
- Captura y validación de los datos de entrada.
- Generación de consultas e informes sobre las bases de datos.

Por su parte los servidores realizan, entre otras, las siguientes funciones:

- Gestión de periféricos compartidos.
- Control de accesos concurrentes a bases de datos compartidas.
- Enlaces de comunicaciones con otras redes de área local o extensa.

Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste le responde proporcionándolo. Normalmente, pero no necesariamente, el cliente y el servidor están ubicados en distintos procesadores. Los clientes se suelen situar en ordenadores personales y/o estaciones de trabajo y los servidores en procesadores departamentales o de grupo.

Entre las principales características de la arquitectura cliente/servidor se pueden destacar las siguientes:

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

1.7.1 Ventajas de la arquitectura cliente-servidor

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema.
- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado.
- Posibilita más puestos concurrentes y mayor volumen de registros: La base de datos permite operar con tablas con gran cantidad de registros y mayor tamaño permitiendo muchos más puestos operando al mismo.
- Mayor Robustez: Sólo el servidor modifica los datos según los requerimientos de las estaciones de trabajo. De esta forma se elimina la posibilidad de problemas de integridad de los datos ocasionadas por cortes de energía, desconexiones de la red, fallas del sistema operativo, etc.
- Mejor seguridad y accesibilidad de la información: El servidor de base de datos se puede acceder vía puertos TCP/IP. Por este motivo, no se requiere compartir la carpeta de la base de datos en la red evitando que quede eventualmente visible. Opcionalmente la base puede quedar protegida con usuario y contraseña, brindando mayor seguridad.
- El motor de base de datos no implica costo extra: No hay que pagar licencias propias del motor de base de datos para usarlo, el mismo se incluye con el sistema sin costo adicional.
- Fácil instalación y mantenimiento: No requiere personal especializado para gestionar la base de datos. La misma es de fácil instalación y configuración.

1.8. Lenguajes a Usar

Actualmente existen varios lenguajes para programar una base de datos, que ofrecen interfaces apropiadas para cada tipo de usuario: administradores de la base de datos, diseñadores, programadores de aplicaciones y usuarios finales. Nosotros ocuparemos MySQL y PHP.

1.8.1 MySQL

MySQL es un sistema de administración para bases de datos relacionales que provee una solución robusta con herramientas multi-usuario, provee también soluciones de base de datos SQL (structured Query Language), Es rápido, robusto y fácil de utilizar [10].

MySQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo, cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir que puede hacer y que no puede hacer con el software en diferentes situaciones [9].

Las principales características de este gestor de bases de datos son las siguientes [8]:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc.).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

1.8.2 PHP

Es un lenguaje de programación usado normalmente para la creación de contenido para sitios web con los cuales se puede programar las páginas HTML, significa PHP Hypertext Pre-processor [12].

Los principales usos del PHP son los siguientes [12]:

- Programación de páginas web dinámicas, habitualmente en combinación con el motor de base datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
- Programación en consola, al estilo de Perl o Shell scripting.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y Qt/GTK+, lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado.

Las principales ventajas de PHP son [11]:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL .
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial [1], entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Permite crear los formularios para la web.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Capítulo



2

Análisis de Requerimientos

En la Facultad de Ciencias de la Computación Existe un departamento de vinculación el cual se encarga de contactar a diversas empresas con egresados de esta facultad con el fin de obtener mejores resultados, y mayor funcionalidad por ello, se implementara un sistema capaz de hacer más sencilla esta tarea.

Este sistema se encargara del gestionamiento de currículos para egresados y practicantes de la Facultad de Ciencias de la Computación con el fin de obtener mejores oportunidades de empleo para los alumnos y al mismo tiempo que las empresas puedan obtener a los mejores candidatos en el área de informática.

Por lo anterior a continuación se describen cada uno de los requisitos y necesidades de este sistema.

2.1 Especificaciones

En la Facultad de Ciencias de la Computación el departamento de vinculación, está a cargo de la Maestra en Ciencias Alma Delia Ambrosio Vázquez Catedrática de esta institución, quien lleva el registro de los currículos de egresados y las vacantes requeridas por las empresas todo ello sin ningún sistema que le permita guardar esta información de manera eficaz.

Por lo tanto se requiere de un sistema Web capaz de registrar toda la información de los egresados y las empresas en una Base de Datos.

Además que este sistema permita realizar actualizaciones en la información por parte del administrador.

Así mismo se requiere que el sistema permita a cualquier persona de la FCC consultar esta información e incluso hacer búsquedas.

Por último que se pueda Estandarizar la creación de los currículos para poder obtener una perspectiva más clara y poder colocar a los candidatos de forma más eficiente de acuerdo a sus áreas de interés.

2.2 Requisitos del sistema de acuerdo a las especificaciones del cliente.

2.2.1 Requisitos Funcionales

- Gestionar la información de Egresados , con la cual deben realizar las siguientes actividades:
 - Alta de Currículo
 - Baja de Currículo
 - Actualización de Currículo

- Gestionar la información de Empresas , con la cual deben realizar la siguientes actividades:
 - Alta de Empresa
 - Baja de Empresa
 - Actualización de Empresa
 - Alta de Oferta

- Consultas de Índices, con las siguientes formas de consultas:
 - Por Matricula
 - Por Rango de Edad
 - Por Sexo
 - Por Carrera
 - Por Área de Interés

2.2.2 Requisitos no funcionales

Un sistema que funcione en cualquier computadora con procesador Pentium IV a 2.8 GHz, con 256 Mb en RAM con software de sistema libre.

2.3 Análisis del sistema

El sistema contara con 2 tipos de usuarios que son, usuarios regulares que pueden ser los egresados o empresas, quienes pueden hacer consultas de la información o registrar información, y el usuario administrador que será el único que puede realizar bajas o actualizaciones de la información.

Estará dividido en diferentes módulos para tener una mejor organización de la información.

Entre los módulos principales encontramos:

Modulo de Registro

Este modulo permite realizar el ingreso de información al sistema y se divide en otros sub-módulos que son:

- Gestión de Currículos:
En este apartado los egresados de la Facultad de Ciencias de la Computación registran toda la información necesaria y relevante para poder conseguir una vacante.

- Registro de Empresa
Aquí las diversas empresas pueden registrar sus datos para tener a la mano toda su información y que sea más fácil ofertar sus vacantes

- Gestión de Oferta

En este apartado las empresas una vez registrada pueden ofertar empleos o prácticas profesionales incluyendo los requisitos para los candidatos.

Modulo de Búsqueda

- Búsqueda para Egresaos

En este apartado los egresados pueden realizar búsquedas de las ofertas más recientes que se publiquen con lo cual tendrán una mejor oportunidad de seleccionar lo que les interesa.

- Búsqueda para Empresas

A su vez las Empresas tendrán la capacidad de buscar diferentes atributos o características de los Candidatos para obtener el recurso que necesitan y con ello seleccionar de una manera precisa a los egresados que cumplan con sus expectativas

Modulo de Actualizaciones y Bajas

En este apartado el Administrador del Sistema será capaz de relazar cambios en la información registrada así como también de eliminar la información que considere innecesaria. Los egresados y empresas solo podrán actualizar su propia información.

La figura 1.0 nos muestra de manera visual cada uno de estos módulos y la interacción que existe entre los actores de nuestro sistema.

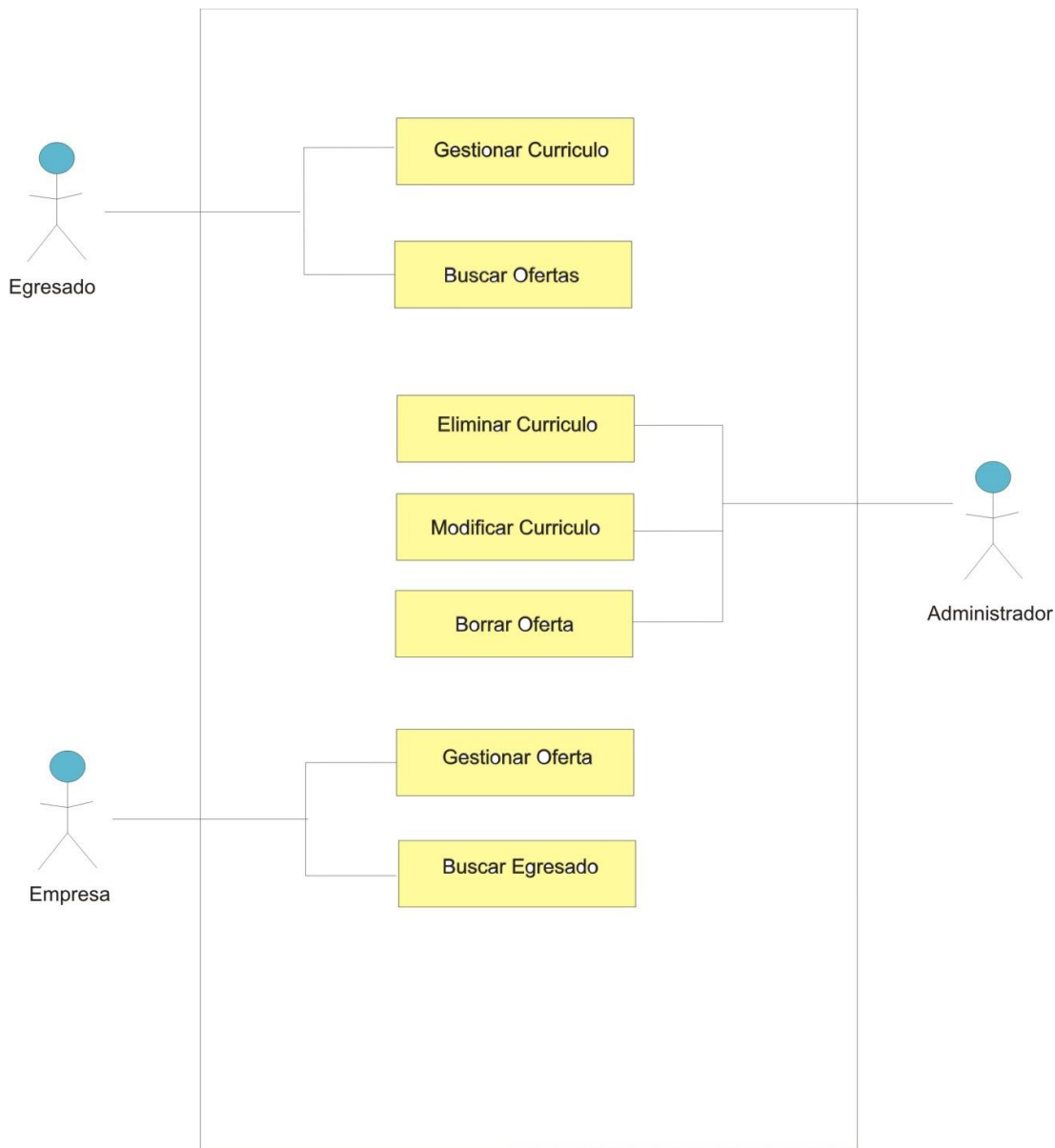


Fig. 1.0 Diagrama General del Sistema SIAC

2.3.1 Gestionar Currículo

Descripción: este modulo es iniciado por el egresado quien tendrá que registrar sus datos dentro del sistema

Flujo de Eventos:

-Flujo Básico: se inicia cuando el usuario accesa al sistema en el menú de registro aparece la opción egresado al entrar en dicho menú se abre automáticamente un Formulario de registro.

El formulario tendrá todos los campos necesarios para registrar la información requerida y algunos catálogos predeterminados para que el usuario seleccione la opción correspondiente.

-Flujo Alternativo: El flujo alternativo será el ingreso incorrecto de la información ya que el formulario de registro cuenta con una validación para los campos más importantes, en caso de que el usuario no ingrese adecuadamente la información se le informara a través de una alerta.

Si el usuario se percata de que algún información es incorrecta podrá tener una opción de cancelar para volver a iniciar el registro las veces que se necesario y hasta que el considere que esta correcta la información.

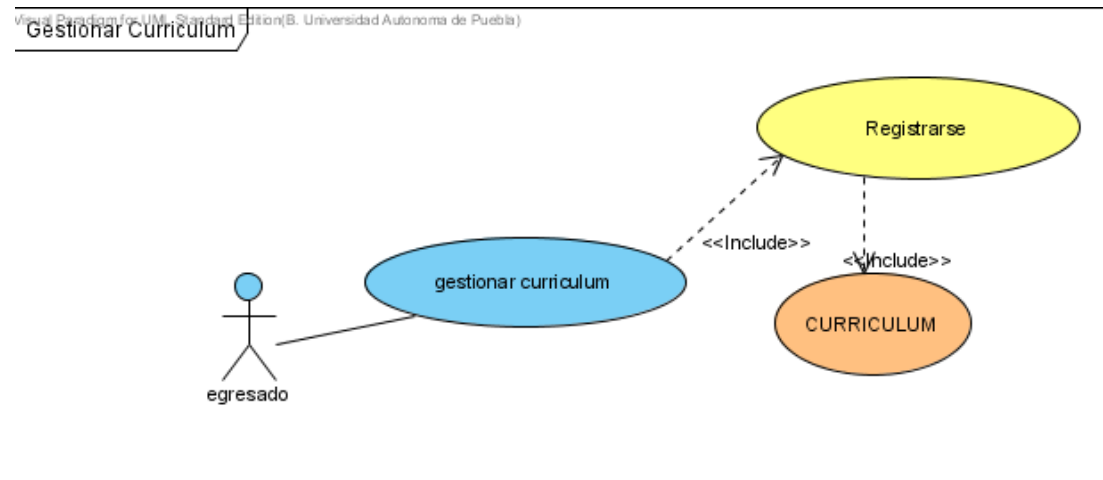


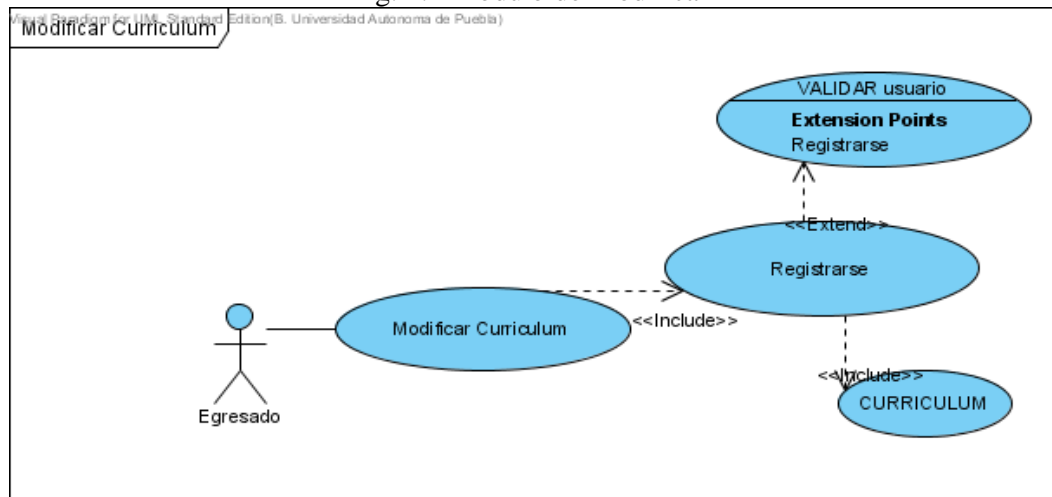
Fig. 1.1 Modulo Gestionar

2.3.2 Modificar Currículo

Flujo Básico se inicia cuando el administrador ingresa al sistema en el apartado de Actualizaciones y le pedirá que se autentifique para poder realizar cambios, de igual forma podrán iniciar este flujo egresados o empresarios para ingresar tendrán que autenticarse también.

Una vez que se realizó la validación se muestra la información que cada uno puede modificar, si es egresado habrá un link a su currículo, si es empresa habrá un link a empresas y ofertas, si es administrador aparecerán todos los links.

Fig. 1.2 Modulo de Modificar



Flujo alternativo si la validación no es correcta se notificara con una serie de alertas dirigidas al usuario para que escriba adecuadamente la información que se requiere.

2.3.3 Eliminar Currículo

Flujo Básico: este proceso inicia cuando el Administrador Accesa al apartado de eliminar donde el sistema le pedirá que se autentifique como administrador, un vez realizado este proceso se desplegar una lista de información que puede seleccionar y eliminar.

Flujo Alternativo: Si el usuario no se valida se desplegaran en pantalla una serie de alertas indicándole que llene correctamente la información de no ser así no tendrá acceso a este modulo.

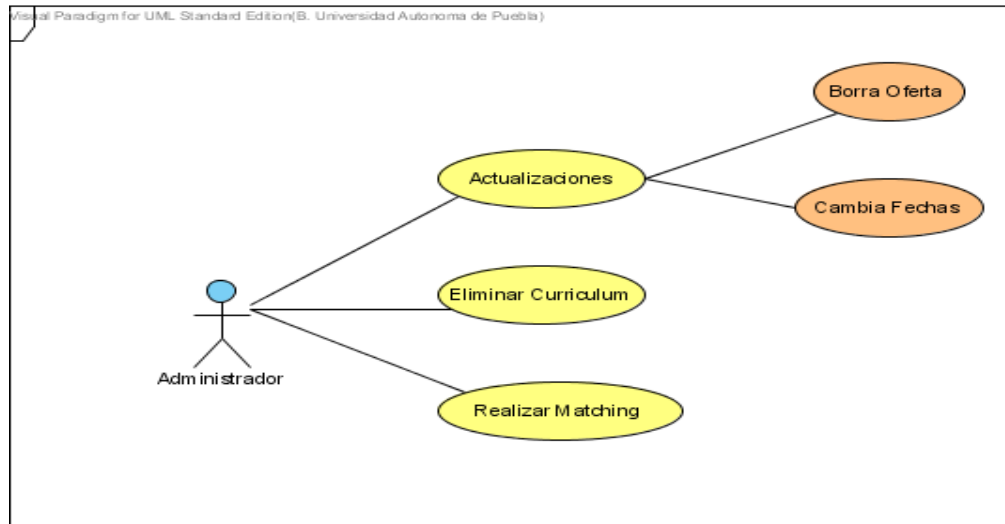


Fig. 1.3 Modulo Eliminar y Actualizaciones

2.3.4 Buscar Oferta

Flujo Básico: Empieza cuando el usuario selecciona la opción de ofertas en el menú principal esta información es visible para todos los usuarios por lo cual ingresando en este apartado se visualizara una lista de ofertas recientes y la información necesaria para contactarse con alguna empresa en caso de requerir mas información será necesario contactar al administrador del sistema.

Flujo Alternativo: Si el usuario ingresa parámetros no validos, o que no estén dentro de la base de datos no encontrara nada el buscador.

2.3.5 Buscar Egresado

Flujo Básico: Inicia Cuando el usuario ingresa al sistema en la sección de búsqueda personalizada donde encontrara un formulario que debe llenar para especificar los parámetros de su búsqueda una vez seleccionados los campos que el creé conveniente se realizara la búsqueda que puede variar de acuerdo a las especificaciones dadas por el usuario.

Si el usuario es muy especifico en su búsqueda lo más probable es que disminuya el numero de coincidencias, en cambio si es muy general puede ser que su búsqueda se haga más grande el numero de coincidencias.

Flujo Alternativo: Si el usuario ingresa parámetros no validos, o que no estén dentro de la base de datos no encontrara nada el buscador.

2.3.6 Gestionar Oferta

Flujo Básico: Inicia Cuando el usuario entra en el menú de registro y ahí selecciona ofertas, parecerá un formulario que debe llenar con los datos necesarios para que se publique su oferta.

Estos datos se validan y se guardan dentro de la base de datos para que estén disponibles cuando se consulten.

Flujo Alternativo: Si el Usuario no ingresa los datos necesarios no se podrá publicar la oferta.

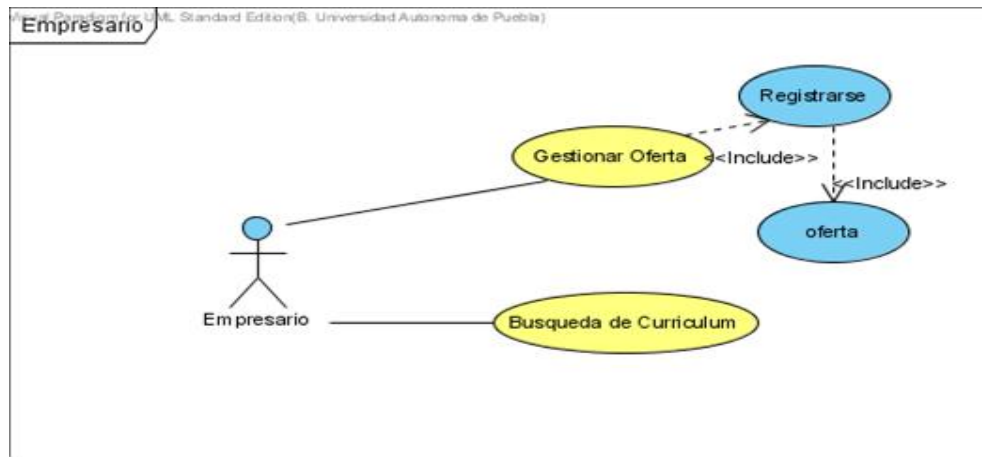


Fig. 1.4 Módulos en los que interviene el Actor Empresa

2.3.7 Registrar Empresa

Flujo Básico: Este inicia cuando el usuario ingresa al sistema en el apartado de registro seleccionando la opción Empresa, Aparecerá un formulario pidiendo al usuario registre a su empresa, muchos de los datos serán indispensables para llevar a cabo el registro y terminara cuando el usuario obtenga su nombre de usuario y contraseña.

Flujo Alternativo: que el Usuario no registre la información necesaria para llevar a cabo el registro entonces no se permitirá el acceso de su información.

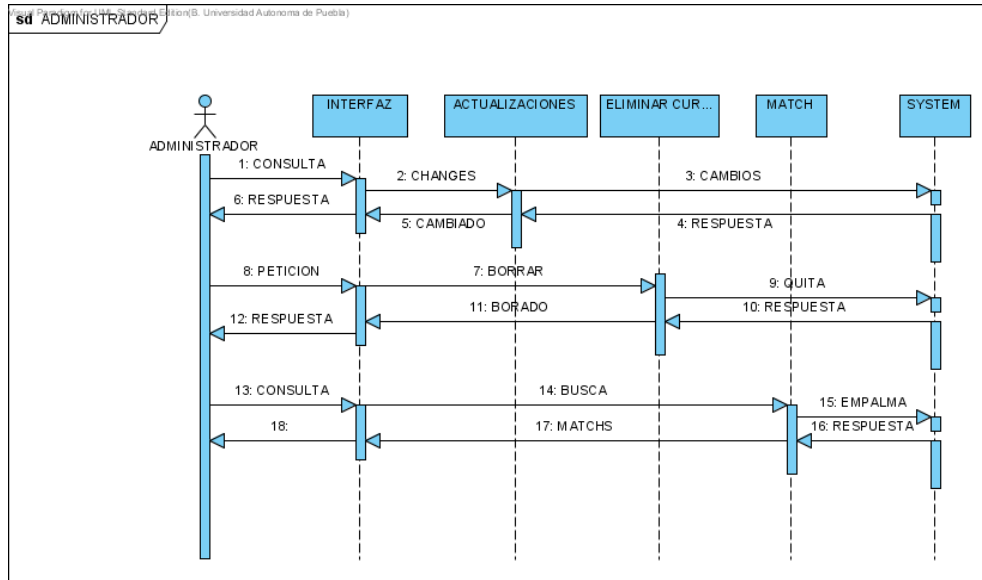


Fig. 1.5 Diagrama de Secuencial donde se muestra la interacción del Administrador

Como se puede observar, el análisis y las especificaciones del sistema fue descrito en un Lenguaje Unificado de Modelos (UML), esto debido a que actualmente, cualquiera que sea el modelo en el que se va a desarrollar un software, toda persona, sin importar el área en el que se desenvuelva, entienda las especificaciones y los procedimientos que se siguieron para llevar a cabo dicho desarrollo.

