



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

FACULTAD DE INGENIERÍA

**TESIS**

**SISTEMA PARA EL CONTROL Y ADMINISTRACIÓN DE  
LA INFORMACIÓN DE LA COORDINACIÓN DE  
SEMINARIOS Y SERVICIO SOCIAL DE LA DIE.**

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

PRESENTAN:

BUCIO DELGADO JESÚS

GARCÍA MORALES KARINA

**DIRECTOR DE TESIS:**

**M. I. AURELIO ADOLFO MILLÁN NÁJERA**

CIUDAD UNIVERSITARIA AGOSTO 2012.





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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

# AGRADECIMIENTOS

---

A todas las personas que participaron e hicieron posible este proyecto, muchas gracias por su apoyo y enseñanzas.

A la Coordinación de Seminarios y Servicio Social de la DIE por apoyar nuestra idea y permitirnos la realización de esta tesis.

Jesús Bucio Delgado y Karina García Morales.

# PRÓLOGO

---

Esta tesis tiene como objetivo analizar, diseñar y desarrollar un sistema para administrar la información de la Coordinación de Seminarios y Servicio Social de la División de Ingeniería Eléctrica (DIE), con el fin de hacer más ágiles los trámites y hacer más eficiente la atención que brinda el personal que labora en la misma.

El software se ha vuelto un factor determinante para las empresas, bancos, universidades, entidades federales, etc.

El uso de los sistemas hoy en día facilita el manejo de información desde una pequeña hasta una gran cantidad de la misma evitando pérdidas.

A continuación se dará una breve explicación de lo que se va a ver en cada uno de los capítulos de esta tesis.

En el capítulo I, se dará una introducción a las tecnologías de la información, se mostrarán las actividades y etapas del desarrollo de un sistema, cómo es que la arquitectura cliente/servidor juega un papel muy importante en el diseño, las bases de datos representan la parte fundamental para el desarrollo de un sistema que necesita compartir información, se realizará un análisis de algunas metodologías como son: metodología en cascada, estructurada, prototipo, incremental y espiral.

En el capítulo II, Análisis de los requerimientos del software, se encontrarán especificados los requerimientos del sistema, en esta etapa es muy importante que no se omita ninguna característica u operación necesaria para dicho sistema, sólo se encarga de analizar qué es lo que requiere el usuario final, esta parte específica claramente que objetivos debe cumplir el sistema.

En el capítulo III, Diseño de la base de datos del sistema, se definirán a detalle los requerimientos del sistema, considérese diseño lógico, en donde se definirán las tablas que serán necesarias en el diseño de la base de datos, éstas ayudan a dividir y organizar los elementos del sistema y la definición de los módulos, en los cuales se especificarán qué tipo de usuarios utilizarán el sistema y qué funciones podrán o no realizar.

El capítulo IV, Diseño de pantallas del sistema, se mostrará la interfaz del sistema y cada una de las pantallas explicando paso a paso la salida de cada una, dependiendo de lo que el usuario especificó que requería en la parte de análisis de requerimientos y complementándolo con la etapa de diseño de la base de datos del sistema.

El capítulo V, Codificación, se implementará el diseño del sistema a código fuente, aquí se presentan las principales funciones del sistema, las cuales servirán como plantillas para cada una de las partes del sistema dependiendo de lo que requiera que haga el sistema con la información.

El capítulo VI, es la descripción de la fase de pruebas, en él se realizarán diferentes pruebas al sistema con el fin de comprobar que el sistema funciona eficientemente o detectar las fallas que éste pudiera presentar para obtener como resultado un sistema de calidad.

En el capítulo VII, se llevará a cabo la implantación del sistema, en éste se dará un informe detallado de los equipos que intervendrán, el software que será necesario instalar en el servidor, la creación de la base de datos paso a paso, la migración de la información existente que será necesaria y es de suma importancia evitar pérdidas.

Habiendo completado correctamente cada una de estas etapas se pretende que el sistema funcione con eficiencia y óptimamente, beneficiando con esto tanto al personal que labora en la Coordinación de Seminarios y Servicio Social de la DIE, así como a los usuarios que requieran de algún servicio en la misma.

# ÍNDICE

---

I.	Introducción a las tecnologías de la información.....	1
	I.I Etapas de desarrollo de un sistema de información.....	4
	I.II Arquitectura cliente/servidor.....	6
	I.III Introducción a las bases de datos.....	9
	I.IV Marco tecnológico.....	24
II.	Análisis de los requerimientos del software.....	28
	II.I Especificación de requerimientos.....	30
	II.II Casos de uso.....	33
III.	Diseño de la base de datos del sistema.....	34
	III.I Diccionario de datos.....	35
	III.II Definición de módulos.....	42
IV.	Diseño de pantallas del sistema .....	43
V.	Codificación.....	101
	V.I Anexo 2. Conexión a la base de datos.....	102
	V.II Anexo 3. Insertar un registro nuevo en una tabla.....	103
VI.	Descripción de la fase de pruebas.....	106
	VI.I Pruebas al sistema.....	108
VII.	Implantación del sistema.....	112
	VII.I Requerimientos mínimos de software y hardware.....	113
	Conclusiones.....	122
		Anexos:
	Anexo 1. Diagrama del modelo entidad-relación.....	123
	Anexo 4. Insertar registros simultáneamente a dos tablas distintas en la base de datos.....	125
	Anexo 5. Eliminación de un registro de una tabla.....	129

Anexo 6. Modificación de datos de un registro en una tabla.....	131
Anexo 7. Consultar datos de un registro en una tabla.....	134
Anexo 8. Combinar registros de dos o más tablas en una base de datos.....	138
Anexo 9. Impresión de datos de un registro de una tabla.....	142
Anexo 10. Mostrar todos los datos de los registros que se realizaron en una tabla.....	147
Bibliografía.....	151
Mesografía.....	152