Ya con el análisis del sistema, se obtuvieron las especificaciones necesarias del mismo, por lo que en el siguiente capítulo trata ya del diseño e implementación de la Base de datos.

Capítulo



3

Diseño

Después de haber realizado la etapa de análisis y especificación de requisitos del sistema, se procedió a diseñar la base de datos siguiendo las fases principales que la conforman: Diseño conceptual, diseño lógico y diseño físico.

3.1 Diseño Conceptual

El primer paso en el diseño de una base de datos es la producción del esquema conceptual. Normalmente, se construyen varios esquemas conceptuales, cada uno para representar las distintas visiones que los usuarios tienen de la información. La figura 3.0 muestra una representación general del sistema.

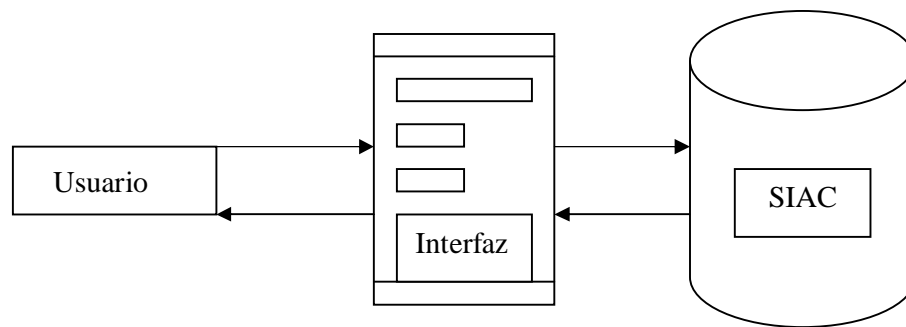


Fig. 3.0 representación general del sistema

Las tareas a realizar en el diseño conceptual son las siguientes:

1. Identificar las entidades.
2. Identificar las relaciones.
3. Identificar los atributos y asociarlos a entidades y relaciones.
4. Determinar los dominios de los atributos.
5. Determinar los identificadores.
6. Determinar las jerarquías de generalización (si las hay).
7. Dibujar el diagrama entidad-relación.
8. Revisar el esquema conceptual local con el usuario.

3.1.1 Identificar las entidades.

Una entidad se define como un concepto del mundo real, de modo que nuestras bases de datos guardan información sobre entidades. Estas entidades pueden ser de diferente carácter:

1. Entidades físicas: un libro, una bebida, un empleado
2. Entidades conceptuales: una empresa
3. Entidades como eventos: una alerta de nuestra agenda que nos recuerda una tarea.

Uno de los pasos de nuestro proceso de planificación es detectar estas entidades que están relacionadas con la base de datos.

La integridad de entidad pretende que cada entidad que se guarda en la base de datos sea identificable de un modo único, es decir, que evitemos la información redundante.

Lo más razonable es que se identifique como identidad aquellas cosas con las que vas a trabajar de modo unitario. Dicho de un modo más claro, la información que se almacena unida (de modo unitario) es más cómodo trabajar con ella, o recuperar esa información en una única operación.

En primer lugar hay que definir los principales objetos que interesan al usuario. Estos objetos serán las entidades. Una forma de identificar las entidades es examinar las especificaciones de requisitos de usuario. En estas especificaciones se buscan los nombres o los sintagmas nominales que se mencionan (por ejemplo: número de empleado, nombre de empleado, número de inmueble, dirección del inmueble, alquiler, número de habitaciones). También se buscan objetos importantes como personas, lugares o conceptos de interés, excluyendo aquellos nombres que sólo son propiedades de otros objetos. Por ejemplo, se pueden agrupar el número de empleado y el nombre de empleado en una entidad denominada *empleado*, y agrupar número de inmueble, dirección del inmueble, alquiler y número de habitaciones en otra entidad denominada *inmueble*.

Para el Sistema Integral para la Administración de Currículos se Identifican las siguientes Entidades:

- Egresado : Representa a los Alumnos egresados de la Facultad de Ciencias de la Computación y queda representada en la siguiente Figura



EGRESADO

Fig. 3.1 Entidad Egresado

- Empresa: Representa las Empresas que buscan Candidatos



EMPRESA

Fig. 3.2 Entidad Empresa

- Oferta: Representa las Ofertas gestionadas por las Empresas esta queda representada en la Fig. 3.3

OFERTAS

Fig. 3.3 Entidad Ofertas

- Áreas de Interés: representa las Áreas de Interés que tienen los egresados de la FCC se representa mediante la Fig. 3.4

Áreas de
Interés

Fig. 3.4 Entidad Áreas de Interés

- Estudios: Representa los estudios que han realizado los egresados ya sea licenciatura, ingeniería, o algún postgrado se representa como sigue.

ESTUDIOS

Fig. 3.5 Entidad Estudios

- Estado: Representa el estado de la república de procedencia de los egresados mediante la sig. figura.

ESTADO

Fig. 3.6 Entidad Estado

- Experiencia: Representa las experiencias de trabajo que tengan los egresados. Se ilustra en la Fig. 3.7

Experiencia

Fig. 3.7 Entidad Experiencia

- TIPO OFERTAS: Representa el tipo de oferta que se ofrece ya sea práctica profesional u Oferta Laboral. Se ilustra en la fig. 3.8

TIPO OFERTAS

Fig. 3.8 Entidad Tipo Ofertas

- ADMINISTRADOR: Representa al usuario que llevara el control del sistema y guarda la contraseña del mismo. Se ilustra en la Fig. 3.9



Fig. 3.9 Entidad Administrador

- USUARIOS: Representa a los usuarios del sistema y guarda sus contraseñas se ilustra en la Fig. 3.10



Fig. 3.10 Entidad Usuarios

3.1.2 Identificar las relaciones

Una vez definidas las entidades, se deben definir las relaciones existentes entre ellas. Del mismo modo que para identificar las entidades se buscaban nombres en las especificaciones de requisitos, para identificar las relaciones se suelen buscar las expresiones verbales. Pero sólo interesan las relaciones que son necesarias. Una vez identificadas todas las relaciones, hay que determinar la cardinalidad mínima y máxima con la que participa cada entidad en cada una de ellas. Conforme se van identificando las relaciones, se les van asignando nombres que tengan significado para el usuario.

Para Nuestro Sistema SIAC (Sistema Integral Para la Administración de Currículos) las relaciones encontradas son las siguientes:

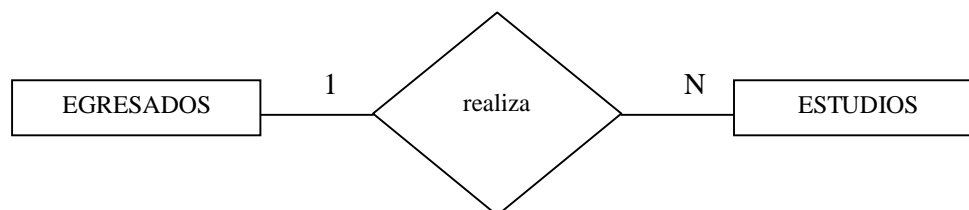


Fig. 3.11 Relación 1

La relación 1 de la Figura 3.12 muestra la relación que hay entre la entidad EGRESADOS y la entidad ESTUDIOS, la cual queda enunciada de la forma siguiente: Un egresado debe realizar uno o más estudios y uno o más estudios debe ser realizado por un egresado.

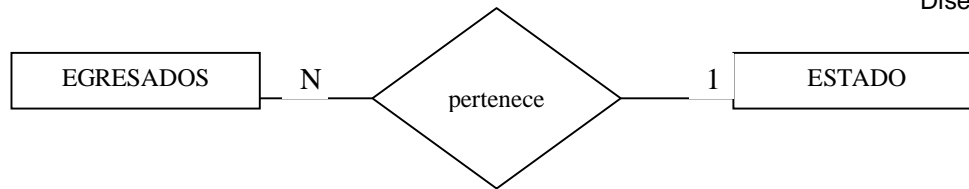


Fig. 3.12 Relación 2

La relación 2 de la Figura 3.13 muestra la relación que hay entre la entidad EGRESADOS y la entidad ESTADO, la cual queda enunciada de la forma siguiente: uno o más egresados deben pertenecer a un estado y un Estado puede ser la procedencia de uno o más egresados.

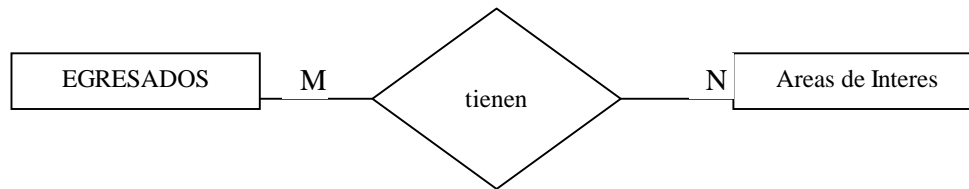


Fig. 3.13 Relación 3

La relación 3 de la Figura 3.14 muestra la relación que hay entre la entidad EGRESADOS y la entidad Áreas de Interés, la cual queda enunciada de la forma siguiente: un egresado puede tener una o más áreas de interés y un área de interés debe pertenecer a uno o más egresados.

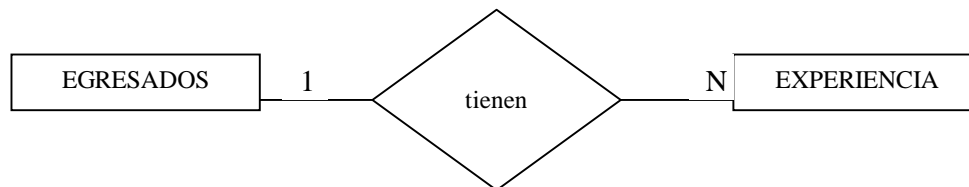


Fig. 3.14 Relación 4

La relación 4 de la Figura 3.15 muestra la relación que hay entre la entidad EGRESADOS y la entidad EXPERIENCIA, la cual queda enunciada de la forma siguiente: un Egresado puede tener una o más experiencias una o más experiencias deben pertenecen a un egresado.



Fig. 3.15 Relación 5

La relación 5 de la Figura 3.16 muestra la relación que hay entre la entidad EGRESADOS y la entidad OFERTAS, la cual queda enunciada de la forma siguiente: uno o más egresados pueden interesarse en una o más ofertas y una o más ofertas pueden ser atendidas por uno o más egresados.

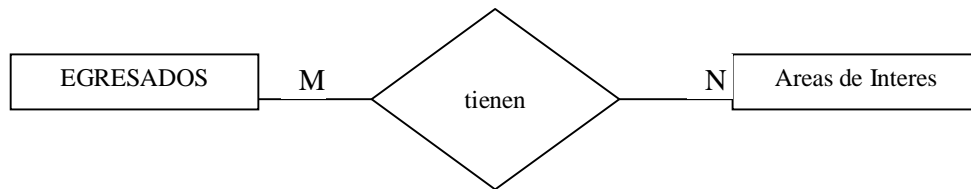


Fig. 3.16 Relación 4

La relación 2 de la Figura 3.12 muestra la relación que hay entre la entidad EGRESADOS y la entidad ESTADO, la cual queda enunciada de la forma siguiente: Varios egresados deben pertenecer a un estado y un Estado debe ser la procedencia de varios estudiantes.

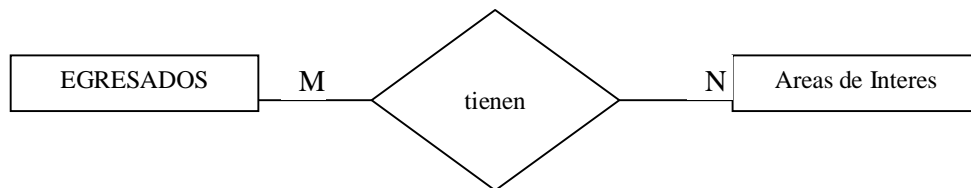


Fig. 3.17 Relación 4

3.1.3 Identificar los atributos y asociarlos a entidades y relaciones.

En las entidades como en las relaciones, se tienen nombres que especifiquen los requisitos que las describen, a esto se les llama atributos, estos atributos identifican propiedades, cualidades, identificadores o características de entidades o relaciones.

También se deben identificar los atributos derivados o calculados, que son aquellos cuyo valor se puede calcular a partir de los valores de otros atributos. El atributo derivado se considera en el diseño físico.

Cuando se están identificando los atributos, se puede descubrir alguna entidad que no se ha identificado previamente, por lo que hay que volver al principio introduciendo esta entidad y viendo si se relaciona con otras entidades.

Conforme se van identificando los atributos, se les asignan nombres que tengan significado para el usuario. De cada atributo se debe anotar la siguiente información:

- Nombre y descripción del atributo.
- Alias o sinónimos por los que se conoce al atributo.
- Tipo de dato y longitud.
- Valores por defecto del atributo (si se especifican).
- Si el atributo siempre va a tener un valor (si admite o no nulos).
- Si el atributo es compuesto y, en su caso, qué atributos simples lo forman.
- Si el atributo es derivado y, en su caso, cómo se calcula su valor.
- Si el atributo es multievaluado.

3.1.3.1 Nombre y descripción del atributo

De acuerdo a nuestras entidades, por orden alfabético, obtuvimos los siguientes atributos:

➤ AREAS DE INTERES

- Id_Area : es un identificador serial de las áreas de interés
- nomarea: es el nombre del área de interés.

➤ EGRESADOS

- Id_Area : es un identificador serial de las áreas de interés
- Matricula: es el número de matrícula del egresado
- Ap_pat: es el apellido paterno
- Nombre: es el nombre del egresado
- Sexo: es el sexo del egresado
- Calle: es la calle de la dirección de el egresado
- Num_int: es el numero del domicilio
- Num_ext: es el numero exterior del domicilio
- Col : es la colonia del domicilio
- Cp : es el código postal del domicilio
- Lugnac : es el lugar de nacimiento del egresado
- Nación: es el país de el egresado
- Fechanac: es la fecha de nacimiento del egresado
- Peso: es el peso en kilogramos del interesado
- Estatura: es la estatura en metros del interesado
- Edocivil: es el estado civil del egresado
- Telcas : es el teléfono de casa
- Telcel: es el teléfono celular
- Email: es la dirección de correo electrónico del interesado
- Email2: es una dirección de correo alternativa
- Curp: es la clave única de registro de población del egresado
- Afore: es el numero de afore
- Sar: es el numero de SAR
- Rfc: es el registro federal de contribuyentes del egresado
- Imss: es el número de seguro social del interesado
- No.cartilla: es el numero de cartilla militar del interesado
- No.pasaporte: es el numero de pasaporte del interesado
- No.cedula es el numero de cedula profesional del egresado.

➤ EMPRESA

- Idemp: es el número serial que identifica a la empresa.
- Nombremp: es el nombre de la empresa
- Razón social: es el nombre de la persona que representa la empresa
- Calle : es la calle del domicilio de la empresa
- Num_int : es el numero interior del domicilio
- Num_ext: es el numero exterior del domicilio
- Col: es el nombre de la colonia del domicilio
- Cp: es el código postal del domicilio
- Tel : es el número telefónico de la empresa
- NoExt: es el numero de extensión del teléfono de la empresa
- Tel2: es un número telefónico alternativo de la empresa
- Email: es la dirección de correo electrónico de la empresa.
- Email2: es una dirección alternativa de correo electrónico
- RFC: es el registro federal de contribuyentes de la empresa

➤ ESTADO

- Id_edo: es el numero serial de cada estado de la república
- Nomedo: es el nombre del estado

➤ ESTUDIOS

- Id_est: es el numero serial de cada estudio realizado por el egresado
- Fecha_ini: es la fecha en que se inicio un estudio.
- Fecha_fin: es la fecha que concluyo sus estudios.
- Titulo_obtenido: es el documento o titulo recibido del estudio correspondiente.
- Tipo: es el tipo de estudio realizado.

➤ EXPERIENCIA

- Id_exp: es el número serial de cada experiencia o trabajo anterior.
- Fecha_ini: es la fecha en que empezó a laborar
- Fecha_fin. Es la fecha en que termino de laborar
- Nom_empresa: es el nombre de la empresa donde laboro
- Domicilio: es el domicilio de la empresa donde laboro
- Puesto: es el puesto que desempeñaba en dicho lugar
- Actividades: es la descripción de las actividades que realizaba
- Tel: es el teléfono de la empresa o patrón.
- Email: es la dirección de correo de la empresa o patrón.

➤ INSTITUCION

- Id_inst: es el numero serial de cada institución a la que asistió o asiste el interesado
- Nombre: nombre de la institución donde realizo estudios el egresado
- Calle: es el nombre de la calle del domicilio de la institución
- Numext: es el numero exterior del domicilio de la institución
- Numint es el numero interior del domicilio de la institución
- Tel: es el teléfono de la institución
- Ciudad. Es el nombre da la ciudad donde se ubica la institución
- Es el país donde se encuentra la institución
- Email: es la dirección de correo de la institución

➤ OBSERVACIONES

- Id_Obs: es el numero serial de cada observación o comentario
- Comentarios: es la descripción o el comentario en sí que se quiere agregar

➤ OFERTAS

- Idof: es el numero serial de cada oferta de trabajo
- Fechasol: es la fecha en que se solicitan las vacantes
- Perfil: es el tipo de elemento solicitado
- Requisitosadm: son los requisitos para la contratación
- Areaaplicacion : es el área de aplicación en la que se desempeñara el trabajador
- Horario: es el horario o jornada laboral
- Dispviajar: es el requisito de disponibilidad de viajar
- Idioma1: es el requisito de hablar una lengua extranjera
- Idioma2: es el requisito de hablar una segunda lengua extranjera
- Idioma 3: es el requisito de hablar una tercera lengua extranjera
- Salario: es el pago que ofrece la empresa
- Prestaciones: son las prestaciones que ofrece la empresa
- Fechaentrevista: es la fecha en que se deberá presentar el candidato a entrevista
- Fechainlab: es la fecha en la que se presen tara a trabajar el empleado
- Catidadvacante: es el numero de lugares disponibles para cada empleo

➤ PETICIONES

- Idemp: es el número serial que identifica a cada empresa.
- Matricula: es el número que identifica a cada egresado.

➤ TIPO OFERTAS

- Idtipo. Es el numero serial que identifica a cada tipo de oferta
- Desctipo: es la descripción de que tipo es la oferta trabajo o practicas

3.1.4 Determinar los dominios de los atributos

El dominio de un atributo es el conjunto de valores que puede tomar el atributo. Un esquema conceptual está completo si incluye los dominios de cada atributo: los valores permitidos para cada atributo, su tamaño y su formato.

3.1.5 Determinar los identificadores

Cada entidad tiene al menos un identificador. En este paso, se trata de encontrar todos los identificadores de cada una de las entidades. Los identificadores pueden ser simples o compuestos. De cada entidad se escogerá uno de los identificadores como clave primaria en la fase del diseño lógico.

3.1.6 Determinar las jerarquías de generalización

En este paso hay que observar las entidades que se han identificado hasta el momento. Hay que ver si es necesario reflejar las diferencias entre distintas ocurrencias de una entidad, con lo que surgirán nuevas subentidades de esta entidad genérica; o bien, si hay entidades que tienen características en común y que realmente son subentidades de una nueva entidad genérica.

3.1.7 Dibujar el diagrama entidad-relación

Una vez identificados todos los conceptos, se puede dibujar el diagrama entidad-relación correspondiente a una de las vistas de los usuarios. Se obtiene así un esquema conceptual local.



Fig. 3.18 Diagrama Entidad Relación

3.1.8 Revisar el esquema conceptual local con el usuario

El proceso de revisar el esquema conceptual con el usuario, es en rasgos generales, mostrar el análisis realizado con explicación del significado de cada uno de los pasos realizados para que pueda comprender que hasta esta etapa del proyecto, si bien no se tiene un producto terminado, se tiene ya gran parte de toda la parte intangible del sistema.

3.2 Diseño Lógico

El objetivo del diseño lógico es convertir el esquema conceptual en un esquema lógico que se ajuste al modelo de SGBD sobre el que se vaya a implementar el sistema, así como la de obtener una representación que use los recursos que el modelo de SGBD posee para estructurar los datos y para modelar las restricciones. En esta etapa se validan las entidades y sus relaciones mediante la normalización, para que una base de datos diseñada esté normalizada debe cumplir con al menos las primeras cuatro formas normales, las cuales mencionamos a continuación:

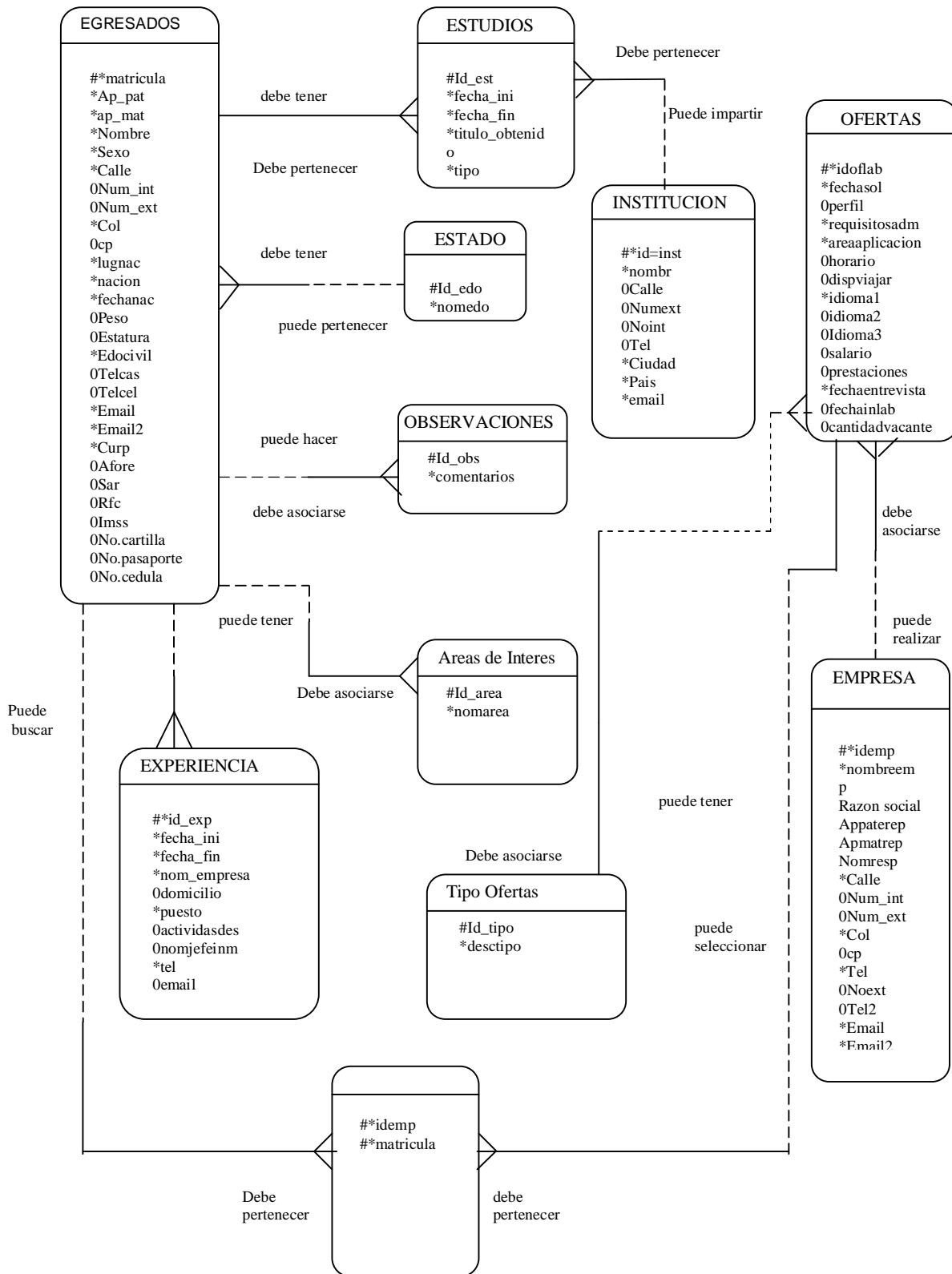
1ra. Una dupla o relación está en primera forma normal si, y sólo si, todos los dominios de la misma contienen valores atómicos, generalmente cuando no se tienen datos de la entidad o bien sólo se tiene el modelo entidad-relación de la base de datos, se asume que ésta está en 1ra. Forma normal.

2da. Una dupla o relación está en segunda forma normal si, y sólo si, está en 1FN (1ra. Forma Normal) y, además, cada atributo no clave es completamente dependiente de la clave primaria.

3ra. Una dupla o relación está en tercera forma normal si, y sólo si, está en 2FN (2da. Forma Normal) y, además, cada atributo no clave no depende de otro atributo no clave dentro de la misma entidad o relación, es decir, no debe existir transitividad entre los atributos.

4ta. De de Óbice-Codd (BCFN). Una dupla o relación está en la forma normal de Boyce-Codd si, y sólo si, todo determinante es una clave candidata. Es decir, si tiene dos o más claves candidatas compuestas que tienen al menos un atributo en común.

FIG 3.21 Diagrama del Diseño Lógico

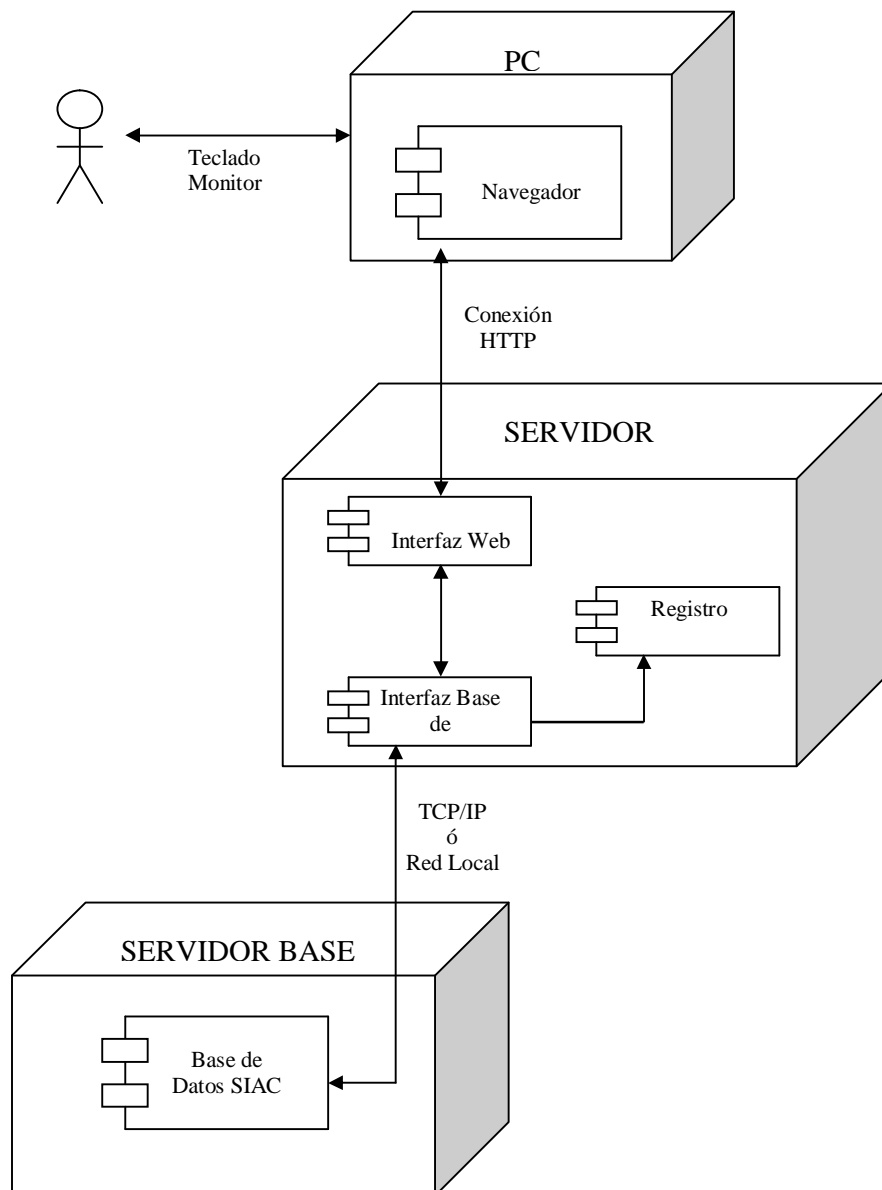


3.3 Diseño Físico

El diseño físico es el proceso de producir la descripción de la implementación de la base de datos en memoria secundaria: estructuras de almacenamiento y métodos de acceso que garanticen un acceso eficiente a los datos.

En general, el propósito del diseño físico es describir cómo se va a implementar físicamente el esquema lógico obtenido en la fase anterior. Concretamente, en el modelo relacional, esto consiste en:

- Obtener un conjunto de relaciones (tablas) y las restricciones que se deben cumplir sobre ellas.
- Determinar las estructuras de almacenamiento y los métodos de acceso que se van a utilizar para conseguir unas prestaciones óptimas.
- Diseñar el modelo de seguridad del sistema.



3.4 Normalización

3.4.1 Diagrama de dependencia Funcional

EGRESADOS

matricula → Ap_pat, ap_mat, Nombre, Sexo, Calle, Num_int, Num_ext, Col, cp, lugnac, nacion, fechanac, Peso, Estatura, Edocivil, Telcas, Telcel, Email, Email2, Curp, Afore, Sar, Rfc, I mss, No.cartilla, No.pasaporte, No.cedula

ESTUDIOS

Id_est → fecha_ini, fecha_fin, titulo_obtenido, tipo.

ESTADO

Id_edo → nomedo

OBSERVACIONES

Id_obs → comentarios

EXPERIENCIA

Id_exp → Fecha_ini, Fecha_fin, nom_empresa, domicilio, puesto, actividades, nomjefe, tel, email

INSTITUCION

Id_inst → nombre, calle, numext, numint, tel, ciudad, pais, email

Áreas de Interés

Id_area → nomarea

OFERTAS

Idoflab → fechasol, perfil, requisitosadm, areaaplicacion, horario, dispviajar, idioma1, idioma2, idioma3, salario, prestaciones, fechaentrevista, fechainlab, catidadvacante

EMPRESA

Idemp → nombresp, razonsocial, appaterno, apmater, nomresp, calle, numint, numext, col, cp, tel, noext, tel2, email, email2, rfc

PETICIONES

Idemp, matricula →

3.4.2 Formas Normales

1. Comprobación de la primera forma Normal: Todos los campos no-clave son totalmente Dependientes de la clave primaria.
2. Comprobación de la segunda forma normal: No hay atributos funcionalmente Dependientes de otras no - claves.
3. Comprobación de la tercera forma normal: Ninguna no-clave es transitivamente Dependiente de otras no-claves.
4. Forma de Boyce/Codd: Cada determinante es clave aspirante (de hecho, cada Determinante es llave primaria).

3.5 Diccionario de Datos

Entidad Egresados

Objetivo: Tabla en la cual se guardan los datos del egresado, es la información más importante ya que con los datos registrados se puede obtener la mayor parte de la información

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
matricula	Es la clave del alumno	Entero	8	No
Ap_pat	Apellido paterno	Cadena	15	No
Ap_mat	Apellido materno	Cadena	15	No
Nombre	Nombre del alumno	Cadena	20	No
sexo	Sexo	Cadena	10	No
Calle	Calle del domicilio	Cadena	30	No
Num_int	Numero interior	Entero	5	Si
Col	Colonia del egresado	Cadena	20	No
Cp	Código postal	Entero	5	Si
Lugnac	Lugar de nacimiento	Cadena	15	No
Nación	Nacionalidad	Cadena	15	No
Fechanac	Fecha de nacimiento	Cadena	10	No
Peso	Peso del egresado	Cadena	10	Si
Estatura	Estatura del egresado	Cadena	10	Si
Edocivil	Estado civil	Cadena	10	No
Telcas	Teléfono del egresado	Entero	10	Si
Telcel	Teléfono celular	Entero	10	Si
Email	Correo electrónico	Cadena	30	No
Curp	Curp del egresado	Cadena	15	No
Afore	Numero de afore	Cadena	15	No
Sar	Numero de sar	Cadena	15	Si
Rfc	RFC del egresado	Cadena	15	Si
Imss	Número de seguro social	Cadena	15	Si
No.cartilla	Numero de cartilla	Cadena	15	Si
No.pasaporte	Numero de pasaporte	Cadena	15	Si
No.cedula	Numero de cedula prof.	Cadena	15	Si

Entidad Estado

Contiene la lista de estados de la república mexicana para que el egresado registre su estado de procedencia.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_Estado	Es el identificador del estado	Entero	8	No
Nomedo	Es el nombre del estado	Cadena	15	No

ENTIDAD OBSERVACIONES

Contiene los comentarios agregados por los egresados acerca de algún aspecto en particular que pueda ser tomado en cuenta para su selección.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_obs	Identificador del comentario	Entero	5	No
Comentario	Es el comentario	Cadena	200	no

ENTIDAD EXPERIENCIA

Contiene los datos de trabajos anteriores del egresado es importante para las referencias que las empresas consideren necesarias.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_exp	Es el identificador de la experiencia	Entero	8	No
Fecha_ini	Es la fecha en que inicio	Cadena	10	No
Fecha_fin	Es la fecha en que termino	Cadena	10	No
Nom_empresa	Es el nombre de la empresa	Cadena	20	No
Domicilio	Domicilio del trabajo	Cadena	30	Si
Puesto	El puesto que tenia	Cadena	20	No
Actividades	Lo que realizaba	Cadena	50	Si
Tel	El teléfono del trabajo	Entero	10	No
Email	Correo del trabajo	cadena	30	Si

ENTIDAD INSTITUCION

Contiene la información de las escuelas donde realizo sus estudios el egresado de manera que se pueda contactar en caso de ser necesario.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_inst	Es el identificador de la escuela	Entero	8	No
Nombre	Nombre de la escuela	Cadena	30	No
Calle	Calle del domicilio	Cadena	30	Si
Numext	Numero exterior	Entero	8	Si
Numint	Numero interior	Entero	5	Si
Tel	Teléfono de la escuela	Entero	10	Si
Ciudad	Ciudad	Cadena	20	No
Pais	País de la escuela	Cadena	20	No
Email	Correo electrónico	Cadena	30	No

ENTIDAD AREAS DE INTERES

Contiene las Áreas de Interés o de desarrollo del egresado con el fin de ubicar adecuadamente al egresado de acuerdo a su perfil.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_area	Es el identificador del área	Entero	8	No
Nomarea	Es el nombre del área	Cadena	30	No

ENTIDAD ESTUDIOS

Contiene la información de los estudios realizados por el egresado, en que fechas los realizó y que documento obtuvo.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Id_est	Es el identificador del estudio	Entero	8	No
Fecha_ini	Es la fecha en que inicio	Cadena	10	No
Fecha_fin	Es la fecha en que termino	Cadena	10	No
Titulo_obtenido	Es el título que obtuvo	Cadena	20	No
tipo	Es el tipo de estudio	Cadena	30	No

ENTIDAD EMPRESA

Es una tabla que contiene la información de cada empresa, muy importante ya que se registran los datos relevantes para poder publicar ofertas hacia los egresados.

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
idemp	Es la clave de la empresa	Entero	8	No
Nombremo	Es el nombre de la empresa	Cadena	30	No
Razon social	Nombre del responsable	Cadena	15	No
Apppat	Apellido paterno	Cadena	20	No
Apmat	Apellido materno	Cadena	10	No
Calle	Calle del domicilio	Cadena	30	No
Num_int	Numero interior	Entero	5	Si
Num_ext	Numero exterior	Entero	8	No
Col	Colonia	Cadena	30	Si
Cp	Código postal	Entero	10	No
Tel	Teléfono	Cadena	15	No
Noext	Extensión	Cadena	10	Si
Tel2	Teléfono alterno	Cadena	10	Si
Email	Correo electrónico	Cadena	30	No
Email2	Correo alterno	Cadena	30	Si
RFC	RFC	cadena	20	Si

ENTIDAD OFERTAS

Nombre	Descripción del Campo	Tipo	Tamaño	Nulo
Idoflab	Clave de la oferta	entero	8	no
Fechasol	Fecha de publicación	cadena	20	no
Perfil	El perfil requerido	cadena	20	si
Requisitos	requisitos	cadena	40	no
Areaaplicacion	Área de aplicación	cadena	30	no
Horario	Horario	Cadena	20	si
Dispviajar	Disponibilidad de viajar	Cadena	20	Si
Idioma	Lengua extranjera	Cadena	20	No
Idioma1	Lengua alternativa 1	Cadena	15	Si
Idioma2	Lengua alternativa 2	Cadena	15	Si
Idioma3	Lengua alternativa 3	Cadena	15	Si
Salario	Salario que se oferta	Cadena	15	Si
Prestaciones	Prestaciones que se ofertan	Cadena	40	Si
Fecha entrevista	Fecha de la Entrevista	Cadena	10	No
Fechainlab	Fecha para iniciar	Cadena	10	Si
Cantvacantes	Cantidad de vacantes	Entero	8	Si

FIG. 3.20 Diagrama de Despliegue donde se pueden observar la interacción de componentes físicos del sistema

3.6 Tablas

Después de realizar el proceso de diseño lógico el paso siguiente es la creación de la base de Datos para lo cual utilizamos el manejador MySQL antes de pasar a la implementación del sistema mostraremos como se creó la base de datos y como quedaron creadas las tablas que almacenaran la información para tener un perspectiva general de nuestra Base de Datos.

La siguiente figura Muestra la base de Datos SIAC en el phpMyAdmin, y cada una de sus entidades que ahora podemos ver como tablas.

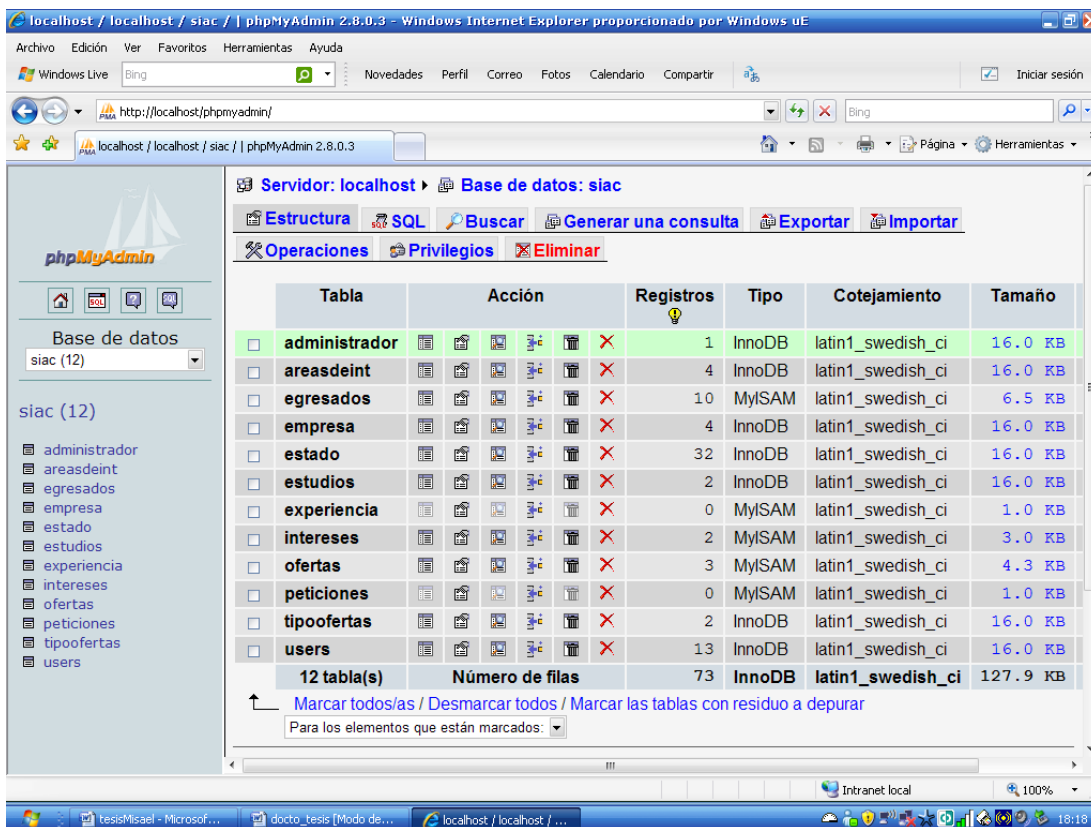


Fig. 3.22 Entidades

La Figura muestra la organización de la tabla egresados y todos sus atributos

Fig.3.23 Tabla Egresados

Server: localhost | Database: siac | Table: egresados

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> matricula	int(20)		UNSIGNED	No		
<input type="checkbox"/> apellidop	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> apellidom	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> Nombre	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> edad	tinyint(4)			No		
<input type="checkbox"/> sexo	varchar(1)	latin1_swedish_ci		No		
<input type="checkbox"/> calle	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> numint	int(6)			Si	NULL	
<input type="checkbox"/> numext	int(6)			Si	NULL	
<input type="checkbox"/> colonia	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> cp	int(6)			Si	NULL	
<input type="checkbox"/> lugarnac	varchar(15)	latin1_swedish_ci		No		
<input type="checkbox"/> nacionalidad	varchar(15)	latin1_swedish_ci		No		
<input type="checkbox"/> fechanac	varchar(11)	latin1_swedish_ci		Si	NULL	
<input type="checkbox"/> peso	varchar(4)	latin1_swedish_ci		Si	NULL	
<input type="checkbox"/> estatura	varchar(5)	latin1_swedish_ci		Si	NULL	

La siguiente figura muestra la tabla áreas de Interés y sus atributos

Server: localhost | Database: siac | Table: areasdeint "InnoDB free: 3072 kB"

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idarea	int(3)			No		auto_increment
<input type="checkbox"/> descareaa	varchar(20)	latin1_swedish_ci		No		

Índices: 0

Nombre de la clave	Tipo	Cardinalidad	Acción	Campo	Tipo	Uso
PRIMARY	PRIMARY	4		idarea	Datos	1,638,4 Bytes
					Índice	0 Bytes
					Total	1,638,4 Bytes

Estadísticas de la fila

Enunciado	Valor
Formato	Compact
Cotejamiento	latin1_swedish_ci
Próxima Autoindex	

Fig. 3.24 Tabla Intereses

La siguiente figura muestra la entidad ofertas y sus atributos

Fig. 3.25 Tabla Ofertas

Base de datos: siac (12)

Tabla: ofertas

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idoferta	int(5)			No		auto_increme
<input type="checkbox"/> fechasol	varchar(11)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> perfil	varchar(15)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> requisitos	varchar(40)	latin1_swedish_ci		No		
<input type="checkbox"/> areaaplicacion	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> horario	varchar(15)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> dispviajar	varchar(4)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> idioma	varchar(10)	latin1_swedish_ci		No		
<input type="checkbox"/> idioma1	varchar(10)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> idioma2	varchar(10)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> salario	varchar(15)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> prestaciones	varchar(15)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> fechaentrev	varchar(11)	latin1_swedish_ci		No		
<input type="checkbox"/> fechainlab	varchar(11)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> cantvacante	int(4)			Sí	NULL	
<input type="checkbox"/> RFCx	varchar(15)	latin1_swedish_ci		No		

La siguiente figura muestra la tabla intereses y sus atributos

Fig. 3.26 Intereses

Base de datos: siac (12)

Tabla: intereses

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
<input type="checkbox"/> idareax	int(5)			No			[Iconos]
<input type="checkbox"/> mat2	int(20)		UNSIGNED	No			[Iconos]

Índices:

Nombre de la clave	Tipo	Cardinalidad	Acción	Campo	Tipo	Uso
PRIMARY	PRIMARY	2	[Iconos]	idareax	Datos	1,8 Bytes
mat2	INDEX	Ninguna	[Iconos]	mat2	Índice	3,072 Bytes
					Total	3,090 Bytes

Estadísticas de la fila

Enunciado	Valor
Formato	fijo
Cotejamiento	latin1_swedish_ci

La siguiente figura muestra la tabla empresas y sus atributos

Fig. 3.27 Tabla Empresas

The screenshot shows the phpMyAdmin interface for a MySQL database named 'siac'. The selected table is 'empresa'. The table structure is as follows:

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> RFC	varchar(15)	latin1_swedish_ci		No		
<input checked="" type="checkbox"/> nombreemp	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> razonsocial	varchar(30)	latin1_S[Sueco, independiente de mayúsculas y minúsculas]				
<input type="checkbox"/> apellidop	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> apellidom	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> Nombre	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> calle	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> numint	int(6)			Sí	NULL	
<input type="checkbox"/> numext	int(6)			Sí	NULL	
<input type="checkbox"/> colonia	varchar(20)	latin1_swedish_ci		No		
<input type="checkbox"/> cp	int(6)			Sí	NULL	
<input type="checkbox"/> te1	int(10)			No		
<input type="checkbox"/> tel2	int(10)			Sí	NULL	
<input type="checkbox"/> email	varchar(20)	latin1_swedish_ci		Sí	NULL	
<input type="checkbox"/> email2	varchar(20)	latin1_swedish_ci		Sí	NULL	

La siguiente figura muestra la tabla estudios y sus atributos

Fig. 3.28 Tabla Estudios

Servidor: localhost ▶ **Base de datos:** siac ▶ **Tabla:** estudios "InnoDB free: 3072 kB"
Operaciones: Examinar Estructura SQL Buscar Insertar Exportar Importar
Operaciones: Vaciar Eliminar

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idestudio	int(10)		UNSIGNED	No		
<input type="checkbox"/> Carrera	varchar(50)	latin1_swedish_ci		No		

Índices:

Nombre de la clave	Tipo	Cardinalidad	Acción	Campo	Tipo	Uso
PRIMARY	PRIMARY	2		idestudio	Datos	1,6,384 Bytes
					Índice	0 Bytes
					Total	1,6,384 Bytes

Estadísticas de la fila

Enunciado	Valor
Formato	Compact
Cotejamiento	latin1_swedish_ci
Creación	24-09-2008 a las 15:53:41

La siguiente figura muestra la tabla administrador y sus atributos

Servidor: localhost ▶ **Base de datos:** siac ▶ **Tabla:** administrador "InnoDB free: 3072 kB"
Operaciones: Examinar Estructura SQL Buscar Insertar Exportar Importar
Operaciones: Vaciar Eliminar

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idadmin	int(5)			No		auto_increment
<input type="checkbox"/> Nombre	varchar(10)	latin1_swedish_ci		No		
<input type="checkbox"/> Contraseña	varchar(10)	latin1_swedish_ci		No		

Índices:

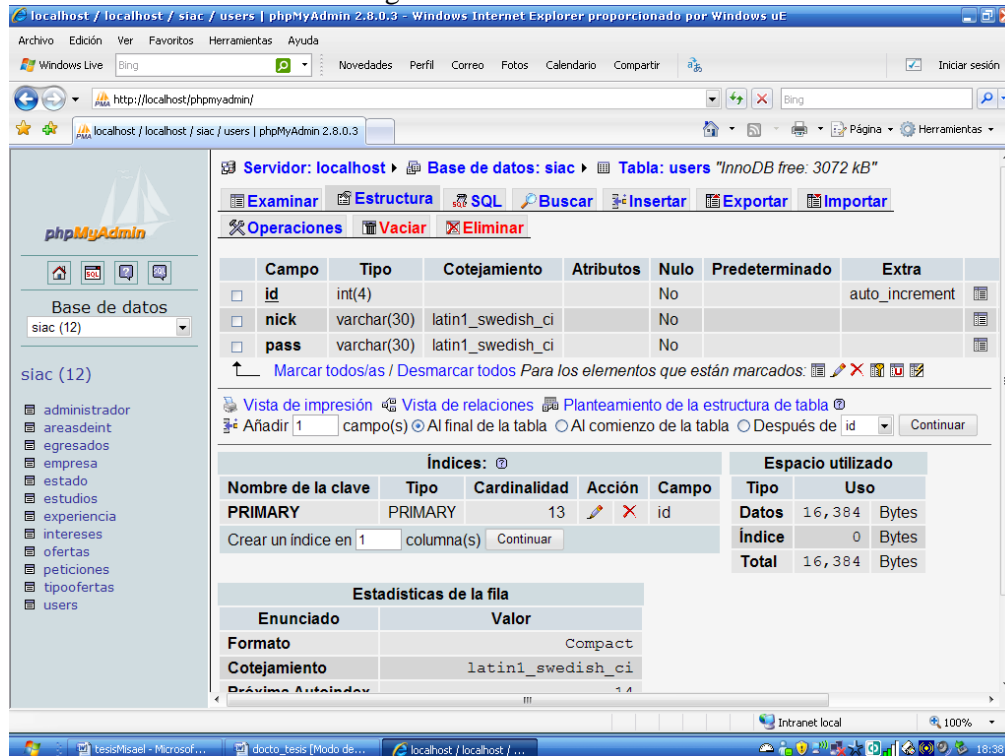
Nombre de la clave	Tipo	Cardinalidad	Acción	Campo	Tipo	Uso
PRIMARY	PRIMARY	1		idadmin	Datos	1,6,384 Bytes
					Índice	0 Bytes
					Total	1,6,384 Bytes

Estadísticas de la fila

Enunciado	Valor
Formato	Compact
Cotejamiento	latin1_swedish_ci

Fig. 3.29 Tabla Administrador

La siguiente figura muestra la tabla users y sus atributos
Fig. 3.30 Tabla Usuarios



La siguiente figura muestra la tabla peticiones y sus atributos

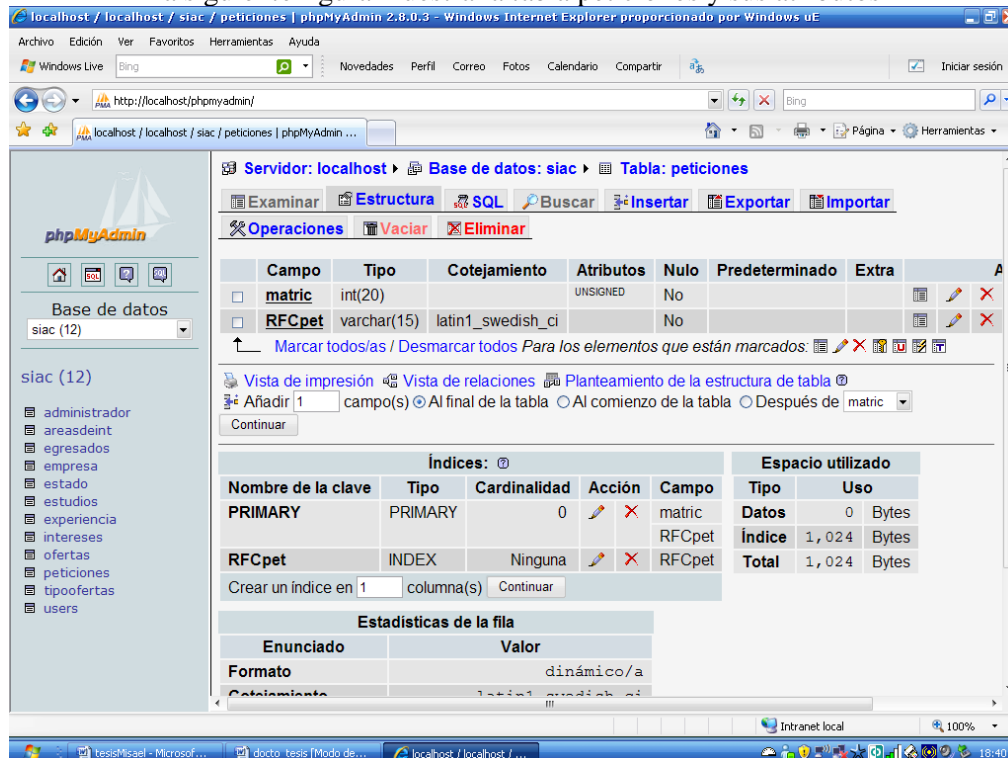


Fig. 3.31 Tabla Peticiones

La siguiente figura muestra la tabla tipooferta y sus atributos

The screenshot displays the phpMyAdmin interface for the 'siac' database, specifically the 'tipoofertas' table. The table structure is as follows:

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idtipo	varchar(10)	latin1_swedish_ci		No		
<input type="checkbox"/> desctipo	varchar(50)	latin1_swedish_ci		No		

Below the table structure, there are options for adding fields and creating indexes. The 'Índices' section shows a PRIMARY index on the 'idtipo' field.

Nombre de la clave	Tipo	Cardinalidad	Acción	Campo
PRIMARY	PRIMARY	2		idtipo

The 'Espacio utilizado' section shows the following data:

Tipo	Uso
Datos	16,384 Bytes
Índice	0 Bytes
Total	16,384 Bytes

The 'Estadísticas de la fila' section provides additional details:

Enunciado	Valor
Formato	Compact
Cotejamiento	latin1_swedish_ci
Creación	18-02-2008 a las 16:24:11

Capítulo



4

Implementación

Después de haber analizado, diseñado e implementado la Base de datos del proyecto, lo siguiente en la lista, es la implementación del sistema con la ayuda de las herramientas HTML, PHP y MySQL.

4.1 Ambiente de las herramientas de implementación

Las herramientas utilizadas para el desarrollo de este proyecto fueron las siguientes:

4.1.1 MySQL y phpMyAdmin

El motor de Bases de Datos MySQL se instala únicamente en la máquina que fungirá como servidor. Aunque la potencia y estabilidad de MySQL es competente a los mejores SGBD como Oracle y SQL Server, el sistema en modo consola es un poco tedioso de utilizar, sobre todo en la creación de las tablas. Para evitar trabajar sobre el modo consola de MySQL utilizaremos phpMyAdmin, que es un frontend que se distribuye de manera gratuita, en el paquete WAMP.

4.2 Descripción general del sistema

4.2.1 Pagina Principal

La siguiente figura muestra la pagina principal del SIAC como podemos apreciar contiene varias secciones que explicaremos a lo largo de este capítulo

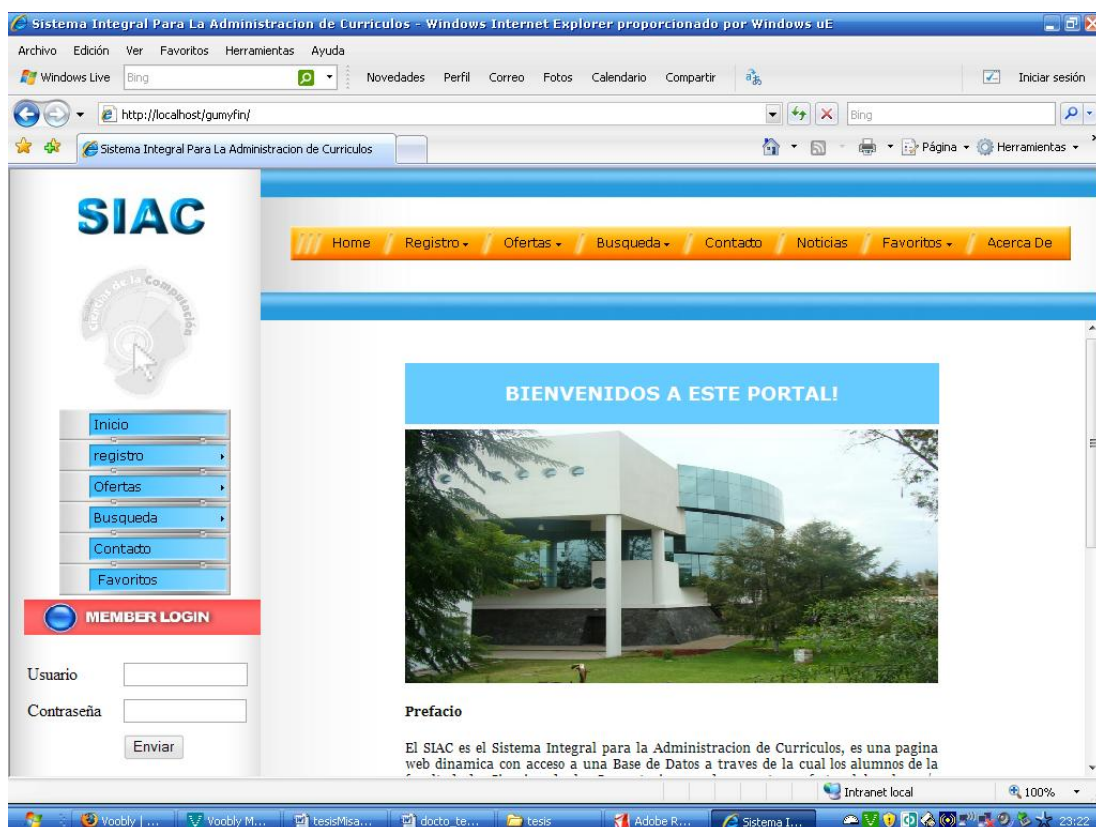


Fig.4.0 Pagina Principal del Sistema

4.2.2 Registro

En el apartado de registro encontraremos una serie de formularios diseñados para un registro de información específica de acuerdo con las necesidades del usuario. A continuación se muestra el despliegue de este apartado

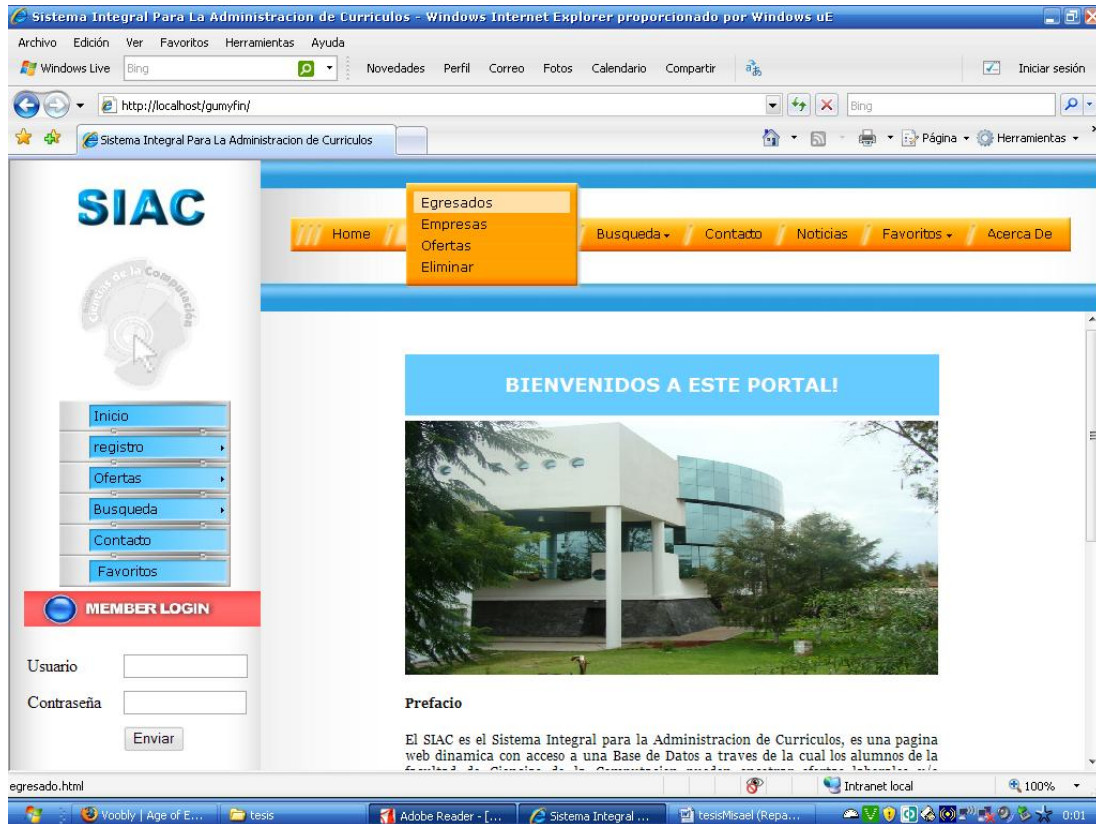


Fig. 4.1 Área de Registro

Como podemos apreciar en la figura el área de registro tiene cuatro opciones

1. la primera es para registrar los datos de las personas que buscan empleo o prácticas profesionales
2. la segunda es para las empresas que ofertan trabajo o practicas.
3. En ofertas las empresas previamente registradas ingresan las ofertas de acuerdo con sus necesidades
4. En eliminar el administrador del sistema borra la información de egresados que ya obtuvieron empleo o bien que tienen mucho tiempo en el sistema.

El registro de egresados se hace mediante formularios donde el interesado ingresa los datos más importantes que se muestran a continuación:

The screenshot shows a web browser window displaying the 'Sistema Integral Para La Administración de Currículos' (SIAC) website. The main content area is titled 'REGISTRA TUS DATOS'. On the left, there is a 'MEMBER LOGIN' section with fields for 'Usuario' and 'Contraseña', and an 'Enviar' button. The main form is divided into two sections: 'Datos Personales' and 'Direccion'. The 'Datos Personales' section includes fields for 'Matricula', 'Apellido P', 'Apellido Mat', 'Nombre(s)', 'Edad' (with a dropdown menu 'Selecciona tu Edad'), and 'Sexo' (with radio buttons for 'Masculino' and 'Femenino'). The 'Direccion' section is currently empty. The browser's address bar shows 'http://localhost/gumyfin/'.

Fig. 4.2 Registro de Datos del egresado

En este primer apartado el usuario ingresa sus datos personales

The screenshot shows the same SIAC website, but with the 'Egresados' menu item selected in the top navigation bar. The main content area now displays the 'Direccion' and 'Datos De Interes' sections. The 'Direccion' section includes fields for 'Calle', 'Num Int', 'Num Ext', 'Colonia', and 'C. P.', along with 'Lugar de nacimiento' and 'Nacionalidad'. The 'Datos De Interes' section includes a 'Fecha de Nacimiento' field. The browser's address bar shows 'http://localhost/gumyfin/'.

Fig. 4.3 Registro de la Dirección

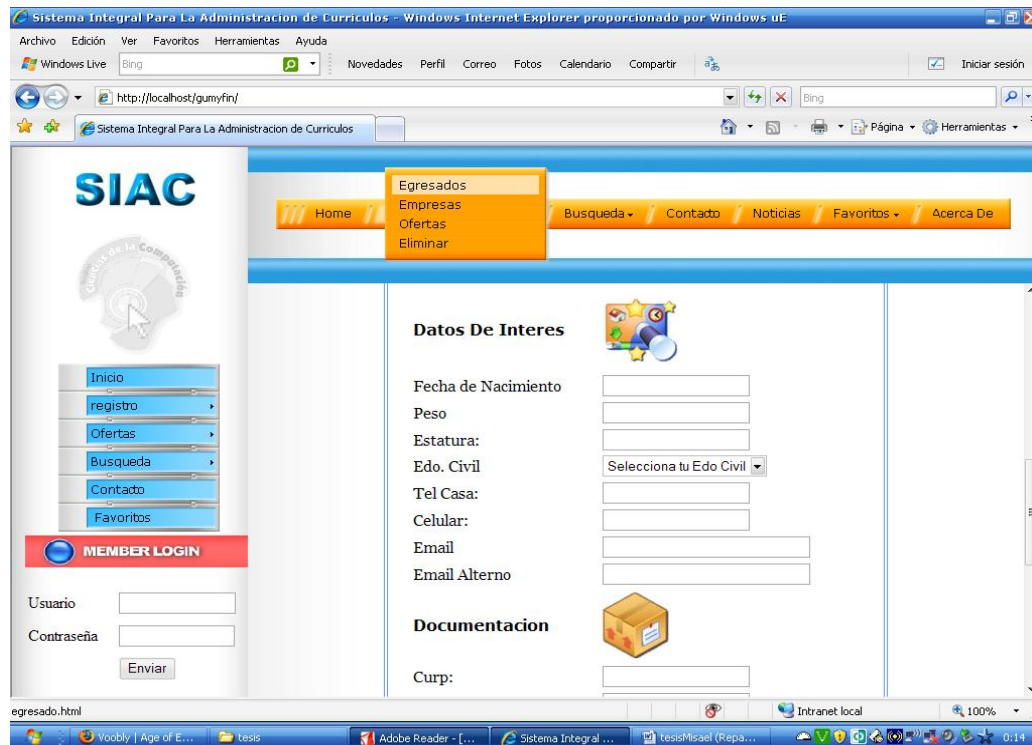


Fig. 4.4 Registro de datos de Interés
En este apartado se registran algunos datos de interés

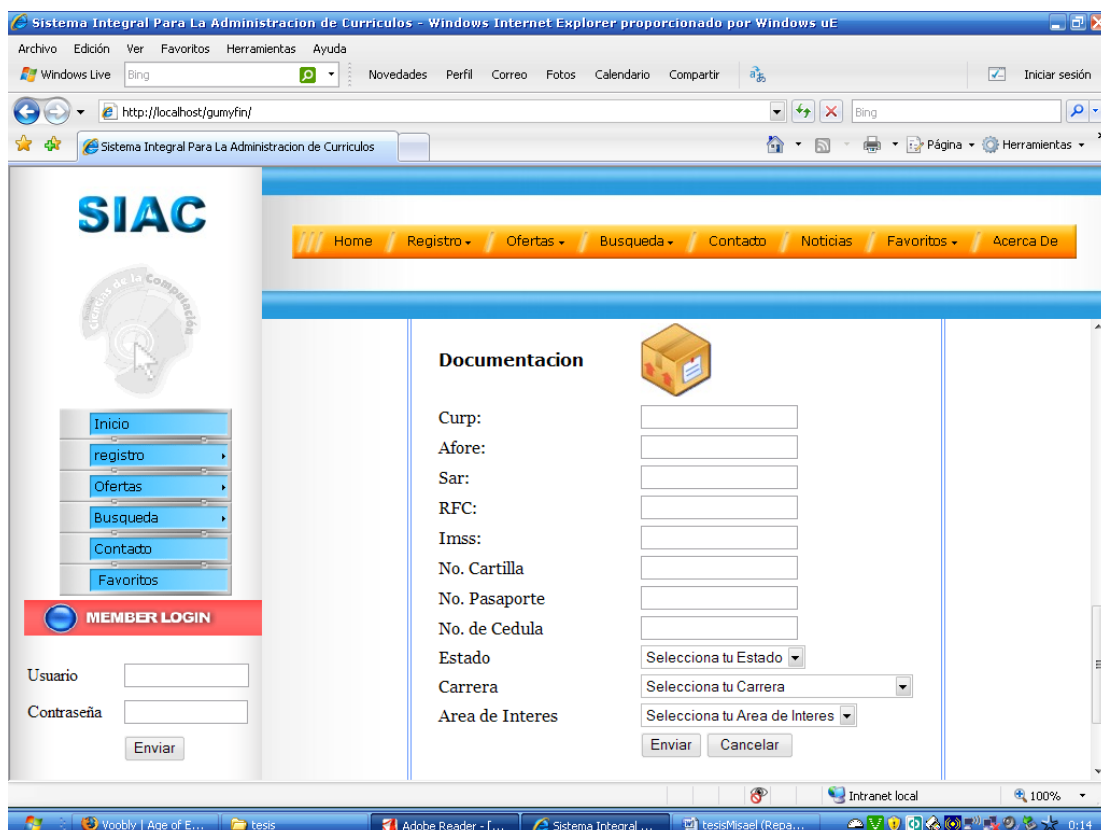


Fig. 4.5 Registro de Documentación
Aquí se registra la documentación con la que cuenta el interesado

El registro de empresas se hace mediante formularios donde la empresa ingresa los datos más importantes que se muestran a continuación:

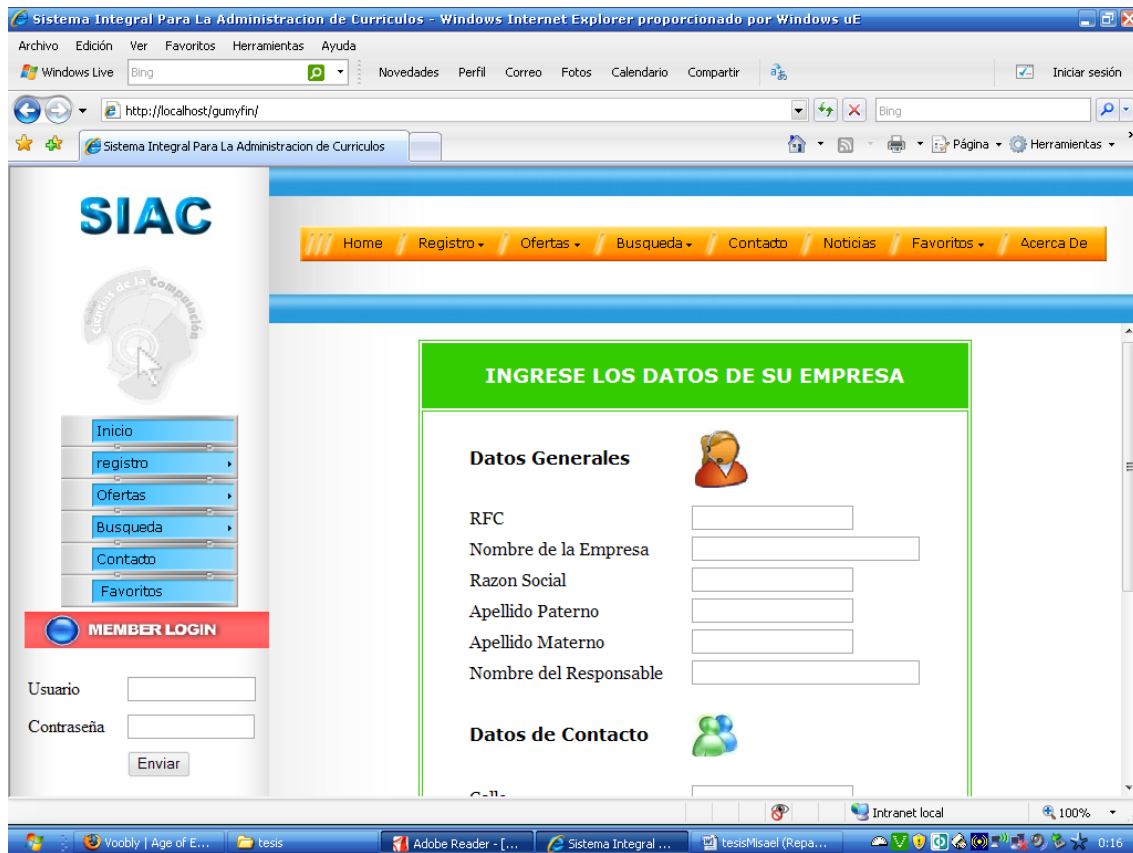


Fig. 4.6 Registro de Datos Generales de la Empresa

En esta primera sección se recaban los datos generales de la empresa como podemos apreciar.