**CAPÍTULO I.**  
**INTRODUCCIÓN A LAS TECNOLOGÍAS DE**  
**INFORMACIÓN.**

## Capítulo I. Introducción a las tecnologías de información.

Los sistemas de información (SI) y las tecnologías de información (TI) han cambiado la forma en que operan las organizaciones actuales. A través de su uso se logran importantes mejoras, pues automatizan los procesos operativos, suministran una plataforma de información necesaria para la toma de decisiones y lo más importante, su implantación logra ventajas competitivas o reduce la ventaja de los rivales.<sup>1</sup>

Las tecnologías de la información han sido conceptualizadas como la integración y convergencia de la computación, las telecomunicaciones y la técnica para el procesamiento de datos, donde sus principales componentes son: el factor humano, los contenidos de la información, el equipamiento, la infraestructura, el software y los mecanismos de intercambio de información, los elementos de política y regulaciones, además de los recursos financieros.<sup>2</sup>

### Las tecnologías de información agrupan cuatro áreas tecnológicas que convergen:

- La microelectrónica: Basa su actividad en el diseño de circuitos integrados, cada vez de mayor capacidad y velocidad de transmisión.
- La computación: Dedicada a la fabricación de hardware que permite el almacenamiento y proceso de los datos.
- La informática: Dedicada al software, sistemas informáticos, que reúnen una serie de procedimientos, orientados a solucionar actividades específicas relacionadas con el procesamiento de la información.
- Las telecomunicaciones: Una gigantesca industria dado sus alcances y avances, son los que posibilitan la integración de los sistemas, mediante el uso de redes locales, amplias o mundiales.<sup>3</sup>

### Definición de sistemas de información

Un sistema de información es un conjunto de elementos interrelacionados, con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones.<sup>4</sup>

---

<sup>1</sup> <http://www.gestiopolis.com/Canales4/mkt/simparalas.htm>

<sup>2</sup> [Ibidem](#)

<sup>3</sup> Cohen, D. y Asin, "Sistemas de información para los negocios

<sup>4</sup> Peña Ayala, Alejandro, "Ingeniería de Software: Una Guía para Crear Sistemas de Información", D.R. © 2006

## **Actividades que realizan los sistemas de información.**

**Entrada de información:** Es el proceso mediante el cual el sistema de información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que proviene o es tomada de otros sistemas o módulos. Esto último se denomina interfaces automáticas. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, los discos magnéticos, las unidades de diskette, los códigos de barras, los escáners, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.<sup>5</sup>

**Almacenamiento de información:** El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros, las cintas magnéticas, los discos flexibles o diskettes, los discos ópticos (CD, DVD, CD-ROM, CD-RW, DVD-RW), las memorias USB y las memorias flash, entre otras.

**Procesamiento de información:** Es la capacidad del sistema de información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecidas. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.<sup>6</sup>

**Salida de información:** La salida es la capacidad de un sistema de información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, diskettes, cintas magnéticas, la voz, los graficadores y los plotters, entre otros. Es importante aclarar que la salida de un sistema de información puede constituir la entrada a otro sistema de información o módulo. En este caso, también existe una interfaz automática de salida.<sup>7</sup>

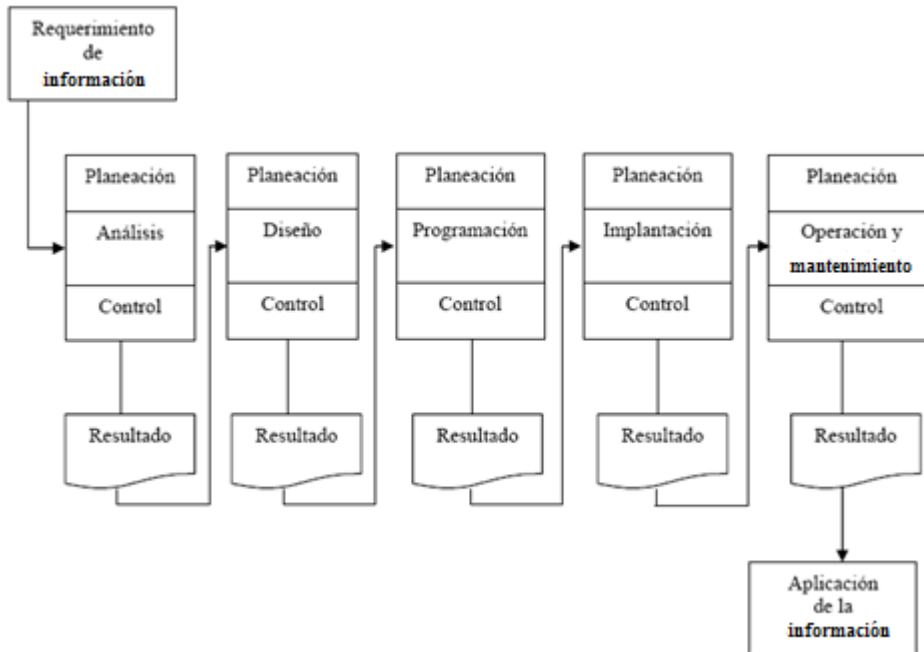
---

<sup>5</sup> Medina Peralta, Manolo, "Sistema de Información de los Servicios de Atención Primaria. La experiencia 2006-2008 del Institut Català de la Salut," *Revista de Innovación Sanitaria y Atención Integrada*: Vol. 1: Iss. 1, Article 7.

<sup>6</sup> Ibídem.

<sup>7</sup> Ibídem.

## I.1 Etapas de desarrollo de un sistema de información.



### Determinación de los requerimientos de información.

La primera etapa que aborda el analista es la determinación de los requerimientos de información a partir de los usuarios particularmente involucrados. Para identificar los requerimientos de información dentro de la empresa, pueden utilizarse diversos instrumentos, los cuales incluyen: el muestreo, el estudio de los datos y formas usadas por la organización, la entrevista, los cuestionarios, la observación de la conducta de quién toma las decisiones, así como de su ambiente y también el desarrollo de prototipos.