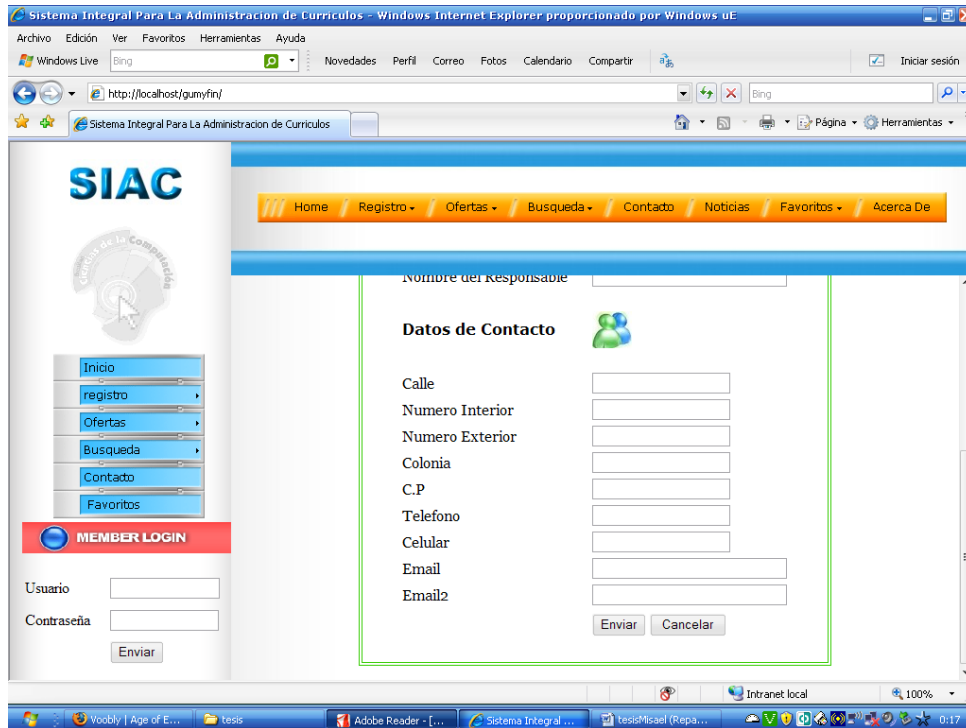


Fig. 4.7 Registro de Datos de Contacto

En esta segunda sección se recaban los datos para poder estar en contacto si es que se requiere.

El registro de las ofertas se realiza de la siguiente manera:

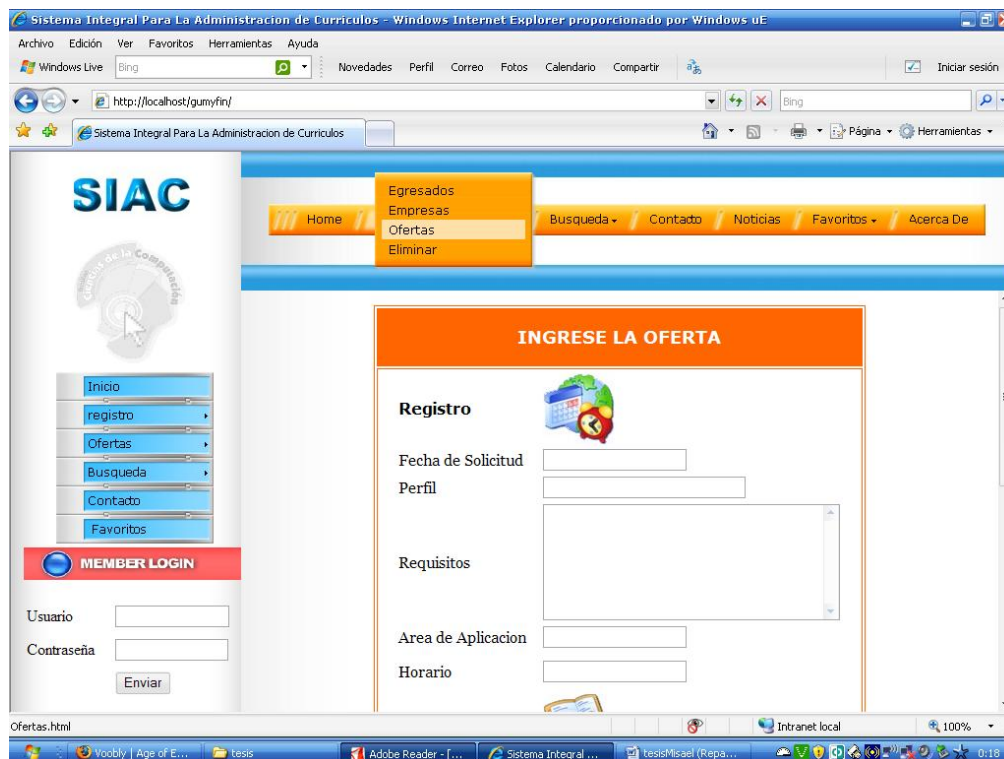


Fig. 4.8 Registro de Oferta

En esta sección se almacena el registro de la oferta

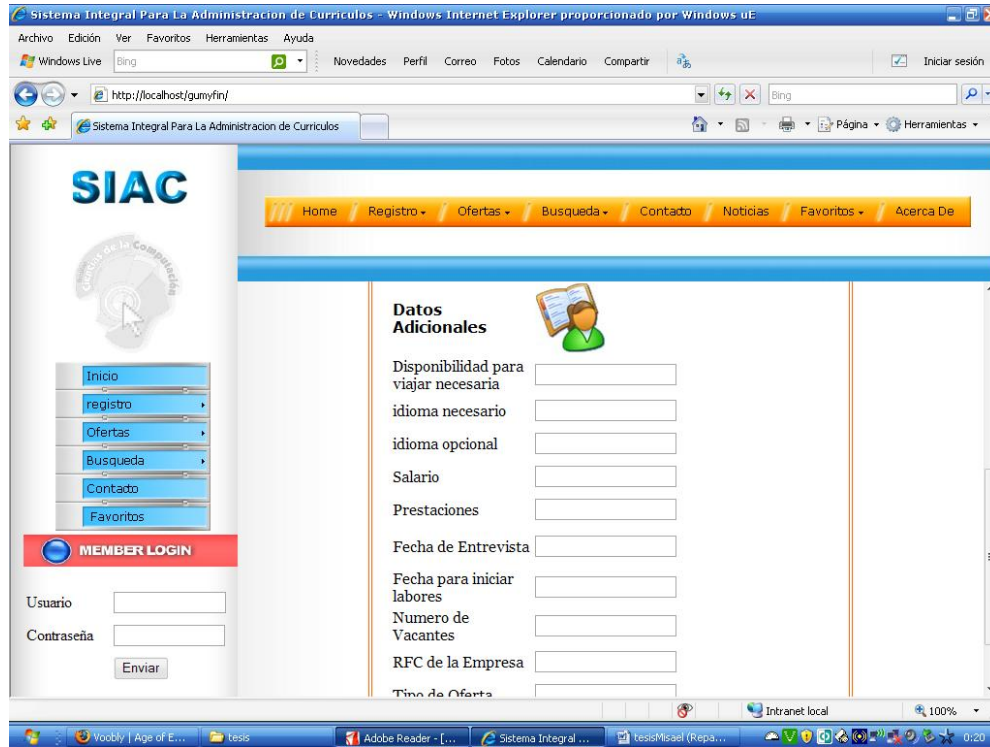


Fig. 4.9 Registro de Datos Adicionales

En esta sección se llenan datos adicionales que pueden hacer más atractiva la oferta

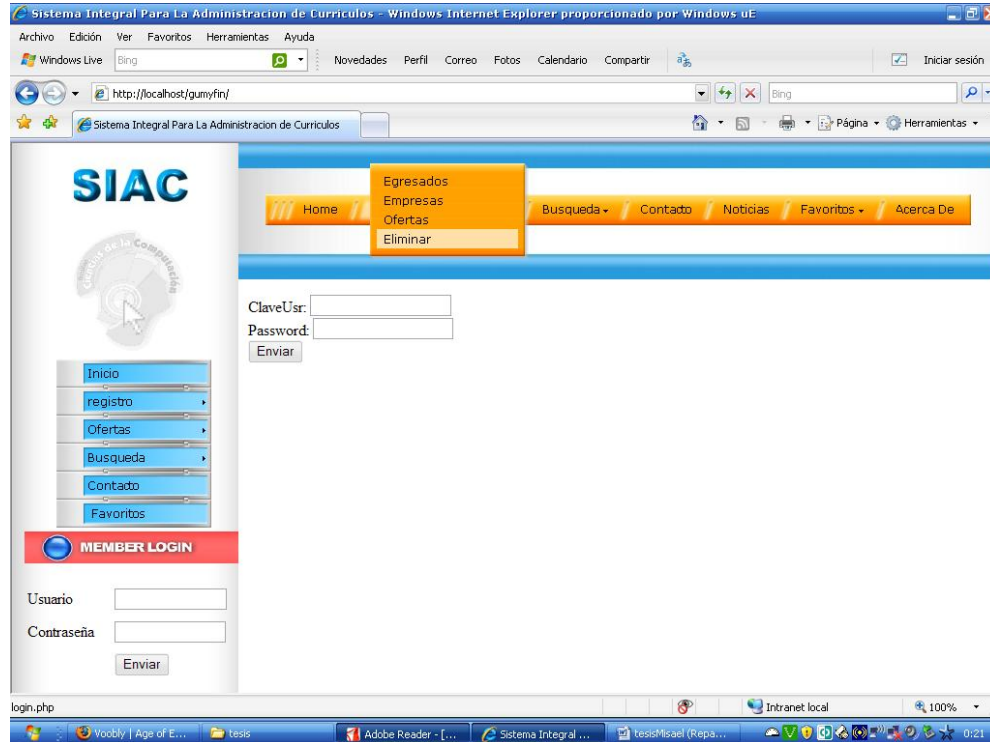


Fig. 4.10 Validación de Ingreso como Administrador

Al ingresar en el apartado eliminar el sistema nos pedirá que ingresemos una clave y contraseña como administrador ya que él será el único capacitado para eliminar registros

Si la validación es correcta entonces se mostrara la sig. Pagina solo para el administrador

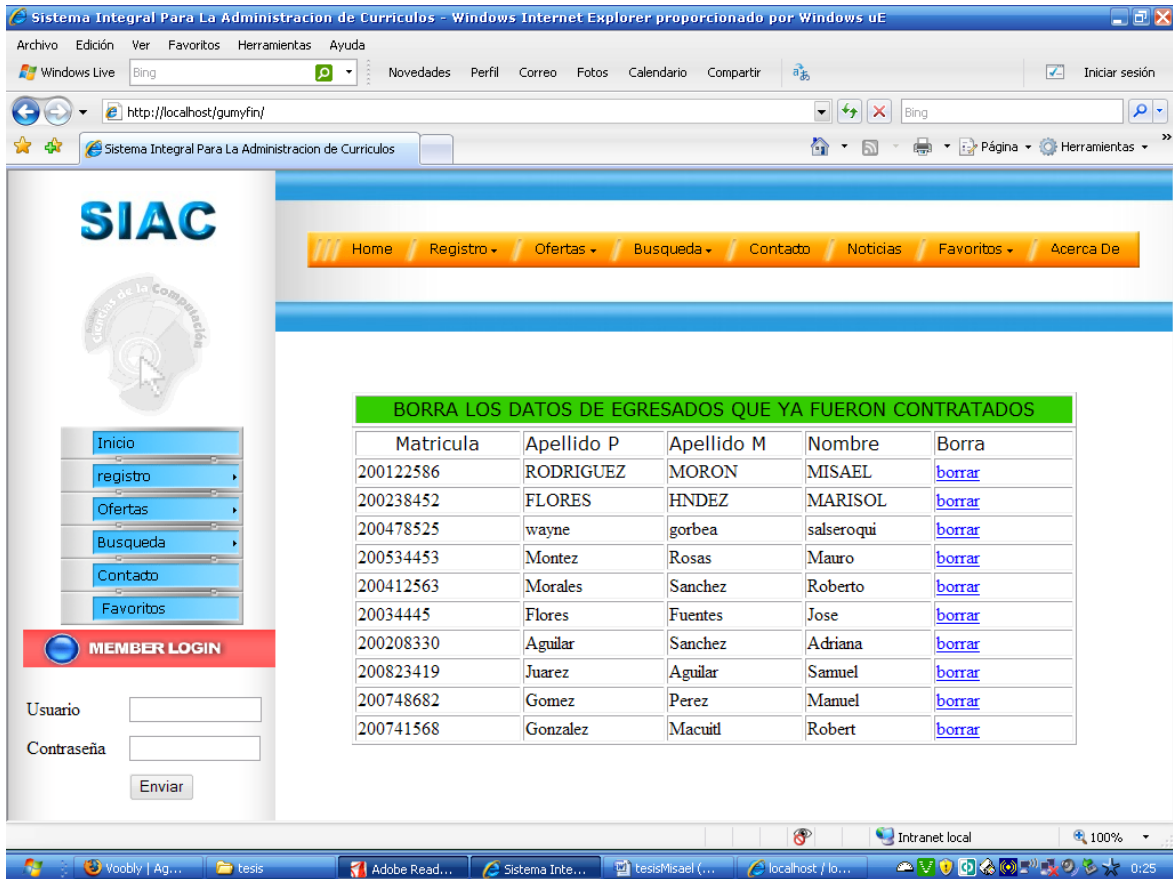
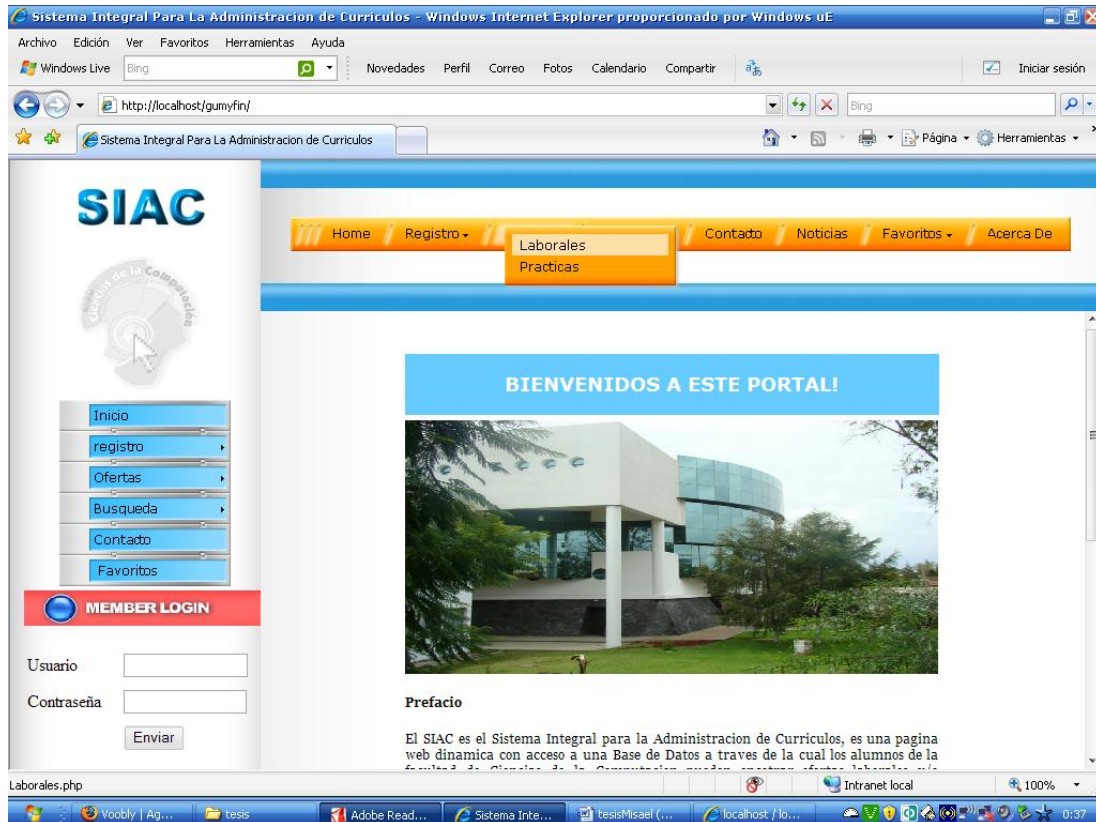


Fig. 4.11 Eliminación de Registros

En esta página el administrador puede seleccionar un registro y eliminarlo completamente

4.2.3 Ofertas

En esta sección se encuentran las ofertas y practicas más recientes y se pueden consultar libremente la consulta aparecerá en una tabla donde se mostraran los datos más sobresalientes de dicha oferta como se muestra a continuación.



En la fig. 4.12 se muestra el menú ofertas

Como podemos ver en la figura la sección ofertas cuenta con dos opciones:

1. Laborales, se refiere a ofertas de trabajo que registran las empresas
2. Practicas, se refiere a prácticas profesionales publicadas por empresas

En la opción laborales encontraremos resultados como los que se muestran a continuación

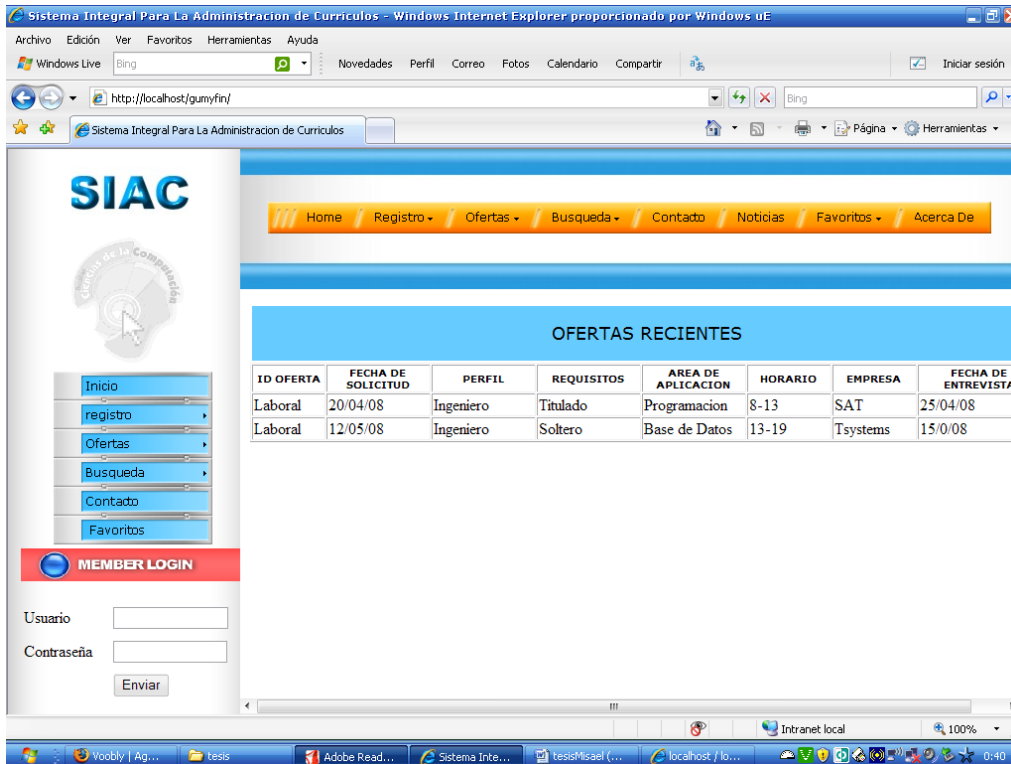


Fig. 4.13 Ofertas Laborales

Como podemos apreciar en esta figura la ofertas son de trabajo y se muestran los datos más importantes.

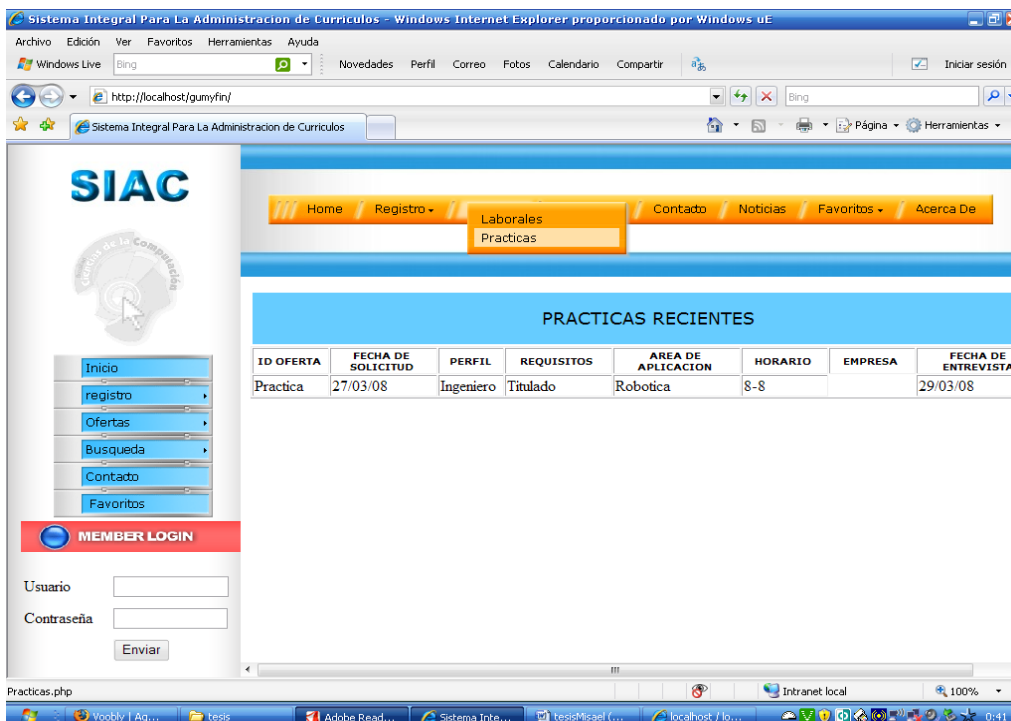
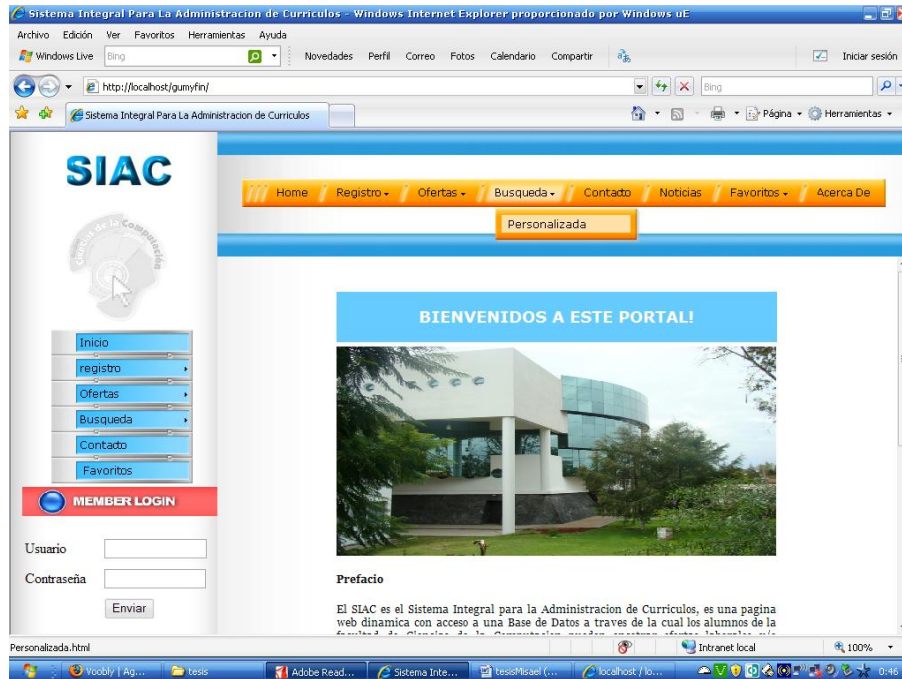


Fig. 4.14 Practicas Profesionales

Aquí podemos ver los resultados que muestran las prácticas disponibles por el momento

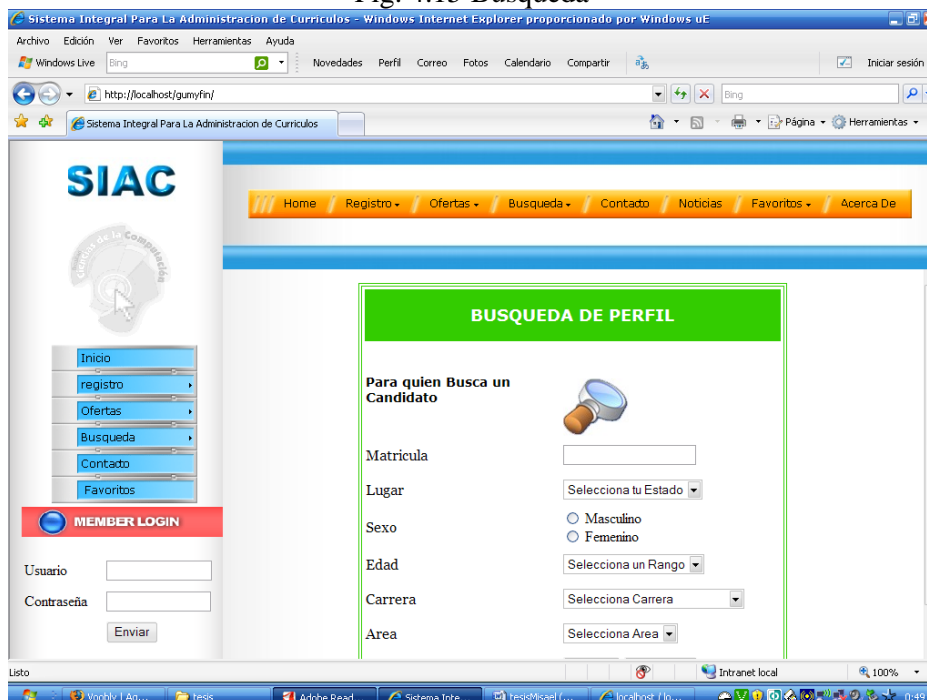
4.2.4 Búsqueda

Esta sección es muy importante ya que los egresados y las empresas podrán realizar una consulta de acuerdo con sus necesidades y de esta manera encontrar al mejor candidato para ocupar la vacante.