### Análisis de las necesidades del sistema.

Las siguientes etapas que ejecuta el analista consisten en analizar las necesidades propias del sistema. Una vez más existen herramientas y técnicas especiales que facilitan al analista la realización de las determinaciones requeridas. Éstas incluyen el uso de los diagramas de flujo de datos que cuentan con una técnica estructurada para representar en forma gráfica la entrada de datos de la empresa, los procesos y salidas de la información. A partir del diagrama de flujo de datos se desarrolla un diccionario de datos que contiene todos los elementos que utiliza el sistema, así como sus especificaciones, si son alfanuméricos y espacios de impresión que requiere. Durante esta fase, el analista de sistemas también utiliza las decisiones estructuradas por realizar, que son decisiones donde las condiciones, acciones y reglas de acción podrán determinarse.

Existen tres métodos para el análisis de las decisiones estructuradas: lenguaje estructurado (en nuestro caso el español) las tablas de decisiones y los árboles de decisiones.<sup>8</sup>

### **Diseño del sistema.**

En esta etapa del ciclo de desarrollo de los sistemas, el analista de sistemas usa la información que recolectó con anterioridad y elabora el diseño lógico del sistema de información. El analista también diseña accesos efectivos al sistema de información, mediante el uso de las técnicas de diseño de forma y pantallas.

Una parte de diseño lógico del sistema de información es el diseño de la interfaz para el usuario con el sistema y evidentemente, es de suma importancia. Serían ejemplos de interfaz para el usuario: el uso del teclado para introducir preguntas o respuestas, el uso de menú en la pantalla, con las opciones que tiene el usuario, el uso de dispositivos como el ratón y muchos otros.<sup>9</sup>

### **Desarrollo y documentación de software.**

En la sexta etapa del ciclo de desarrollo de los sistemas, el analista trabaja con los programadores para desarrollar todo el software original que sea necesario. Dentro de las técnicas estructuradas para el diseño y documentación del software se tiene: el método HIPO, los diagramas de flujo, los diagramas Warnier-Orr y el pseudocódigo. Aquí es donde, el analista de sistemas transmite al programador los requerimientos de programación durante esta fase, el analista también colabora con los usuarios para desarrollar la documentación indispensable del software, incluyendo los manuales de procedimiento.<sup>10</sup>

### **Pruebas y mantenimiento del sistema.**

El sistema de información debe probarse antes de ser utilizado. El costo es menor si se detectan los problemas antes de la entrega del sistema. El programador realiza algunas pruebas por cuenta, otras se llevan a cabo en colaboración con el analista de sistemas. En un principio se hace una serie de pruebas, con datos de tipo, para identificar las posibles fallas del sistema; más adelante, se utilizan los datos del sistema real. El mantenimiento del sistema y su documentación empiezan justamente en esta etapa y después esta función se realizará de forma rutinaria a lo largo de toda la vida del sistema.<sup>11</sup>

### **Implantación y evaluación de sistema.**

En esta última etapa del desarrollo del sistema, el analista ayuda a la implantación del sistema de información. Esto incluye el adiestramiento que el usuario requerirá si bien parte de esta capacitación la dan las casas comerciales, la supervisión del adiestramiento es una responsabilidad de analista de sistema. Más aún el analista necesita planear la suave transición que trae consigo un cambio de sistema.<sup>12</sup>

---

<sup>8</sup> <http://rastadiego.blogspot.mx/2008/08/ciclo-de-desarrollo-del-sistema.html>

<sup>9</sup> *Ibidem.*

<sup>10</sup> *Ibidem.*

<sup>11</sup> *Ibidem.*

<sup>12</sup> *Ibidem.*

## **I.II Arquitectura cliente/servidor.**

### **¿Qué es una arquitectura?**

Una arquitectura es un conjunto de componentes funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos como un todo, de manera que pueden ser utilizados eficazmente dentro de la organización.<sup>13</sup>

### **¿Qué es un cliente?**

Es el que inicia un requerimiento de servicio. El requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de redes LAN o WAN. La ubicación de los datos o de las aplicaciones es totalmente transparente para el cliente.<sup>14</sup>

### **¿Qué es un servidor?**

Es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente. Los servidores pueden estar conectados a los clientes a través de redes LANs o WANs, para proveer de múltiples servicios a los clientes y ciudadanos tales como impresión, acceso a bases de datos, fax, procesamiento de imágenes, etc.<sup>15</sup>

### **Concepto de arquitectura cliente/servidor.**

La tecnología cliente/servidor es el procesamiento cooperativo de la información por medio de un conjunto de procesadores, en el cual múltiples clientes, distribuidos geográficamente, solicitan requerimientos a uno o más servidores centrales.

Desde el punto de vista funcional, se puede definir la computación cliente/servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información de forma transparente aún en entornos multiplataforma. Se trata pues, de la arquitectura más extendida en la realización de sistemas distribuidos.<sup>16</sup>

Un sistema cliente/servidor es un sistema de información distribuido basado en las siguientes características:

- Servicio: Unidad básica de diseño. El servidor los proporciona y el cliente los utiliza.

---

<sup>13</sup> <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>

<sup>14</sup> [Ibídem](#)

<sup>14</sup> <http://fceca.unicauca.edu.co/old/admonproyectos.htm>

<sup>15</sup> [Ibídem](#)

<sup>16</sup> Tanenbaum, A.: Sistemas Distribuidos.

- Recursos compartidos: Muchos clientes utilizan los mismos servidores y a través de ellos comparten tanto recursos lógicos como físicos.
- Protocolos asimétricos: Los clientes inician “conversaciones”. Los servidores esperan su establecimiento pasivamente.
- Transparencia de localización física de los servidores y clientes: El cliente no tiene por qué saber dónde se encuentra situado el recurso que desea utilizar.
- Independencia de la plataforma HW y SW que se emplee.
- Sistemas débilmente acoplados. Interacción basada en envío de mensajes.
- Encapsulamiento de servicios. Los detalles de la implementación de un servicio son transparentes al cliente.
- Escalabilidad horizontal (añadir clientes) y vertical (ampliar potencia de los servidores).
- Integridad: Datos y programas centralizados en servidores facilitan su integridad y mantenimiento.