En la página de búsqueda podremos mencionar varios atributos para que nuestra búsqueda arroje resultados deseables

Fig. 4.15 Búsqueda



4.2.5 Contacto

Es una página con un email sender el cual puede mandar emails directamente al administrador y este a su vez realizar el papel de intermediario para encontrar coincidencias y que se obtenga un máximo aprovechamiento de la información

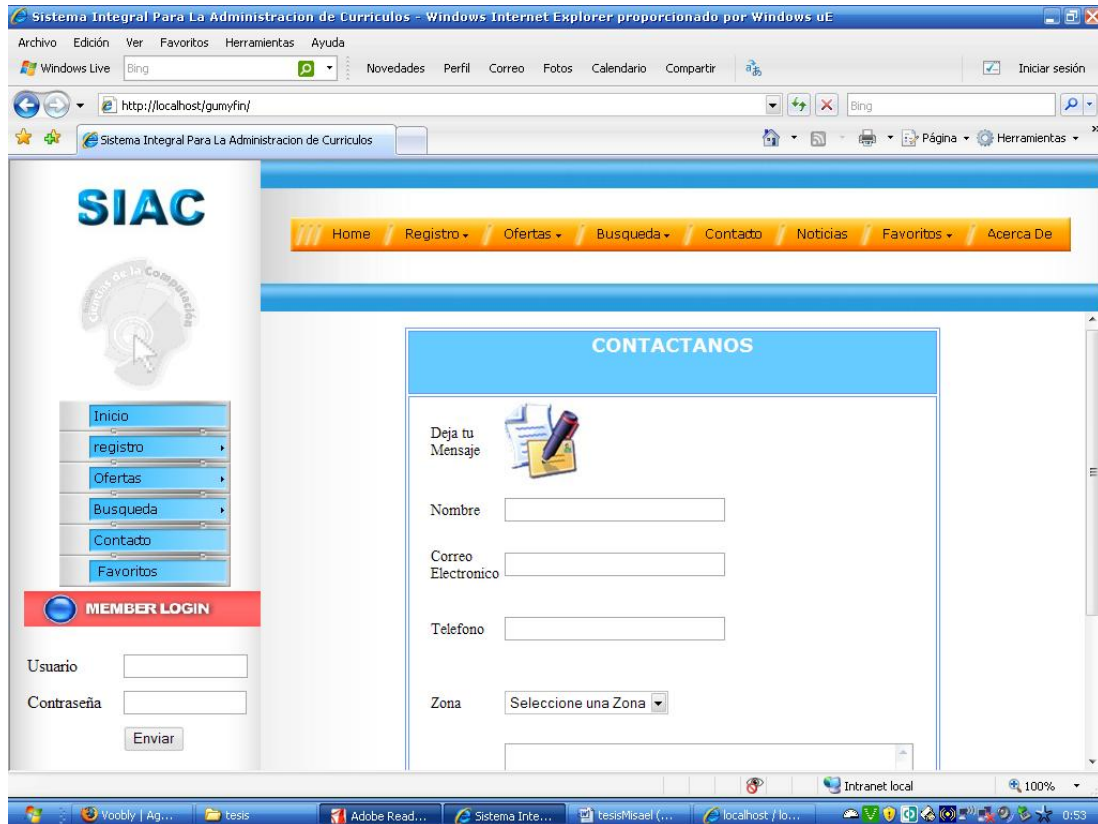


Fig. 4.16 Contacto

El sistema cuenta con una liga al buscador más importante con el fin de ampliar la información

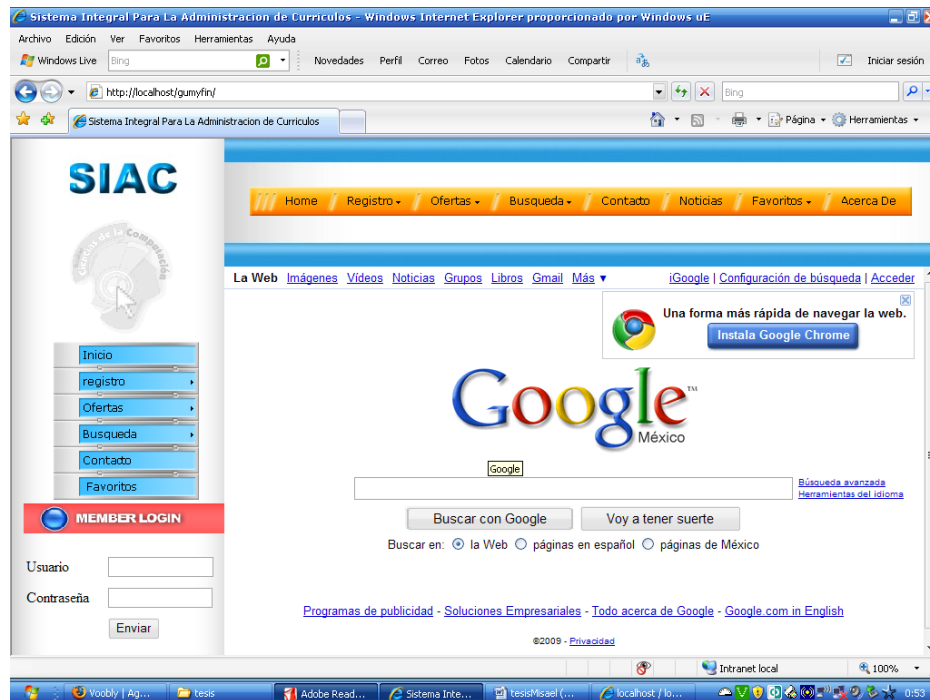


Fig. 4.17 liga con google

También cuenta con una liga directa a la página de la FCC con lo cual se asegura un mejor desempeño ya que la mayoría de los alumnos de la FCC ingresa a la página de la facultad buscando información de interés.

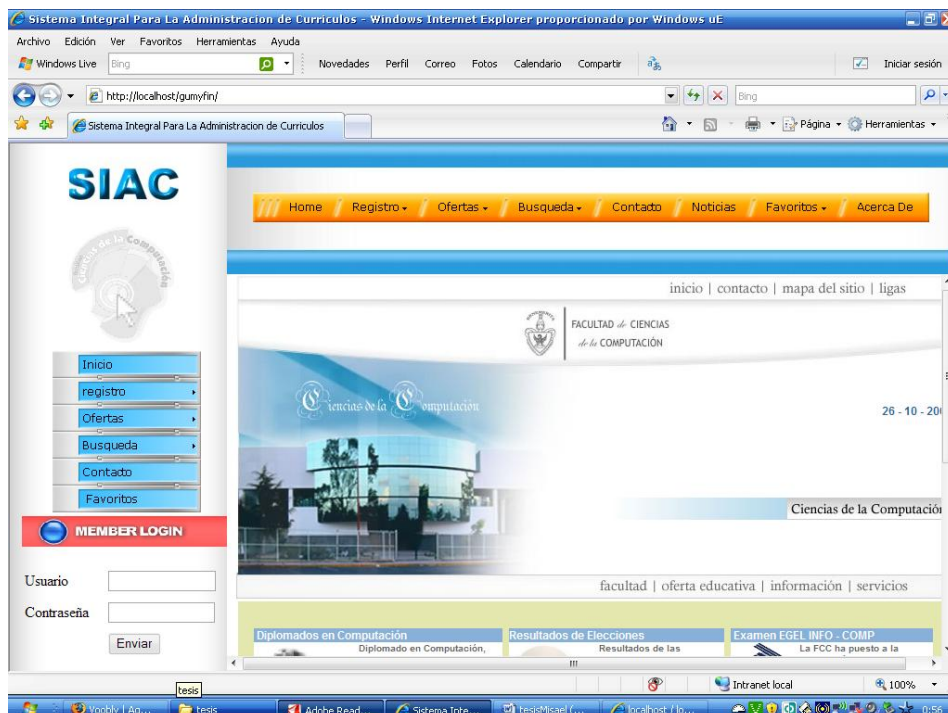


Fig. 4.18 liga facultad de ciencias de la Computación


4.3 Pruebas y Resultados

En este apartado realizaremos pruebas a nuestro sistema y mostraremos los resultados obtenidos al utilizarlo-

4.3.1 Registro de Un Egresado

A continuación realizaremos un registro de un egresado, de una empresa y también de una oferta y haremos énfasis en algunos puntos de importancia acerca del funcionamiento del sistema


1.- Se registran los datos personales

Caso de Experimento	Registro de un Egresado
Objetivo	Verificar que el sistema almacena información de Egresados
Inf. De Entrada	
Resultado Esperado	El sistema muestra el fomulario donde se llenan los camops
Resultado Obtenido	El sistema es funcional en el almacenamiento de información
Resultado de la Prueba	Satisfactoria


2.- Se registran los datos correspondientes a la dirección

Caso de Experimento	Registro de la Dirección
Objetivo	Verificar que el sistema almacena información de Egresados
Inf. De Entrada	
Resultado Esperado	El sistema muestra el fomulario donde se llenan los camops
Resultado Obtenido	El sistema es funcional en el almacenamiento de información
Resultado de la Prueba	Satisfactoria

3.- Se registran los datos de Interés

Caso de Experimento	Registro de los Datos de Interés
Objetivo	Verificar que el sistema almacena informacion de Egresados
Inf. De Entrada	<div style="border: 1px solid black; padding: 10px;"> <p>Datos De Interés </p> <p>Fecha de Nacimiento <input type="text" value="12/09/87"/></p> <p>Peso <input type="text" value="76"/></p> <p>Estatura: <input type="text" value="1.80"/></p> <p>Edo. Civil <input type="text" value="Soltero"/></p> <p>Tel Casa: <input type="text" value="2134256"/></p> <p>Celular: <input type="text" value="2224576891"/></p> <p>Email <input type="text" value="jorgito_2@hotmail.com"/></p> <p>Email Alterno <input type="text"/></p> </div>
Resultado Esperado	El sistema muestra el formulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de informacion
Resultado de la Prueba	Satisfactoria


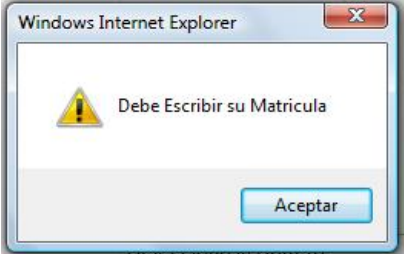
4.- Se registran los Datos de la Documentación y se envían todos los Datos


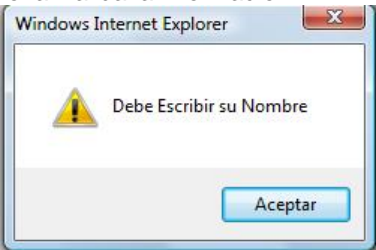
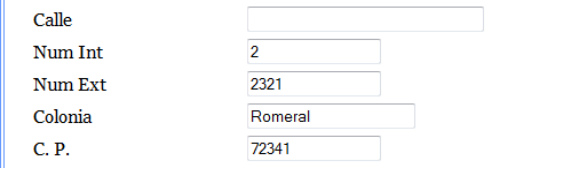
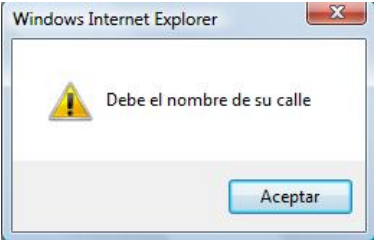
Caso de Experimento	Registro de la documentación
Objetivo	Verificar que el sistema almacena informacion de Egresados
Inf. De Entrada	<div style="border: 1px solid black; padding: 10px;"> <p>Documentación </p> <p>Curp: <input type="text" value="ROMM830327HPLDR"/></p> <p>Afore: <input type="text" value="no"/></p> <p>Sar: <input type="text" value="no"/></p> <p>RFC: <input type="text" value="ROMM830327NY2"/></p> <p>Imss: <input type="text" value="no"/></p> <p>No. Cartilla <input type="text" value="123456677"/></p> <p>No. Pasaporte <input type="text" value="no"/></p> <p>No. de Cedula <input type="text" value="no"/></p> <p>Estado <input type="text" value="Puebla"/></p> <p>Carrera <input type="text" value="Licenciatura en Cs. de la Computacion"/></p> <p>Area de Interés <input type="text" value="Robotica"/></p> <p><input type="button" value="Enviar"/> <input type="button" value="Cancelar"/></p> </div>
Resultado Esperado	El sistema muestra el formulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de informacion
Resultado de la Prueba	Satisfactoria

5. Se completa el registro escogiendo un login y un password

Caso de Experimento	Completar el Registro
Objetivo	Completa el registro pidiendo al usuario que ingrese un login y un password
Inf. De Entrada	
Resultado Esperado	El sistema indicara si el ingreso es correcto
Resultado Obtenido	El sistema manda el siguiente mensaje 
Resultado de la Prueba	Satisfactoria


Se realizan pruebas para verificar que los formularios están validados y de esta manera filtrar la información que se va a enviar ya que como sabemos hay campos obligatorios que no pueden ser nulos dentro de la base de datos. Esto se realiza con una sencilla función creada en el lenguaje javascript la cual verifica que no haya atributos vacios en el formulario y solo entonces envía la información en caso contrario manda unas alertas como las que veremos a continuación.


Caso de Experimento	Envío de Información Incorrecta
Objetivo	Verificar que el sistema valida la información que se envia
Inf. De Entrada	
Resultado Esperado	El sistema muestra las alertas de error
Resultado Obtenido	El sistema es funcional valida la información 
Resultado de la Prueba	Satisfactoria


Caso de Experimento	Envío de Información Incorrecta
Objetivo	Verificar que el sistema valida la información que se envía
Inf. De Entrada	
Resultado Esperado	El sistema muestra las alertas de error
Resultado Obtenido	El sistema es funcional valida la información 
Resultado de la Prueba	Satisfactoria
Caso de Experimento	Envío de Información Incorrecta
Objetivo	Verificar que el sistema valida la información que se envía
Inf. De Entrada	
Resultado Esperado	El sistema muestra las alertas de error
Resultado Obtenido	El sistema es funcional valida la información 
Resultado de la Prueba	Satisfactoria

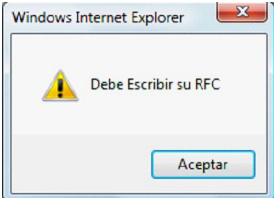
Nota: Se están incluyendo los casos más importantes sin embargo se realizaron validaciones de todos los atributos que así lo requieren.

4.3.2 Registro de una Empresa

Caso de Experimento	Registro de una Empresa
Objetivo	Verificar que el sistema almacena información de las Empresas
Inf. De Entrada	
Resultado Esperado	El sistema muestra el fomulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de información
Resultado de la Prueba	Satisfactoria

Caso de Experimento	Registro de una Empresa
Objetivo	Verificar que el sistema almacena información de las Empresas
Inf. De Entrada	
Resultado Esperado	El sistema muestra el fomulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de información
Resultado de la Prueba	Satisfactoria

Caso de Experimento	Envío de Información Errónea
Objetivo	Verificar que el sistema valida la información enviada
Inf. De Entrada	
Resultado Esperado	El sistema muestra el fomulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de información

	
--	--

Resultado de la Prueba	Satisfactoria
-------------------------------	---------------

4.3.3 Registro de Una Oferta

Caso de Experimento	Registro de la Oferta
----------------------------	------------------------------

Objetivo	Verificar que el sistema envia y registra la informacion
-----------------	--

Inf. De Entrada	
------------------------	--


Resultado Esperado	El sistema muestra el fomulario donde se llenan los campos
---------------------------	--

Resultado Obtenido	El sistema es funcional en el almacenamiento de informacion
---------------------------	---

Resultado de la Prueba	Satisfactoria
-------------------------------	---------------

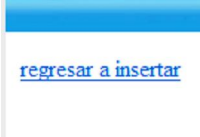
Caso de Experimento	Registro de los Datos adicionales de la oferta
----------------------------	---

Objetivo	Verificar que el sistema envia y registra la informacion
-----------------	--


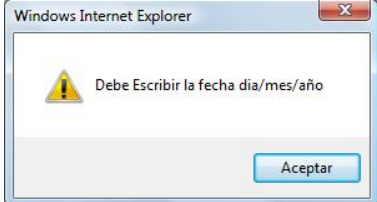
Inf. De Entrada	
------------------------	--

Resultado Esperado	El sistema muestra el fomulario donde se llenan los campos
---------------------------	--


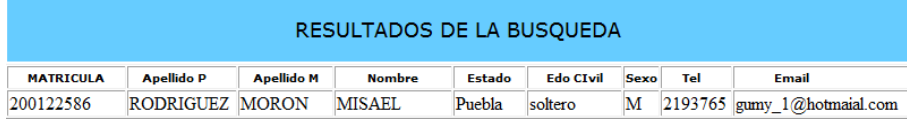
Resultado Obtenido	El sistema es funcional en el almacenamiento de informacion
---------------------------	---


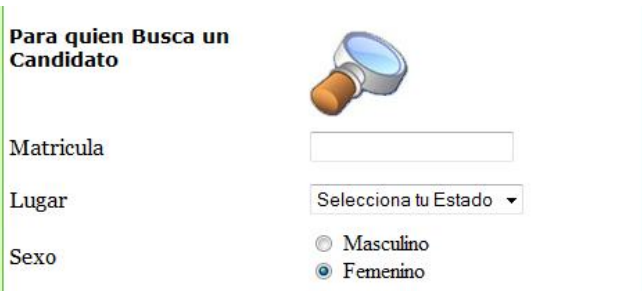
	
--	--

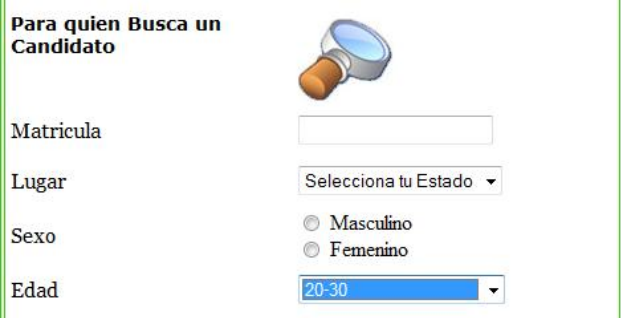
Resultado de la Prueba	Satisfactoria
-------------------------------	---------------

Caso de Experimento	Envío de Información Errónea
Objetivo	Verificar que el sistema valida la información enviada
Inf. De Entrada	
Resultado Esperado	El sistema muestra el formulario donde se llenan los campos
Resultado Obtenido	El sistema es funcional en el almacenamiento de información 
Resultado de la Prueba	Satisfactoria

4.3.4 Búsqueda

Caso de Experimento	Búsqueda de Información																		
Objetivo	Verificar que el sistema realiza las consultas de información																		
Inf. De Entrada																			
Resultado Esperado	El sistema muestra el resultado de la consulta en una tabla																		
Resultado Obtenido	 <table border="1"> <thead> <tr> <th>MATRICULA</th> <th>Apellido P</th> <th>Apellido M</th> <th>Nombre</th> <th>Estado</th> <th>Edo Civil</th> <th>Sexo</th> <th>Tel</th> <th>Email</th> </tr> </thead> <tbody> <tr> <td>200122586</td> <td>RODRIGUEZ</td> <td>MORON</td> <td>MISAEAL</td> <td>Puebla</td> <td>soltero</td> <td>M</td> <td>2193765</td> <td>gumy_1@hotmail.com</td> </tr> </tbody> </table>	MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email	200122586	RODRIGUEZ	MORON	MISAEAL	Puebla	soltero	M	2193765	gumy_1@hotmail.com
MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email											
200122586	RODRIGUEZ	MORON	MISAEAL	Puebla	soltero	M	2193765	gumy_1@hotmail.com											
Resultado de la Prueba	Satisfactoria																		