**El esquema de funcionamiento de un sistema cliente/servidor sería:**

- El cliente solicita una información al servidor.
- El servidor recibe la petición del cliente.
- El servidor procesa dicha solicitud.
- El servidor envía el resultado obtenido al cliente.
- El cliente recibe el resultado y lo procesa.

**Componentes de la arquitectura cliente/servidor**

El modelo cliente/servidor es un modelo basado en la idea del servicio, en el que el cliente es un proceso consumidor de servicios y el servidor es un proceso proveedor de servicios. Además esta relación está establecida en función del intercambio de mensajes, que es el único elemento de acoplamiento entre ambos.

De estas líneas se deducen los tres elementos fundamentales sobre los cuales se desarrollan e implantan los sistemas cliente/servidor: el proceso cliente es quien inicia el diálogo, el proceso servidor pasivamente espera a que lleguen peticiones de servicio y el middleware que corresponde a la interfaz que provee la conectividad entre el cliente y el servidor para poder intercambiar mensajes.<sup>17</sup>

Para entender en forma más ordenada y clara los conceptos y elementos involucrados en esta tecnología se puede aplicar una descomposición o arquitectura de niveles. Esta descomposición

---

<sup>17</sup> John Wiley: Introduction to Client / Server Systems: A Practical Guide for Systems Professionals.

principalmente consiste en separar los elementos estructurales de esta tecnología en función de aspectos más funcionales de la misma:

- Nivel de presentación: Agrupa a todos los elementos asociados al componente cliente.
- Nivel de aplicación: Agrupa a todos los elementos asociados al componente servidor.
- Nivel de comunicación: Agrupa a todos los elementos que hacen posible la comunicación entre los componentes cliente y servidor.
- Nivel de base de datos: Agrupa a todas las actividades asociadas al acceso de los datos.

## **Elementos principales**

### **Cliente**

Un cliente es todo proceso que reclama servicios de otro. Una definición un poco más elaborada podría ser la siguiente: cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor. Se le conoce con el término front-end.<sup>18</sup>

Éste normalmente maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de la red. Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
- Formatear resultados.

### **Servidor**

Un servidor es todo proceso que proporciona un servicio a otros. Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se le conoce con el término back-end. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.<sup>19</sup> Las principales funciones que lleva a cabo el proceso servidor se enumeran a continuación:

- Aceptar los requerimientos de bases de datos que hacen los clientes.
- Procesar requerimientos de bases de datos.
- Formatear datos para transmitirlos a los clientes.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

---

<sup>18</sup> *Ibíd*em

<sup>19</sup> *Ibíd*em, página 5.



Puede darse el caso que un servidor actúe a su vez como cliente de otro servidor. Existen numerosos tipos de servidores, cada uno de los cuales da lugar a un tipo de arquitectura cliente/servidor diferente.

### **Middleware**

El middleware es un módulo intermedio que actúa como conductor entre sistemas, permitiendo a cualquier usuario de sistemas de información comunicarse con varias fuentes de información que se encuentran conectadas por una red. En el caso que nos concierne, es el intermediario entre el cliente/servidor y se ejecuta en ambas partes.

La utilización del middleware permite desarrollar aplicaciones en arquitectura cliente/servidor independizando los servidores y clientes, facilitando la interrelación entre ellos y evitando dependencias de tecnologías propietarias. El concepto de middleware no es un concepto nuevo. Los primeros “monitores de teleproceso” de los grandes sistemas basados en tecnología cliente/servidor ya se basaban en él, pero es con el nacimiento de la tecnología basada en sistemas abiertos cuando el concepto de middleware toma su máxima importancia.<sup>20</sup> El middleware se estructura en tres niveles:

- Protocolo de transporte.
- Network Operating System (NOS).
- Protocolo específico del servicio.

#### **Las principales características de un middleware son:**

- Simplifica el proceso de desarrollo de aplicaciones al independizar los entornos propietarios.
- Permite la interconectividad de los sistemas de información del organismo.
- Proporciona mayor control del negocio al poder contar con información procedente de distintas plataformas sobre el mismo soporte.
- Facilita el desarrollo de sistemas complejos con diferentes tecnologías y arquitecturas.<sup>21</sup>

## **I.III Introducción a las bases de datos.**

### **Concepto de bases de datos.**

Las bases de datos (cuya abreviatura es *BD*) son una entidad en la cual se pueden almacenar datos de manera estructurada, con la menor redundancia posible (repetición de los mismos datos en diferentes lugares). Diferentes programas y diferentes usuarios deben poder utilizar estos datos. Por lo tanto, el concepto de base de datos generalmente está relacionado con el de red ya que se

---

<sup>20</sup> *Ibíd*em, página 5.

<sup>21</sup> *Ibíd*em página 5.

debe poder compartir esta información. De allí el término base. "Sistema de Información" es el término general utilizado para la estructura global que incluye todos los mecanismos para compartir datos que se han instalado.

### **Tipos de bases de datos.**

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al criterio elegido para su clasificación:

#### ***Según la viabilidad de los datos almacenados.***

##### Bases de datos estáticas

Éstas 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**.<sup>22</sup>

##### 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.<sup>23</sup>

#### ***Según el contenido.***

##### Bases de datos bibliográficas

Sólo contienen un surrogante (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. Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.<sup>24</sup>

##### Bases de datos de texto completo

---

<sup>22</sup> [http://es.wikipedia.org/wiki/Base\\_de\\_datos](http://es.wikipedia.org/wiki/Base_de_datos)

<sup>23</sup> <http://www.monografias.com/trabajos34/base-de-datos/base-de-datos.shtml>

<sup>24</sup> <http://es.kioskea.net/contents/bdd/bddintro.php3>

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.

Modelos de bases de datos.

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una “descripción” de algo conocido como *contenedor de datos* (el lugar 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<sup>25</sup>.

### **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<sup>26</sup>.

#### **Bases 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).

---

<sup>25</sup> <http://www.basesdedatos.org/>

<sup>26</sup> [Ibidem](#)

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría más por programadores que por usuarios finales<sup>27</sup>.

### **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 bases 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.

### **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.

---

<sup>27</sup> [Ibidem](#)

**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 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.<sup>28</sup>

#### **Bases de datos documentales:**

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

#### **Bases de datos deductivas:**

Un sistema de **bases 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 **bases de datos lógicas**, a raíz de que se basan en lógica matemática.<sup>29</sup>

#### **Modelo entidad-relación.**

Los diagramas o modelos entidad-relación (denominado por su siglas, ERD “Diagram Entity relationship”) son una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades.<sup>30</sup>

#### **Cardinalidad de las relaciones.**

El diseño de relaciones entre las tablas de una base de datos puede ser la siguiente:

**Relaciones de uno a uno:** una instancia de la entidad A se relaciona con una y solamente una de la entidad B.

---

<sup>28</sup> [Ibidem](#)

<sup>29</sup> <http://aceproject.org/main/espanol/et/etg03.htm>

<sup>30</sup> [Ibidem](#)

**Relaciones de uno a muchos:** cada instancia de la entidad A se relaciona con varias instancias de la entidad B.

**Relaciones de muchos a muchos:** cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B.

## Normalización

### ¿Qué es normalización?

Normalización es un proceso que clasifica relaciones, objetos, formas de relación y demás elementos en grupos, en base a las características que cada uno posee. Si se identifican ciertas reglas, se aplica una categoría; si se definen otras reglas, se aplicará otra categoría.

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 tabla 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.

### Primera forma normal

Una tabla está en primera forma normal (1FN) si:

- Todos los atributos son atómicos. Un atributo es atómico si los elementos del dominio son indivisibles, mínimos.
- La tabla contiene una llave primaria única.
- La llave primaria no contiene atributos nulos.
- No debe existir variación en el número de columnas.
- Los campos no llave deben identificarse por la llave (Dependencia funcional).
- Debe existir una independencia del orden tanto de las filas como de las columnas, es decir, si los datos cambian de orden no deben cambiar sus significados.
- Una tabla no puede tener múltiples valores en cada columna.

Esta forma normal elimina los valores repetidos dentro de una BD.

### Segunda forma normal (2FN).

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. (Todos los atributos que no son clave principal deben depender únicamente de la clave principal).

### **Tercera forma normal.**

La tabla ya está en la segunda forma normal y ningún atributo no-primario de la tabla es dependiente transitivamente de una clave primaria.<sup>31</sup>

### **Metodologías para el desarrollo de sistemas de información.**

Dentro del desarrollo de sistemas de información es necesario el seguimiento de una metodología que enfoquen los requerimientos que las organizaciones demanden. A continuación se definen algunas metodologías: modelo en cascada, estructurado, prototipo, incremental y espiral.

#### **El modelo de desarrollo en cascada.**

En él, cada etapa deja el camino preparado para la siguiente, de forma que ésta última no debe comenzar hasta que no ha acabado aquella. De esta forma, se reduce mucho la complejidad de la gestión, ya que basta con no dar por terminada una etapa hasta que haya cumplido totalmente con sus objetivos. De esta forma, la siguiente puede apoyarse con total confianza en ella. A la hora, por ejemplo, de fijar plazos, se podrían establecer planes de una forma totalmente secuencial, quedando perfectamente delimitadas las responsabilidades de los equipos que desarrollen cada etapa.

En la realidad la aplicación de este modelo no suele ser tan radical. Aunque se intenta conseguir la mayor secuencialidad posible, es difícil evitar las "vueltas atrás". Si después de la terminación de alguna etapa los resultados no son los esperados, en la práctica es muy posible que el problema esté en la mala realización de una etapa anterior. Y esto es así porque no sabemos cómo decidir con total certidumbre que una etapa ha sido perfectamente desarrollada hasta que se observan las consecuencias, quizás varias etapas y bastante tiempo después de que fue "cerrada". En estos casos, habrá que volver a ella, refinando el producto de una forma iterativa hasta que se considere que tiene la calidad deseada.

En el modelo "puro", las fases en que se suele dividir el ciclo de vida en este modelo son las siguientes:

- A. Definición (análisis de los requerimientos software).
- B. Diseño (podría dividirse en preliminar y detallado).

---

<sup>31</sup>[http://es.wikipedia.org/wiki/Normalizaci%C3%B3n\\_de\\_bases\\_de\\_datos](http://es.wikipedia.org/wiki/Normalizaci%C3%B3n_de_bases_de_datos)

- C. Codificación.
- D. Pruebas.
- E. Implantación.
- F. Mantenimiento.

### **Definición.**

La definición de requisitos o especificación de características que ha de cumplir el software que vamos a desarrollar es la primera etapa del modelo en cascada y la más importante. Al fin y al cabo, lo que sea o no sea el producto final depende de decisiones tomadas en esta etapa. Se trata fundamentalmente de estudiar las necesidades y preferencias del usuario. Es también muy importante dejar clara constancia de las decisiones tomadas en esta etapa, para ser consideradas posteriormente. Por ello, la documentación producida en esta fase debe ser concreta y estar siempre disponible durante el resto del proceso.

Pero, normalmente, nuestro software no será más que una parte de un sistema mayor. Y puede ser que "herede" problemas de indefinición de este sistema. Por ejemplo, como el hardware es muy difícil de modificar, a menudo los programas sufren cambios de última hora para "tapar" sus defectos. Esto hace que, en un proyecto real, sean bastante normales los caminos de vuelta desde etapas posteriores a ésta de definición.