Caso de Experimento	Búsqueda por Lugar de Procedencia																																																																																										
Objetivo	Verificar que el sistema realiza las consultas de información																																																																																										
Inf. De Entrada																																																																																											
Resultado Esperado	El sistema muestra el resultado de la consulta en una tabla																																																																																										
Resultado Obtenido	<table border="1"> <thead> <tr> <th colspan="10">RESULTADOS DE LA BUSQUEDA</th> </tr> <tr> <th>MATRICULA</th> <th>Apellido P</th> <th>Apellido M</th> <th>Nombre</th> <th>Estado</th> <th>Edo Civil</th> <th>Sexo</th> <th>Tel</th> <th colspan="2">Em</th> </tr> </thead> <tbody> <tr> <td>20034445</td> <td>Flores</td> <td>Fuentes</td> <td>Jose</td> <td>Puebla</td> <td></td> <td>M</td> <td>121454</td> <td colspan="2">papasnu</td> </tr> <tr> <td>200122586</td> <td>RODRIGUEZ</td> <td>MORON</td> <td>MISAEAL</td> <td>Puebla</td> <td>soltero</td> <td>M</td> <td>2193765</td> <td colspan="2">gumy_1(</td> </tr> <tr> <td>200208330</td> <td>Aguilar</td> <td>Sanchez</td> <td>Adriana</td> <td>Puebla</td> <td></td> <td>F</td> <td>2439919</td> <td colspan="2">adri772(</td> </tr> <tr> <td>200412563</td> <td>Morales</td> <td>Sanchez</td> <td>Roberto</td> <td>Puebla</td> <td></td> <td>M</td> <td>2854575</td> <td colspan="2">moralesj</td> </tr> <tr> <td>200478525</td> <td>wayne</td> <td>gorbea</td> <td>salseroqui</td> <td>Puebla</td> <td></td> <td>M</td> <td></td> <td colspan="2"></td> </tr> <tr> <td>200534453</td> <td>Montez</td> <td>Rosas</td> <td>Mauro</td> <td>Puebla</td> <td></td> <td>M</td> <td>245825</td> <td colspan="2">muarinhc</td> </tr> <tr> <td>200823419</td> <td>Juarez</td> <td>Aguilar</td> <td>Samuel</td> <td>Puebla</td> <td></td> <td>M</td> <td>1457896</td> <td colspan="2">sami@h</td> </tr> </tbody> </table>	RESULTADOS DE LA BUSQUEDA										MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Em		20034445	Flores	Fuentes	Jose	Puebla		M	121454	papasnu		200122586	RODRIGUEZ	MORON	MISAEAL	Puebla	soltero	M	2193765	gumy_1(200208330	Aguilar	Sanchez	Adriana	Puebla		F	2439919	adri772(200412563	Morales	Sanchez	Roberto	Puebla		M	2854575	moralesj		200478525	wayne	gorbea	salseroqui	Puebla		M				200534453	Montez	Rosas	Mauro	Puebla		M	245825	muarinhc		200823419	Juarez	Aguilar	Samuel	Puebla		M	1457896	sami@h	
RESULTADOS DE LA BUSQUEDA																																																																																											
MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Em																																																																																			
20034445	Flores	Fuentes	Jose	Puebla		M	121454	papasnu																																																																																			
200122586	RODRIGUEZ	MORON	MISAEAL	Puebla	soltero	M	2193765	gumy_1(
200208330	Aguilar	Sanchez	Adriana	Puebla		F	2439919	adri772(
200412563	Morales	Sanchez	Roberto	Puebla		M	2854575	moralesj																																																																																			
200478525	wayne	gorbea	salseroqui	Puebla		M																																																																																					
200534453	Montez	Rosas	Mauro	Puebla		M	245825	muarinhc																																																																																			
200823419	Juarez	Aguilar	Samuel	Puebla		M	1457896	sami@h																																																																																			
Resultado de la Prueba	Satisfactoria																																																																																										
Caso de Experimento	Búsqueda de Información																																																																																										
Objetivo	Verificar que el sistema realiza las consultas de información																																																																																										
Inf. De Entrada																																																																																											
Resultado Esperado	El sistema muestra el resultado de la consulta en una tabla																																																																																										
Resultado Obtenido	<table border="1"> <thead> <tr> <th colspan="10">RESULTADOS DE LA BUSQUEDA</th> </tr> <tr> <th>MATRICULA</th> <th>Apellido P</th> <th>Apellido M</th> <th>Nombre</th> <th>Estado</th> <th>Edo Civil</th> <th>Sexo</th> <th>Tel</th> <th colspan="2">Email</th> </tr> </thead> <tbody> <tr> <td>200208330</td> <td>Aguilar</td> <td>Sanchez</td> <td>Adriana</td> <td>Puebla</td> <td></td> <td>F</td> <td>2439919</td> <td colspan="2">adri772@hotmail.com</td> </tr> <tr> <td>200238452</td> <td>FLORES</td> <td>HNDEZ</td> <td>MARISOL</td> <td>Quintana Roo</td> <td>soltera</td> <td>F</td> <td>7564056</td> <td colspan="2">losiram@hotmail.com</td> </tr> </tbody> </table>	RESULTADOS DE LA BUSQUEDA										MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email		200208330	Aguilar	Sanchez	Adriana	Puebla		F	2439919	adri772@hotmail.com		200238452	FLORES	HNDEZ	MARISOL	Quintana Roo	soltera	F	7564056	losiram@hotmail.com																																																			
RESULTADOS DE LA BUSQUEDA																																																																																											
MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email																																																																																			
200208330	Aguilar	Sanchez	Adriana	Puebla		F	2439919	adri772@hotmail.com																																																																																			
200238452	FLORES	HNDEZ	MARISOL	Quintana Roo	soltera	F	7564056	losiram@hotmail.com																																																																																			
Resultado	Satisfactoria																																																																																										

Caso de Experimento	Búsqueda por Rango de Edad																																																																																										
Objetivo	Verificar que el sistema realiza las consultas de información																																																																																										
Inf. De Entrada																																																																																											
Resultado Esperado	El sistema muestra el resultado de la consulta en una tabla																																																																																										
Resultado Obtenido	<table border="1"> <thead> <tr> <th colspan="9">RESULTADOS DE LA BUSQUEDA</th> </tr> <tr> <th>MATRICULA</th> <th>Apellido P</th> <th>Apellido M</th> <th>Nombre</th> <th>Estado</th> <th>Edo Civil</th> <th>Sexo</th> <th>Tel</th> <th>Email</th> </tr> </thead> <tbody> <tr> <td>20034445</td> <td>Flores</td> <td>Fuentes</td> <td>Jose</td> <td>Puebla</td> <td></td> <td>M</td> <td>121454</td> <td>papasuncle</td> </tr> <tr> <td>200122586</td> <td>RODRIGUEZ</td> <td>MORON</td> <td>MISAEI</td> <td>Puebla</td> <td>soltero</td> <td>M</td> <td>2193765</td> <td>guny_1@hotmail.com</td> </tr> <tr> <td>200208330</td> <td>Agular</td> <td>Sanchez</td> <td>Adriana</td> <td>Puebla</td> <td></td> <td>F</td> <td>2439919</td> <td>adri772@hotmail.com</td> </tr> <tr> <td>200238452</td> <td>FLORES</td> <td>HNDEZ</td> <td>MARISOL</td> <td>Quintana Roo</td> <td>soltera</td> <td>F</td> <td>7564056</td> <td>losiram@hotmail.com</td> </tr> <tr> <td>200478525</td> <td>wayne</td> <td>gorbea</td> <td>salseroqui</td> <td>Puebla</td> <td></td> <td>M</td> <td></td> <td></td> </tr> <tr> <td>200534453</td> <td>Montez</td> <td>Rosas</td> <td>Mauro</td> <td>Puebla</td> <td></td> <td>M</td> <td>245825</td> <td>muarinhotmail.com</td> </tr> <tr> <td>200741568</td> <td>Gonzalez</td> <td>Macuitl</td> <td>Robert</td> <td>Guerrero</td> <td></td> <td>M</td> <td>2165454</td> <td>robertp_1@yahoo.com</td> </tr> <tr> <td>200748682</td> <td>Gomez</td> <td>Perez</td> <td>Mamel</td> <td>Oaxaca</td> <td></td> <td>M</td> <td>4848488</td> <td>mande@yahoo.com</td> </tr> </tbody> </table>	RESULTADOS DE LA BUSQUEDA									MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email	20034445	Flores	Fuentes	Jose	Puebla		M	121454	papasuncle	200122586	RODRIGUEZ	MORON	MISAEI	Puebla	soltero	M	2193765	guny_1@hotmail.com	200208330	Agular	Sanchez	Adriana	Puebla		F	2439919	adri772@hotmail.com	200238452	FLORES	HNDEZ	MARISOL	Quintana Roo	soltera	F	7564056	losiram@hotmail.com	200478525	wayne	gorbea	salseroqui	Puebla		M			200534453	Montez	Rosas	Mauro	Puebla		M	245825	muarinhotmail.com	200741568	Gonzalez	Macuitl	Robert	Guerrero		M	2165454	robertp_1@yahoo.com	200748682	Gomez	Perez	Mamel	Oaxaca		M	4848488	mande@yahoo.com
RESULTADOS DE LA BUSQUEDA																																																																																											
MATRICULA	Apellido P	Apellido M	Nombre	Estado	Edo Civil	Sexo	Tel	Email																																																																																			
20034445	Flores	Fuentes	Jose	Puebla		M	121454	papasuncle																																																																																			
200122586	RODRIGUEZ	MORON	MISAEI	Puebla	soltero	M	2193765	guny_1@hotmail.com																																																																																			
200208330	Agular	Sanchez	Adriana	Puebla		F	2439919	adri772@hotmail.com																																																																																			
200238452	FLORES	HNDEZ	MARISOL	Quintana Roo	soltera	F	7564056	losiram@hotmail.com																																																																																			
200478525	wayne	gorbea	salseroqui	Puebla		M																																																																																					
200534453	Montez	Rosas	Mauro	Puebla		M	245825	muarinhotmail.com																																																																																			
200741568	Gonzalez	Macuitl	Robert	Guerrero		M	2165454	robertp_1@yahoo.com																																																																																			
200748682	Gomez	Perez	Mamel	Oaxaca		M	4848488	mande@yahoo.com																																																																																			
Resultado de la Prueba	Satisfactoria																																																																																										

CONCLUSIONES

Conclusiones.

Este proyecto: Sistema Integral para la Administración de Currículos, fue desarrollado con las técnicas de la Ingeniería de software y bases de datos relacionales, basados en el modelo de prototipos y escrito en el Lenguaje Unificado de Modelado e implementado con HTML, PHP y MySQL, y como su nombre lo indica, es un sistema web basado en la arquitectura cliente-servidor.

El sistema, es un producto de software es fiable, y cumple con los objetivos señalados en la fase de análisis y definición del sistema descritas en el capítulo 2, además en cualquier momento que sea necesario, se podrá actualizar, o agregar más funciones, de acuerdo a las nuevas necesidades que se requieran.

El acceso al front-end del sistema depende directamente del gestor MySQL, por lo que la medida de la seguridad de este será la que proporcione MySQL, el cual a pesar de ser un motor gratuito ha probado ser confiable y estable.

Personalmente me llena de satisfacción poder concluir dicho trabajo debido a que ayudará en gran parte a darle una mejor oportunidad de encontrar trabajo o practicas a los alumnos de la facultad de ciencias de la computación.

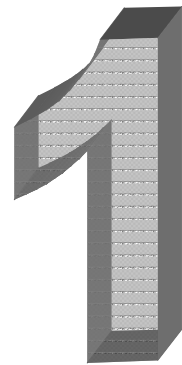
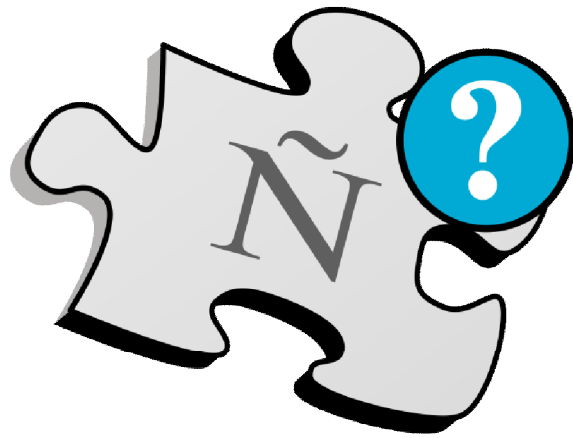
Trabajo a Futuro

- Realizar un modulo en el cual los egresados puedan adjuntar un archivo con su currículo personalizado.
- Realizar una función para almacenar el archivo con la fotografía de el Interesado
- Crear un Filtro de Búsqueda más avanzado que permita encontrar perfiles que se acerquen más a las necesidades de las empresas.
- Gestionar la incorporación de este sistema a la página de la FCC.

BIBLIOGRAFIA

- [1] Ambrosio, A. D. (2007). "Análisis y Diseño de Bases de datos". Puebla: Diplomado en Bases de Datos, Facultad de Ciencias de la computación, BUAP.
- [2] Castaño M., Piattini V. (2001) "Fundamentos y Modelos de Bases de Datos". Madrid: Alfaomega Grupo Editor.
- [3] Cota, A. (1994) "Ingeniería de Software". Soluciones Avanzadas. Julio de 1994.
- [4] Abraham Silberschatz; "Fundamentos de bases de datos". Editorial McGraw-Hill, 2da. Edición 1993, 3ra. Edición 1998 y 5ta. Edición 2006.
- [5] DRA. Ma. De la Concepción Pérez de Céliz; "Ingeniería de Software". Diplomado de Bases de Datos, Otoño 2007.
- [6] Roger S. Presuman; "Ingeniería de Software, un enfoque practico". McGraw-Hill, 4ta. Edición.
- [7] M. C. Yolanda Moyao Martínez; "Introducción a los sistemas de bases de datos". Diplomado de Bases de Datos, Otoño 2007.
- [8] Jacobson, I. (1998). "Applying UML in The Unified Process" Presentación. Rational Software.
Presentación disponible en <http://www.rational.com/uml> como UMLconf.zip
- [9] Ullman (2002) "A first course in database systems" Addison-Wesley, Pearson Education
- [10] Larry Ullman; "MySQL". Prentice May, 1ra. Edición 2004
- [11] Abraham Gutiérrez; "Php 4 a través de ejemplos". Ginés Bravo, Alfa omega, Ra-Ma, 1ra. Edición 2004.
- [12] Jacobo Pavón Puertas; "Creación de un portal con PHP y MySQL". Alfa omega Ra-Ma, 1ra. Edición 2004.

Anexo



Código Fuente

ANEXO: CODIGO FUENTE DEL SISTEMA

Para poder continuar con el trabajo a futuro de este proyecto o bien para que sea de utilidad y de ayuda con fines de aprendizaje es necesario incluir el código fuente de nuestro sistema.

A continuación se incluye el código fuente de algunas funciones del sistema, los lenguajes utilizados son HTML, PHP, SQL, Java Script. Se indicara cual lenguaje que se está utilizando en cada caso para tener un mejor panorama ya que de esta manera podremos comprender qué relación guardan las funciones entre si y como se realizan los accesos a la base de datos.

- Creación de la Base de datos

```
--
-- Estructura de tabla para la tabla `egresados`
--
```

```
CREATE TABLE `egresados` (
  `matricula` int(20) unsigned NOT NULL,
  `apellidop` varchar(20) NOT NULL,
  `apellidom` varchar(20) NOT NULL,
  `Nombre` varchar(20) NOT NULL,
  `edad` tinyint(4) NOT NULL,
  `sexo` varchar(1) NOT NULL,
  `calle` varchar(20) NOT NULL,
  `numint` int(6) default NULL,
  `numext` int(6) default NULL,
  `colonia` varchar(20) NOT NULL,
  `cp` int(6) default NULL,
  `lugarnac` varchar(15) NOT NULL,
  `nacionalidad` varchar(15) NOT NULL,
  `fechanac` varchar(11) default NULL,
  `peso` varchar(4) default NULL,
  `estatura` varchar(5) default NULL,
  `edocivil` varchar(10) NOT NULL,
  `telhome` int(15) default NULL,
  `telcell` varchar(12) default NULL,
  `email` varchar(20) default NULL,
  `email2` varchar(20) default NULL,
  `curp` varchar(20) NOT NULL,
  `afore` varchar(20) default NULL,
  `sar` varchar(15) default NULL,
  `RFC` varchar(20) default NULL,
  `imss` varchar(15) default NULL,
  `nocartilla` varchar(15) default NULL,
  `nopasaporte` varchar(15) default NULL,
  `nocedula` varchar(15) default NULL,
  `idedox` int(5) NOT NULL,
  `idestudiox` int(5) NOT NULL,
  `idareax` int(5) NOT NULL,
```

```

PRIMARY KEY (`matricula`),
KEY `idedox` (`idedox`),
KEY `idestudiox` (`idestudiox`),
KEY `ideareax` (`idareax`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `administrador`
--

CREATE TABLE `administrador` (
  `idadmin` int(5) NOT NULL auto_increment,
  `Nombre` varchar(10) NOT NULL,
  `Contraseña` varchar(10) NOT NULL,
  PRIMARY KEY (`idadmin`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

-- Estructura de tabla para la tabla `areasdeint`
--

CREATE TABLE `areasdeint` (
  `idarea` int(3) NOT NULL auto_increment,
  `descareaa` varchar(20) NOT NULL,
  PRIMARY KEY (`idarea`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

--
-- Estructura de tabla para la tabla `empresa`
--

CREATE TABLE `empresa` (
  `RFC` varchar(15) NOT NULL,
  `nombreemp` varchar(20) NOT NULL,
  `razonsocial` varchar(30) NOT NULL,
  `apellidop` varchar(20) NOT NULL,
  `apellidom` varchar(20) NOT NULL,
  `Nombre` varchar(20) NOT NULL,
  `calle` varchar(20) NOT NULL,
  `numint` int(6) default NULL,
  `numext` int(6) default NULL,
  `colonia` varchar(20) NOT NULL,
  `cp` int(6) default NULL,
  `te1` int(10) NOT NULL,
  `tel2` int(10) default NULL,
  `email` varchar(20) default NULL,
  `email2` varchar(20) default NULL,
  PRIMARY KEY (`RFC`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `ofertas`
--

CREATE TABLE `ofertas` (
  `idoferta` int(5) NOT NULL auto_increment,
  `fechasol` varchar(11) default NULL,
  `perfil` varchar(15) default NULL,

```

```

`requisitos` varchar(40) NOT NULL,
`areaaplicacion` varchar(20) NOT NULL,
`horario` varchar(15) default NULL,
`dispviajar` varchar(4) default NULL,
`idioma` varchar(10) NOT NULL,
`idioma1` varchar(10) default NULL,
`idioma2` varchar(10) default NULL,
`salario` varchar(15) default NULL,
`prestaciones` varchar(15) default NULL,
`fechaentrev` varchar(11) NOT NULL,
`fechainlab` varchar(11) default NULL,
`cantvacante` int(4) default NULL,
`RFCx` varchar(15) NOT NULL,
`idtipox` varchar(10) NOT NULL,
PRIMARY KEY (`idoferta`),
KEY `RFCx` (`RFCx`),
KEY `idtipox` (`idtipox`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

```

```

CREATE TABLE `tipofertas` (
  `idtipo` varchar(10) NOT NULL,
  `desctipo` varchar(50) NOT NULL,
  PRIMARY KEY (`idtipo`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

--
-- Estructura de tabla para la tabla `users`
--

```

```

CREATE TABLE `users` (
  `id` int(4) NOT NULL auto_increment,
  `nick` varchar(30) NOT NULL,
  `pass` varchar(30) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=14 ;

```

- CREACION DE LA PAGINA WEB

El siguiente código crea la página principal del sistema a su vez hace referencia a las distintas secciones que componen la misma.

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Sistema Integral Para La Administración de Currículos</title>
</head>

<frameset rows="" cols="234,*" frameborder="no" border="0" framespacing="0">

```

```

<frame src="izquierda.html" name="leftFrame" scrolling="No" noresize="noresize"
id="leftFrame" title="leftFrame" />
<frameset rows="141,*" cols="*" framespacing="0" frameborder="no" border="0">
  <frame src="Arriba.html" name="topFrame" scrolling="No" noresize="noresize"
id="topFrame" title="topFrame" />
  <frame src="centro.html" name="mainFrame" id="mainFrame" title="Centro" />
</frameset>
</frameset>
<noframes><body>
</body>
</noframes></html>

```

Esta función es la pagina central

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Sistema Integral</title>
<style type="text/css">
<!--

```

```

body {
  text-align:center;
  margin:0 auto;

  url(images/back.gif);
}

```

background-image:

```

#contenedor{
margin: 0 auto;
text-align:left;
width:70%;
}

```

```

.style3 {
  Helvetica, sans-serif;

```

font-family: Verdana, Arial,

font-weight: bold;
color: #FFFFFF;

```

}
.style11 {font-family: Verdana, Arial, Helvetica, sans-serif;
color: #FFFFFF;
font-size: 18px;
font-weight: bold;
}

```

```

.style15 {
  Roman", Times, serif;

```

font-family: Georgia, "Times New
font-size: 12px;

```

}
.style16 {
  Roman", Times, serif;

```

font-family: Georgia, "Times New
font-weight: bold;

```

font-size: 12px;
}
-->
</style></head>

<body>
<p>&nbsp;</p>
<div id="contenedor">
<table width="503" border="0" align="center">
<tr>
<td width="500" height="55" bgcolor="#66CCFF" class="style3"><div
align="center"><span class="style11">BIENVENIDOS A ESTE
PORTAL!</span></div></td>
</tr>
</table>
<table width="503" height="200" border="0" align="center" bordercolor="#66CCFF">
<tr>
<td width="503" height="200"><p align="justify" class="style16"></p>
<p align="justify" class="style16">Prefacio</p>
<p align="justify" class="style15">El SIAC es el Sistema Integral para la
Administracion de Curriculos, es una pagina web dinamica con acceso a una Base de
Datos a traves de la cual los alumnos de la facultad de Ciencias de la Computacion
pueden encontrar ofertas laborales y/o practicas profesionales.</p>
<p align="justify" class="style15">La idea Principal es vincular a nuestros
compañeros Egresados con las Empresas esto debido a la gran competencia que
existe en el mercado en cuanto al area de Informatica. Esta Web estara en contacto con
la Mtra. Alma Delia Ambrosio Vazquez Catedratica de la FCC y quien tiene a cargo
precisamente el departamento de vinculacion por ello sera la Administradora principal de
este sistema.</p>

<p align="justify" class="style15">Estimados alumnos de la FCC y Estimados
Empresarios les deseo la mejor de las suertes y que aqui puedan encontrar esa oferta o
en su caso a esa persona que ustedes necesitan recordandoles que nuestros alumnos
esta preparados para cualquier enmienda y siempre en busca de la Excelencia academica
de la Benemerita Universidad Autonoma De Puebla.</p>
<p class="style15">Atte. Ing en Cs de la Computacion Misael Rodriguez Moron
(WebMaster) </p>
<p class="style15"><a
href="mailto:gumy_1@hotmail.com">email: gumy_1@hotmail.com </a></p>
<p align="justify" class="style16">&nbsp;</p> </td>
</tr>
</table>
<p align="center">&nbsp;</p>
</div>
</body>
</html>

```

- **PARTE IZQUIERDA DE LA PAGINA LA CUAL CONTIENE EL MENU PRINCIPAL**

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

```

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Sistema Integral Para la Administracion de curriculos</title>
```

```
<script type="text/javascript" language="javascript">
```