Por si hubiera pocos problemas debemos tener en cuenta que el entorno en el que nos movemos suele ser muy variable, de forma que las características que piden los usuarios suelen cambiar muy rápidamente con el tiempo. Además, las innovaciones tecnológicas hacen posible modificar ciertas partes del sistema para ganar, por ejemplo en eficiencia. Por todo esto, la facilidad de modificación del software resultante es siempre un requisito fundamental, que debe compaginarse con los demás.

### **Diseño.**

Una vez planteada la especificación del programa, hay que analizar desde un punto de vista técnico las posibles soluciones. Entre ellas, se elegirá la que se considere más adecuada. A partir de ese momento, se decidirá la estructura general del programa (subdivisión en partes y relaciones entre ellas). Para cada una de las partes se seguirá recursivamente un proceso similar, hasta que tengamos totalmente definido el programa y estemos listos para pasar a la fase de codificación.

En el análisis de cada una de las partes nos encontraremos normalmente con que hay varias soluciones posibles (por ejemplo varios algoritmos para realizar la misma tarea). La elección de una de ellas suele realizarse de una forma más o menos intuitiva: no hay metodologías efectivas que nos ayuden en esta decisión.



Como puede deducirse de lo dicho hasta aquí, la descomposición en niveles de abstracción también será útil en esta fase. Cada etapa del proceso recursivo descrito puede constituir un nivel de abstracción. Si además, utilizamos las posibilidades de ocultación de información que nos permite esta metodología, podremos descomponer nuestro programa en pequeños módulos fáciles de modificar.

En el nivel más bajo del diseño hay que decidir la estructura de control y el flujo de datos del módulo. El uso de la programación estructurada facilita enormemente la comprensión de los algoritmos, al limitar los flujos de control posibles.

### **Codificación.**

En un proyecto grande ésta es la etapa más sencilla (en contra de lo que suele suponer cualquier persona que comienza a aprender un lenguaje de programación). Si el diseño es adecuado y suficientemente detallado la codificación de cada módulo es algo casi automático.

Una de las principales decisiones a tomar en esta fase es la del lenguaje a emplear, aunque a veces en el diseño ya está de alguna forma implícito. Desde hace tiempo la tendencia es a utilizar lenguajes de más alto nivel, sobre todo a medida de que se dispone de compiladores más eficientes. Esto ayuda a los programadores a pensar más a cerca de su propio nivel que del de la máquina y la productividad suele mejorarse. Como contrapartida este tipo de lenguajes son más difíciles de aprender. Y además hay que tener en cuenta que los programadores suelen ser conservadores y reacios a aprender nuevos lenguajes: prefieren usar los que ya conocen. La existencia, en una organización, de una gran cantidad de programas desarrollados en un determinado lenguaje, hace además muy dura la decisión de cambiar a uno nuevo.

Evaluar la calidad de la codificación es una tarea nada fácil. Para un mismo diseño son posibles muchas implementaciones diferentes. Y no hay criterios claros que permitan decidir cuál es la mejor. En este punto, las métricas del software pueden ser utilizadas en nuestra ayuda.

Cuando intervienen varias personas, pueden aparecer problemas a la hora de realizar modificaciones, debido a que cada uno tiene su propio estilo. Por eso se hace necesario definir estándares de estilo para facilitar la legibilidad y claridad del software producido.

### **Prueba.**

En esta fase hay que comprobar que las especificaciones se cumplen perfectamente y en todos los casos. En la realidad es prácticamente imposible probar un programa totalmente: por ello siempre suele quedar algún error escondido. Este problema se agrava cuando sobre él se realizan repetidos cambios y correcciones. Si no los gestionamos de una forma adecuada podemos acabar con un conjunto de parches que más que soluciones aportan problemas.

Actualmente se están comenzando a utilizar técnicas de verificación y validación como alternativa a la simple prueba de programas. La verificación y validación es una disciplina de ingeniería de sistemas, que intenta evaluar el software desde un punto de vista sistémico. Utiliza una aproximación estructurada para analizar y probar el software en relación con todos los aspectos del sistema en el cual se incluye y en especial con el hardware, los usuarios y las interfaces con otras piezas de software.

Idealmente, la verificación y validación se realiza paralelamente al desarrollo de software, durante todo su ciclo de vida (por lo que no entra en el modelo en cascada, estrictamente hablando) y pretende alcanzar los siguientes objetivos:

Descubrir pronto errores de alto riesgo, dando al equipo de diseño la oportunidad de elaborar una solución adecuada, evitando que se vea obligado a poner un "parche" si el error se detecta demasiado tarde.

Evaluar el ajuste de los productos desarrollados a las especificaciones del sistema.

Proporcionar al equipo de gestión información actualizada sobre la calidad y el progreso del esfuerzo de desarrollo.

### **Implantación.**

La implantación es el proceso de verificar e instalar nuevo equipo, entrenar a los usuarios, instalar la aplicación y construir todos los archivos de datos necesarios para utilizarla.

Cada estrategia de implantación tiene sus méritos de acuerdo con la situación que se considere dentro de la empresa. Sin importar cuál sea la estrategia utilizada, los encargados de desarrollar el sistema procuran que el uso inicial del sistema se encuentre libre de problemas.

Los sistemas de información deben mantenerse siempre al día, la implantación es un proceso de constante evolución.

### **Mantenimiento**

Una de las etapas más críticas, ya que se destina un 75% de los recursos, es el mantenimiento del Software ya que al utilizarlo como usuario final puede ser que no cumpla con todas nuestras expectativas.

### **Método de desarrollo por análisis estructurado.**

Muchos especialistas en sistemas de información reconocen la dificultad de comprender de manera completa sistemas grandes y complejos. El método de desarrollo del análisis estructurado tiene como finalidad superar esta dificultad por medio de:

1. La división del sistema en componentes.
2. La construcción de un modelo del sistema.

El análisis estructurado se concentra en especificar lo que se requiere que haga el sistema o la aplicación. Permite que las personas observen los elementos lógicos (lo que hará el sistema) separados de los componentes físicos (computadora, terminales, sistemas de almacenamiento, etc.). Después de esto se puede desarrollar un diseño físico eficiente para la situación donde será utilizado.

El análisis estructurado es un método para el análisis de sistemas manuales o automatizados, que conduce al desarrollo de especificaciones para sistemas nuevos o para efectuar modificaciones a los ya existentes. Este análisis permite al analista conocer un sistema o proceso en una forma lógica y manejable al mismo tiempo que proporciona la base para asegurar que no se omite ningún detalle pertinente.

### **Componentes.**

**Símbolos gráficos:** Iconos y convenciones para identificar y describir los componentes de un sistema junto con las relaciones entre estos componentes.

**Diccionario de datos:** descripción de todos los datos usados en el sistema. Puede ser manual o automatizado.

**Descripciones de procesos y procedimientos:** declaraciones formales que usan técnicas y lenguajes que permiten a los analistas describir actividades importantes que forman parte del sistema.

**Reglas:** estándares para describir y documentar el sistema en forma correcta y completa.

### **Diseño estructurado.**

El diseño estructurado es otro elemento del método de desarrollo por análisis estructurado que emplea la descripción gráfica, se enfoca en el desarrollo de especificaciones del software.

El objetivo del diseño estructurado es programas formados por módulos independientes unos de otros desde el punto de vista funcional.

La herramienta fundamental del diseño estructurado es el diagrama estructurado que es de naturaleza gráfica y evitan cualquier referencia relacionada con el hardware o detalles físicos. Su finalidad no es mostrar la lógica de los programas (que es la tarea de los diagramas de flujo).

Los diagramas estructurados describen la interacción entre módulos independientes junto con los datos que un módulo pasa a otro cuando interacciona con él.<sup>32</sup>

### **El modelo de desarrollo por prototipo.**

Es una metodología valiosa para identificar con rapidez las necesidades particulares de información del usuario. Es una técnica compleja que requiere del conocimiento cabal del ciclo de vida del desarrollo de sistemas antes de llegar a implantarlo con éxito.

El método del prototipo de sistemas consta de 5 etapas:

**1). Identificación de requerimientos conocidos:** La determinación de los requerimientos de una aplicación es tan importante para el método de desarrollo de prototipos como lo es para el ciclo de desarrollo de sistemas o análisis estructurado. Por consiguiente, antes de crear un prototipo, los analistas y usuario deben de trabajar juntos para identificar los requerimientos conocidos que tienen que satisfacer.

**2). Desarrollo de un modelo de trabajo:** Es fácil comenzar el proceso de construcción del prototipo con el desarrollo de un plan general que permita a los usuarios conocer lo que se espera de ellas y del proceso de desarrollo. Un cronograma para el inicio y el fin de la primera interacción es de gran ayuda. En el desarrollo del prototipo se preparan los siguientes componentes:

- a) El lenguaje para el dialogo o conversación entre el usuario y el sistema.
- b) Pantallas y formatos para la entrada de datos.
- c) Módulos esenciales de procesamiento.
- d) Salida del sistema

**3). Utilización del prototipo:** Es responsabilidad del usuario trabajar con el prototipo y evaluar sus características y operación. La experiencia del sistema bajo condiciones reales permite obtener la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios, así como las características inadecuadas

**4). Revisión del prototipo:** Durante la evaluación los analistas de sistemas desean capturar información sobre los que les gusta y lo que les desagrada a los usuarios.

---

<sup>32</sup>SENN, James A. (1992) Análisis y Diseño de Sistemas de Información. Segunda Edición. Editorial McGrawHill. México .

Los cambios al prototipo son planificados con los usuarios antes de llevarlos a cabo, sin embargo es el analista responsable de tales modificaciones.

**5). Repetición del proceso las veces que sea necesarias:** El proceso antes descrito se repite varias veces, el proceso finaliza cuando los usuarios y analistas están de acuerdo en que el sistema ha evolucionado lo suficiente como para incluir todas las características necesarias.

### ¿Cuáles son los tipos de prototipos?

- a. Prototipos de remiendo: Tiene que ver con la construcción de un sistema que si bien funciona, se encuentra remendado o parchado. A este concepto se le denomina como tableta de prototipo, en donde se desarrolla un circuito integrado mediante un modelo operable de elementos sobrepuestos a uno definido.
- b. Modelo de escala funcional: Son modelos no funcionales que se construyen a escala, con el objeto de evaluar ciertos aspectos del diseño.
- c. Primer modelo de escala completa: Implica crear un primer sistema o escala completa llamado con frecuencia "Piloto". Permite una interacción realista con el sistema reduciendo los costos de solución de cualquier problema que emerja con el nuevo sistema. Éste es útil cuando se planea implantar el mismo sistema de información en varias instalaciones.
- d. Un modelo que cuenta con ciertas características esenciales: La construcción de un método funcional que incluye algunas pero no todas las características que tendrá el sistema final.<sup>33</sup>

### El modelo de desarrollo incremental.

En una visión genérica, el proceso se divide en 4 partes: Análisis, Diseño, Código y Pruebas. Sin embargo, para la producción del software, se usa el principio de trabajo en cadena o "Pipeline", utilizado en muchas otras formas de programación. Con esto se mantiene al cliente en constante contacto con los resultados obtenidos en cada incremento. Es el mismo cliente el que incluye o desecha elementos al final de cada incremento a fin de que el software se adapte mejor a sus necesidades reales. El proceso se repite hasta que se elabore el producto completo.

El modelo incremental combina elementos del modelo cascada (aplicado repetidamente) así como la filosofía iterativa del prototipado.

La parte inicial es el núcleo del producto (es la parte mas importante).

---

<sup>33</sup> SENN, James A. Análisis y Diseño de Sistemas de Información. Segunda Edición. Editorial McGrawHill. México .

Una nueva versión del producto surge cuando nuevas características han sido implementadas a medida que han sido sugeridas por el usuario.

Este modelo es aplicable cuando es difícil establecer los requisitos iniciales de un proyecto y es más apropiado para proyectos pequeños. Las nuevas versiones pueden ser planeadas de modo que los requisitos técnicos puedan ser administrados.