ESTA FUNCION ES MUY IMPORTANTE YA QUE VALIDA LOS FORMULARIOS

```
function validar(){
var forma = document.forms[0];
if (forma.usuario.value==""){
alert("Debe Escribir su Nombre de
Usuario");
return false;
}
if (forma.password.value==""){
alert("Debe Escribir su
Contraseña");
return false;
}
return true;
}
function ejecutar(){
var forma = document.forms[0];
if (validar()) forma.submit();
}
</script>
<style type="text/css">
<!--
body {
background-image:
url(images/layout_30.gif);
}
-->
</style></head>
```

```
<body>
<table width="226" height="462" border="0">
<tr>
<td width="220" height="202" valign="top"><div align="center">
<p></p>
<p></p>
</div></td>
</tr>
<tr>
<td height="90" valign="top">
<script language="JavaScript" type="text/javascript" src="./vertical_scr.js">
</script>
```

```

<table border='0' align="center" cellpadding='0' cellspacing='0'>
  <tr><td></td>
</tr>
<tr><td><a
                        href="centro.html"
                        target="mainFrame"
onmouseover="exM(m1,'none','",event);chgBg(m1,'m1t1m0',3)"
onmouseout="chgBg(m1,'m1t1m0',0,1)"></a></td>
</tr>
<tr><td></td>
</tr>
<tr><td><a
                        href="javascript:void(0);"
onmouseover="exM(m1,'m1mn1','m1t1m1',event);chgBg(m1,'m1t1m1',3)"
onmouseout="coM(m1,'m1mn1','m1t1m1');chgBg(m1,'m1t1m1',0)"></a></td>
</tr>
<tr><td></td>
</tr>
<tr><td><a
                        href="javascript:void(0);"
onmouseover="exM(m1,'m1mn2','m1t1m2',event);chgBg(m1,'m1t1m2',3)"
onmouseout="coM(m1,'m1mn2','m1t1m2');chgBg(m1,'m1t1m2',0)"></a></td>
</tr>
<tr><td></td>
</tr>
<tr><td><a
                        href="Personalizada.html"
                        target="mainFrame"
onmouseover="exM(m1,'m1mn3','m1t1m3',event);chgBg(m1,'m1t1m3',3)"
onmouseout="coM(m1,'m1mn3','m1t1m3');chgBg(m1,'m1t1m3',0)"></a></td>
</tr>
<tr><td></td>
</tr>
<tr><td><a
                        href="email.php"
                        target="mainFrame"
onmouseover="exM(m1,'none','",event);chgBg(m1,'m1t1m4',3)"
onmouseout="chgBg(m1,'m1t1m4',0,1)"></a></td>
</tr>
<tr><td></td>
</tr>
<tr><td><a
                        href="http://www.cs.buap.mx"
                        target="mainFrame"
onmouseover="exM(m1,'none','",event);chgBg(m1,'m1t1m5',3)"
onmouseout="chgBg(m1,'m1t1m5',0,1)"></a></td>
</tr>
<tr><td></td>
</tr>
</table>
<script language="JavaScript">
<!--

// -----
// Add items to the menu here
// -----

```

```

var e_1=getTopLevelMenu(1);
var e_2=getTopLevelMenu(2);
var e_3=getTopLevelMenu(3);
addMenuItem ("Egresados","egresado.html","mainFrame",e_1,"");
addMenuItem ("Empresas","Empresas.html","mainFrame",e_1,"");
addMenuItem ("Ofertas","Ofertas.html","mainFrame",e_1,"");
addMenuItem ("Eliminar","borraegresado.php","mainFrame",e_1,"");
addMenuItem ("Laborales","Laborales.php","mainFrame",e_2,"");
addMenuItem ("Practicas","Practicas.php","mainFrame",e_2,"");
addMenuItem ("Personalizada","Personalizada.html","mainFrame",e_3,"");
//-->
</script> </td>
</tr>
<tr>
<td height="155" valign="top">
<p></p>
<form id="form1" name="form1" method="post" action="login.php"
target="mainFrame">
<table width="215" border="0" align="left">
<tr>
<td width="84" height="30">Usuario</td>
<td width="50"><input name="usuario" type="text" size="15" /></td>
</tr>
<tr>
<td width="84" height="30">Contrase&ntilde;a</td>
<td width="50"><input name="password" type="password" id="password"
size="15" /></td>
</tr>
<tr>
<td height="30">&nbsp;</td>
<td><input type="button" onclick="ejecutar()" name="accion" id="accion"
value="Enviar" /></td>
</tr>
</table>
</form> </td>
</tr>
</table>
</body>
</html>

```

ESTA FUNCION CREA LA PAGINA DE REGISTRO DE DATOS DE EL EGRESADO

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Egresados</title>

<script type="text/javascript" language="javascript">

function validar(){

var forma = document.forms[0];

```

```

if (forma.matricula.value==""){
    alert("Debe Escribir su Matricula");
    return false;
}
if (forma.Nombre.value==""){
    alert("Debe Escribir su Nombre");
    return false;
}
if (forma.apellidop.value==""){
    alert("Debe Escribir su apellido
    Paterno");
    return false;
}
if (forma.apellidom.value==""){
    alert("Debe Escribir su apellido
    Materno ");
    return false;
}
if (forma.calle.value==""){
    alert("Debe el nombre de su
    calle");
    return false;
}
if (forma.telhome.value==""){
    alert("Debe ingresar un Numero");
    return false;
}

if (forma.email.value==""){
    alert("Escriba su direccion de
    correo por ejemplo gumy@hotmail.com");
    return false;
}

return true;
}

function ejecutar(){
    var forma = document.forms[0];
    if (validar()) forma.submit();
}
</script>

<style type="text/css">
<!--
body {
    background-color: #FFFFFF;
    background-image:
    url(images/back.gif);
    background-repeat: repeat;
}

```

```

.style3 {font-family: "Courier New", Courier, monospace; }
.style4 {font-family: "Courier New", Courier, monospace; font-size: small; }
.style9 {color: #FFFFFF; font-family: Geneva, Arial, Helvetica, sans-serif; }
.style11 {
                                font-family:    Verdana,    Arial,
Helvetica, sans-serif;
                                color: #FFFFFF;
                                font-size: 18px;
                                font-weight: bold;
}
.style14 {
                                font-family:    Verdana,    Arial,
Helvetica, sans-serif;
                                font-weight: bold;
}
.style15 {font-family: Georgia, "Times New Roman", Times, serif; }
-->
</style></head>

<body>
<div align="center">
  <table width="500" height="800" border="1" bordercolor="#6699FF">
    <tr>
      <td width="468" height="55" bgcolor="#66CCFF"><div align="center"
class="style11">REGISTRA TUS DATOS </div></td>
    </tr>
    <tr>
      <td height="741" valign="top" bgcolor="#FFFFFF"><form id="form1" name="form1"
method="GET" action="insertaegresados.php">
        <table width="447" border="0" align="center">
          <tr>
            <td height="75" class="style14"><div align="left">Datos Personales </div></td>
            <td><div align="left"></div></td>
          </tr>
          <tr>
            <td width="200" height="24" class="style15"><div
align="left">Matricula</div></td>
            <td width="237"><div align="left">
              <input type="text" name="matricula" />
            </div></td>
          </tr>
          <tr>
            <td height="23" class="style15"><div align="left">Apellido P</div></td>
            <td><div align="left">
              <input type="text" name="apellidop" />
            </div></td>
          </tr>
          <tr>
            <td height="23" class="style15"><div align="left">Apellido Mat </div></td>
            <td><div align="left">
              <input type="text" name="apellidom" />
            </div></td>
          </tr>
          <tr>

```

```

<td height="24" class="style15"><div align="left">Nombre(s)</div></td>
<td><div align="left">
  <input name="Nombre" type="text" size="30" />
</div></td>
</tr>
<tr>
<td height="24" class="style15"><div align="left">Edad</div></td>
<td><div align="left">
  <select name="edad">
    <option value="0">Selecciona tu Edad</option>
    <option value="18">18</option>
    <option value="19">19</option>
    <option value="20">20</option>
    <option value="21">21</option>
    <option value="22">22</option>
    <option value="23">23</option>
    <option value="24">24</option>
    <option value="25">25</option>
    <option value="26">26</option>
    <option value="27">27</option>
    <option value="28">28</option>
    <option value="29">29</option>
    <option value="30">30</option>
    <option value="31">31</option>
    <option value="32">32</option>
    <option value="33">33</option>
    <option value="34">34</option>
    <option value="35">35</option>
    <option value="36">36</option>
    <option value="37">37</option>
    <option value="38">38</option>
    <option value="39">39</option>
    <option value="40">40</option>
    <option value="41">41</option>
    <option value="42">42</option>
    <option value="43">43</option>
    <option value="44">44</option>
    <option value="45">45</option>
    <option value="46">46</option>
    <option value="47">47</option>
    <option value="48">48</option>
    <option value="49">49</option>
    <option
value="50">50</option>
  </select>
</div></td>
</tr>
<tr>
<td height="24" class="style15"><div align="left">Sexo</div></td>
<td><label>
  <div align="left">
    <input type="radio" name="sexo" value="M" />
    Masculino</div>
<div align="left">
    <input type="radio" name="sexo" value="F" />

```

```

        Femenino      </div>
      </label></td>
    </tr>
    <tr>
      <td height="81" class="style14"><div align="left">Direccion</div></td>
      <td><div align="left"></div></td>
    </tr>
    <tr>
      <td height="23" class="style15"><div align="left">Calle</div></td>
      <td><div align="left">
        <input name="calle" type="text" size="30" />
      </div></td>
    </tr>
    <tr>
      <td height="24" class="style15"><div align="left">Num Int</div></td>
      <td><div align="left">
        <input name="numint" type="text" size="15" />
      </div></td>
    </tr>
    <tr>
      <td height="24" class="style15"><div align="left">Num Ext </div></td>
      <td><div align="left">
        <input name="numint2" type="text" size="15" />
      </div></td>
    </tr>
    <tr>
      <td height="21" class="style15"><div align="left">Colonia</div></td>
      <td><div align="left">
        <input type="text" name="colonia" />
      </div></td>
    </tr>
    <tr>
      <td height="23" class="style15"><div align="left">C. P. </div></td>
      <td><div align="left">
        <input name="cp" type="text" size="15" />
      </div></td>
    </tr>
    <tr>
      <td height="23" class="style15"><div align="left">Lugar de nacimiento</div></td>
      <td><div align="left">
        <input type="text" name="lugarnac" />
      </div></td>
    </tr>
    <tr>
      <td height="26" class="style15"><div align="left">Nacionalidad</div></td>
      <td><div align="left">
        <input type="text" name="nacionalidad" />
      </div></td>
    </tr>
    <tr>
      <td height="84" class="style14"><div align="left">Datos De Interes </div></td>
      <td><div align="left"></div></td>
    </tr>

```

```

<tr>
  <td height="23" class="style15"><div align="left">Fecha de Nacimiento</div></td>
  <td><div align="left">
    <input name="fechanac" type="text" size="20" />
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Peso</div></td>
  <td><div align="left">
    <input name="peso" type="text" size="20" />
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Estatura:</div></td>
  <td><div align="left">
    <input name="estatura" type="text" size="20" />
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Edo. Civil </div></td>
  <td><div align="left">
    <select name="select2">
      <option value="0">Selecciona tu Edo Civil</option>
      <option value="1">Soltero</option>
      <option value="2">Casado</option>
      <option value="3">Divorciado</option>
      <option value="4">Viudo</option>
    </select>
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Tel Casa: </div></td>
  <td><div align="left">
    <input name="telhome" type="text" size="20" />
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Celular:</div></td>
  <td><div align="left">
    <input name="telcell" type="text" size="20" />
  </div></td>
</tr>
<tr>
  <td height="24" class="style15"><div align="left">Email </div></td>
  <td><div align="left">
    <input name="email" type="text" size="30" />
  </div></td>
</tr>
<tr>
  <td height="23" class="style15"><div align="left">Email Alterno </div></td>
  <td><div align="left">
    <input name="email2" type="text" size="30" />
  </div></td>
</tr>
<tr>

```

```

        <td height="74" class="style14"><div align="left">Documentacion</div></td>
        <td><div align="left"></div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">Curp:</div></td>
        <td><div align="left">
            <input type="text" name="curp" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">Afore:</div></td>
        <td><div align="left">
            <input type="text" name="afore" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">Sar:</div></td>
        <td><div align="left">
            <input type="text" name="sar" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">RFC:</div></td>
        <td><div align="left">
            <input type="text" name="RFC" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">Imss:</div></td>
        <td><div align="left">
            <input type="text" name="imss" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">No. Cartilla </div></td>
        <td><div align="left">
            <input type="text" name="nocartilla" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">No. Pasaporte </div></td>
        <td><div align="left">
            <input type="text" name="nopasaporte" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">No. de Cedula </div></td>
        <td><div align="left">
            <input type="text" name="nocedula" />
        </div></td>
    </tr>
    <tr>
        <td height="23" class="style15"><div align="left">Estado</div></td>
        <td><div align="left">

```

```

<select name="idedox">
  <option value="0">Selecciona tu Estado</option>
  <option value="1">Aguas calientes</option>

  <option value="2">Baja California Norte</option>

  <option value="3">Baja California Sur</option>
  <option value="4">Campeche</option>
  <option value="5">Chiapas</option>
  <option value="6">Chihuahua</option>
  <option value="7">Coahuila</option>
  <option value="8">Colima</option>
  <option value="9">Distrito Federal</option>
  <option value="10">Durango</option>
  <option
value="11">Estado de Mexico</option>
  <option value="12">Guanajuato</option>
  <option value="13">Guerrero</option>
  <option value="14">Hidalgo</option>
  <option value="15">Jalisco</option>
  <option value="16">Michoacan</option>
  <option value="17">Morelos</option>
  <option value="18">Nayarit</option>
  <option value="19">Nuevo Leon</option>
  <option value="20">Oaxaca</option>
  <option
value="21">Puebla</option>
  <option value="22">Queretaro</option>
  <option value="23">Quintana Roo</option>
  <option value="24">San Luis Potosi</option>
  <option value="25">Sinaloa</option>
  <option value="26">Sonora</option>
  <option value="27">Tabasco</option>
  <option value="28">Tamaulipas</option>
  <option value="29">Tlaxcala</option>
  <option value="30">Veracruz</option>
  <option value="31">Yucatan</option>
  <option value="32">Zacatecas</option>
</select>
</div></td>
</tr>
<tr>
<td height="23" class="style15"><div align="left">Carrera</div></td>
<td><div align="left">
  <select name="idestudiox">
    <option value="0">Selecciona tu Carrera</option>
    <option value="1">Ingenieria en Cs. de la Computacion</option>
    <option value="2">Licenciatura en Cs. de la Computacion</option>
  </select>
</div></td>
</tr>
<tr>
<td height="23" class="style15"><div align="left">Area de Interes</div></td>
<td><div align="left">
  <select name="idareax">

```



```

font-weight: bold;
}
.style15 {font-family: Georgia, "Times New Roman", Times, serif; }
.style17 {font-family: "Courier New", Courier, monospace; font-size: x-small; }
.style12 {font-family: Verdana, Arial, Helvetica, sans-serif; color: #000000; font-size: 14px;
font-weight: bold; }
-->
</style></head>

<body>
<?php
include("conexion.php");
$link=conectarse();
$matricula=$_GET['matricula'];
$apellidop=$_GET['apellidop'];
$apellidom=$_GET['apellidom'];
$Nombre=$_GET['Nombre'];
$edad=$_GET['edad'];
$sexo=$_GET['sexo'];
$calle=$_GET['calle'];
$numint=$_GET['numint'];
$numext=$_GET['numext'];
$colonia=$_GET['colonia'];
$cp=$_GET['cp'];
$lugarnac=$_GET['lugarnac'];
$nacionalidad=$_GET['nacionalidad'];
$fechanac=$_GET['fechanac'];
$peso=$_GET['peso'];
$estatura=$_GET['estatura'];
$edocivil=$_GET['edocivil'];
$telhome=$_GET['telhome'];
$telcell=$_GET['telcell'];
$email=$_GET['email'];
$email2=$_GET['email2'];
$curp=$_GET['curp'];
$afore=$_GET['afore'];
$sar=$_GET['sar'];
$RFC=$_GET['RFC'];
$imss=$_GET['imss'];
$nocartilla=$_GET['nocartilla'];
$nopasaporte=$_GET['nopasaporte'];
$nocedula=$_GET['nocedula'];
$idedox=$_GET['idedox'];
$idestudiox=$_GET['idestudiox'];
$idareax=$_GET['idareax'];
mysql_query("insert
                                                    into
egresados(matricula,apellidop,apellidom,Nombre,sexo,calle,numint,numext,colonia,cp,lug
arnac,nacionalidad,fechanac,peso,estatura,edocivil,telhome,telcell,email,email2,curp,afor
e,sar,RFC,imss,nocartilla,nopasaporte,nocedula,idedox) values

('$matricula','$apellidop','$apellidom','$Nombre','$sexo','$calle','$numint','$numext','$colonia',
'$cp','$lugarnac','$nacionalidad','$fechanac','$peso','$estatura','$edocivil','$telhome','$t
elcell','$email','$email2','$curp','$afore','$sar','$RFC','$imss','$nocartilla','$nopasaporte',
'$nocedula','$idedox','$idestudiox','$idareax')",$link);

```

```

//header("Location:contraseña.html");
?>
<div align="center">
  <table width="500" height="100" border="1" bordercolor="#6699FF">
    <tr>
      <td width="468" height="55" bgcolor="#66CCFF"><div align="center"
class="style11">COMPLETA TU REGISTRO </div></td>
    </tr>
    <tr>
      <td height="150" valign="top" bgcolor="#FFFFFF"><form id="form1" name="form1"
method="POST" action="botregister.php">
        <table width="495" border="0">
          <tr>
            <td height="53" class="style12">Obten tu ID y tu Password</td>
            <td>&nbsp;</td>
          </tr>
          <tr>
            <td width="319"><p><span class="style15">Escribe una Alias</span><span
class="style17">(maximo 10 caracteres) </span></p> </td>
            <td width="166"><input type="text" name="Nick" /></td>
          </tr>
          <tr>
            <td class="style15">Escribe tu Contrase&ntilde;a </td>
            <td><input type="text" name="Pass" /></td>
          </tr>
          <tr>
            <td>&nbsp;</td>
            <td><input type="submit" value="Guardar" name="Send">
              <input type="reset" name="Submit2" value="Cancelar" /></td>
          </tr>
        </table>
      </form> </td>
    </tr>
  </table>
<div align="center" class="style4">&nbsp;</div>
<p class="style3">&nbsp;</p>

</body>
</html>

```

ESTA ES LA FUNCION DE BUSQUEDA

```

<html>
<head>
<title> Busqueda </title>
</head>

<style type="text/css">
<!--
body {

background-color: #FFFFFF;
background-image: url(back.gif);
background-repeat: repeat;

}
.style3 {font-family: "Courier New", Courier, monospace; }
.style4 {font-family: "Courier New", Courier, monospace; font-size: small; }
.style9 {color: #FFFFFF; font-family: Geneva, Arial, Helvetica, sans-serif; }
.style11 {
font-family: Verdana, Arial, Helvetica,
sans-serif;

color: #000000;
font-size: 10px;
font-weight: bold;

}
.style14 {
font-family: Verdana, Arial, Helvetica,
sans-serif; font-size: 18px;
}
.style15 {font-family: Georgia, "Times New Roman", Times, serif; }
-->
</style></head>
<body>
<div align="center"></div>
<p align="center" class="style2">
<?php

include("conexion.php");
$link=conectarse();
$buscar=$_POST['matricula'];
$buscar1=$_POST['Nombre'];
$buscar2=$_POST['idedox'];
$buscar3=$_POST['sexo'];
$buscar4=$_POST['idestudiox'];
$buscar5=$_POST['idareax'];
$buscar6=$_POST['edad'];

$edad2=30;

/*
for ($i = 0; $i >= 9; $i++)
{

$buscar5[$i]=$_POST['edad'];
$_POST['edad']=$_POST['edad']+1;
$sql5="SELECT * FROM egresados,estado WHERE edad LIKE '$buscar5[i]'and idedo=idedox
ORDER BY
matricula";
}
*/

```

```
$sql="SELECT * FROM egresados,estado WHERE matricula LIKE '$buscar' and idedo=idedox
ORDER BY
matricula";
```

```
$sql2="SELECT * FROM egresados,estado WHERE idedo LIKE '$buscar2' and idedox=idedo
ORDER BY
matricula";
```

```
$sql4="SELECT * FROM egresados,estado WHERE sexo LIKE '$buscar3' and idedox=idedo
ORDER BY
matricula";
```

```
$sql3="SELECT * FROM egresados,estudios,estado WHERE idestudio LIKE '$buscar4'and
idestudiox=idestudio and idedo=idedox ORDER BY
matricula";
```

```
$sql5="SELECT * FROM egresados,areasdeiint,estado WHERE idarea LIKE '$buscar5'and
idareax=idarea and idedo=idedox ORDER BY
matricula";
```

```
if($buscar6==20)
{
$sql6="SELECT * FROM egresados,estado WHERE(edad>20 and edad<31)AND idedo=idedox
ORDER BY matricula";
}
```

```
else if($buscar6==31)
{
$sql7="SELECT * FROM egresados,estado WHERE(edad>30 AND edad <41)AND idedo=idedox
ORDER BY matricula";
}
```

```
else if($buscar6==41)
{
$sql8="SELECT * FROM egresados,estado WHERE(edad>40 AND edad <51)AND idedo=idedox
ORDER BY matricula";
}
```

```
$result=mysql_query($sql,$link) ;
$result1=mysql_query($sql2,$link);
$result2=mysql_query($sql3,$link);
$result3=mysql_query($sql4,$link);
$result4=mysql_query($sql5,$link);
$result5=mysql_query($sql6,$link);
$result6=mysql_query($sql7,$link);
$result7=mysql_query($sql8,$link);
```

```
?>
```

```
<table width="800" border="0" align="center">
<tr>
<td width="800" height="55" bgcolor="#66CCFF" class="style3"><div align="center"><span
class="style14">RESULTADOS DE LA BUSQUEDA</span></div></td>
</tr>
</table>
```

```
<TABLE width="800" BORDER=1 align="center" CELSPACING=1 CELLPADDING=1>
<TR>
```

```

        <td height="20" width="100" class="style11" align="center"><B>MATRICULA</B></TD><td
height="20" width="80" class="style11" align="center"><B>Apellido P</B></TD><td height="20"
width="80" class="style11" align="center"><B>Apellido M</B></TD><td height="20" width="100"
class="style11" align="center"><B>Nombre</B></TD><td height="20" width="60" class="style11"
align="center"><B>Estado</B></TD><td height="20" width="80" class="style11"
align="center"><B>Edo Clvil</B></TD><td height="20" width="15" class="style11"
align="center"><B>Sexo</B></TD><td height="20" width="50" class="style11"
align="center"><B>Tel</B></TD><td height="20" width="100" class="style11"
align="center"><B>Email</B></TD>

```

```

    </TR>
<?php
while($row=@mysql_fetch_array($result))
{

printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</t
d><td>%s</td><td>%s
</td></tr>",

$row["matricula"],$row["apellidop"],$row["apellidom"],$row["Nombre"],$row["nomedo"],$row["edoci
vil"],$row["sexo"],$row["telhome"],$row["email"]);
}
while($row=@mysql_fetch_array($result1))
{

printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</t
d><td>%s</td><td>%s
</td></tr>",

$row["matricula"],$row["apellidop"],$row["apellidom"],$row["Nombre"],$row["nomedo"],$row["edoci
vil"],$row["sexo"],$row["telhome"],$row["email"]);
}

while($row=@mysql_fetch_array($result2))
{

printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</t
d><td>%s</td><td>%s
</td></tr>",

$row["matricula"],$row["apellidop"],$row["apellidom"],$row["Nombre"],$row["nomedo"],$row["edoci
vil"],$row["sexo"],$row["telhome"],$row["email"]);
}

while($row=@mysql_fetch_array($result3))
{

printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</t
d><td>%s</td><td>%s
</td></tr>",

$row["matricula"],$row["apellidop"],$row["apellidom"],$row["Nombre"],$row["nomedo"],$row["edoci
vil"],$row["sexo"],$row["telhome"],$row["email"]);
}

while($row=@mysql_fetch_array($result4))
{

printf("<tr><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</td><td>%s</t
d><td>%s</td><td>%s

```