El objetivo es trabajar junto al usuario para descubrir sus requisitos de manera incremental antes de que el producto final sea obtenido.

El modelo iterativo de desarrollo de software se basa en que antes de entregar el sistema de una vez, tanto el desarrollo como las entregas se dividen en incrementos.

Los requisitos del usuario se priorizan y los requisitos de prioridad más alta se incluyen en los incrementos más tempranos.

Cuando el desarrollo de un incremento comienza, sus requisitos son inamovibles, aunque los requisitos de incrementos posteriores pueden continuar desarrollándose.

Los clientes no tienen que esperar hasta tener el sistema completo.

El primer incremento satisface los requisitos más críticos.

Los primeros incrementos sirven como prototipo y ayudan en la tarea de detectar los posteriores requisitos.<sup>34</sup>

### **El modelo de desarrollo en espiral.**

El desarrollo en espiral es un modelo de ciclo de vida desarrollado por Barry Boehm en 1985, utilizado generalmente en la Ingeniería de software.

Las actividades de este modelo son una espiral, cada bucle es una actividad.

Para cada actividad habrá cuatro tareas:

- Planificación: Determinación de objetivos, alternativas y restricciones.
- Revisamos todo lo hecho, evaluándolo y con ello decidimos si continuamos con las fases siguientes y planificamos la próxima actividad.
- Análisis de riesgo: Análisis de alternativas e identificación/resolución de riesgos.
- Ingeniería: Desarrollo del producto del “siguiente nivel”.
- Tareas de la actividad propia y se prueba.
- Análisis de alternativas e identificación y resolución de riesgos.
- Evaluación del cliente: Valorización de los resultados de la ingeniería.<sup>35</sup>

---

<sup>34</sup> <http://xherrera334.blogspot.es/i2007-10/>

<sup>35</sup> <http://www.mitecnologico.com/Main/ModeloDeEspiral>

En la tabla 1 Se muestran algunas comparaciones de las distintas metodologías.

Metodología	Características
Cascada	<ul style="list-style-type: none"> <li>-La calidad del producto resultante es alta.</li> <li>-Permite la departamentalización y control de gestión.</li> <li>-El horario se establece con los plazos normalmente adecuados para cada etapa de desarrollo.</li> <li>-Este proceso conduce a entregar el proyecto a tiempo.</li> <li>-Es sencilla y facilita la gestión de proyectos.</li> <li>-Permite tener bajo control el proyecto.</li> <li>-Limita la cantidad de interacción entre equipos que se produce durante el desarrollo</li> </ul>
Estructurado	<ul style="list-style-type: none"> <li>-El desarrollo de diagramas y esquemas consume mucho tiempo.</li> <li>-Esta técnica conduce a la especificación de módulos de programas que son funcionalmente independientes.</li> </ul>
Prototipo	<ul style="list-style-type: none"> <li>-Las características del sistema que se consideran poco importantes quedarán fuera del prototipo al trabajar con módulos manipulables.</li> <li>-Adoptarlo como el sistema final: Los usuarios y profesionales de sistemas pueden considerar al prototipo como el sistema final cuando aún es incompleto e inadecuado.</li> </ul>
Incremental	<ul style="list-style-type: none"> <li>-No recomendable para casos de sistemas de tiempo real, de alto nivel de seguridad, de procesamiento distribuido, y/o de alto índice de riesgos.</li> <li>-Requiere de mucha planeación, tanto administrativa como técnica.</li> <li>-Requiere de metas claras para conocer el estado del proyecto.</li> <li>-Los errores en los requisitos se detectan tarde.</li> </ul>
Espiral	<ul style="list-style-type: none"> <li>-Es difícil evaluar los riesgos.</li> <li>-Necesita de la participación continua por parte del cliente.</li> <li>-Cuando se subcontrata hay que producir previamente una especificación completa de lo que se necesita y esto lleva tiempo.</li> </ul>

Tabla 1 Comparación entre las distintas metodologías de desarrollo de software.

Después de analizar las diferentes metodologías se tomó la decisión de utilizar la metodología en cascada ya que es la más conocida, sencillo en cuanto a su gestión y adecuada para llevar a cabo el proyecto de la tesis con excelente calidad.

## I.IV Marco tecnológico.

En esta sección se explicará el software utilizado para llevar a cabo el proyecto de esta tesis.

### **Servidor apache, PHP, MySQL, Adobe DreamWeaver y Wampserver.**

#### **Servidor apache.**

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.12 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años.<sup>36</sup>

**PHP** es un lenguaje de programación interpretado (Lenguaje de alto rendimiento), diseñado originalmente para la creación de web dinámicas. Se usa principalmente para la interpretación del lado del servidor (*server-side scripting*) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica.<sup>37</sup>

**MySQL** es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

---

<sup>36</sup> [http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache)

<sup>37</sup> [Ibidem](#)



Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.<sup>38</sup>

### **Aplicaciones de MySQL**

MySQL es muy utilizado en aplicaciones web como Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python) y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.<sup>39</sup>

### **Características**

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

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferente velocidad de operación, soporte físico, capacidad, distribución geográfica.
- Transacciones y claves foráneas.
- Conectividad segura.

---

<sup>38</sup> [Ibidem](#)

<sup>39</sup> [Ibidem](#)

- Replicación.
- Búsqueda e indexación de campos de texto.

MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Ésta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder y procesar datos guardados en un computador, se necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

MySQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GPL o requiere introducir código MySQL en aplicaciones comerciales, usted puede comprar una versión comercial licenciada.<sup>40</sup>

**Adobe Dreamweaver** es una aplicación en forma de suite (basada en la forma de estudio de Adobe Flash) que está destinada a la construcción, diseño y edición de sitios y aplicaciones Web basados en estándares. Creado inicialmente por Macromedia (actualmente producido por Adobe Systems) es el programa más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y recientemente, por su soporte de los estándares del World Wide Web Consortium. Su principal competidor es Microsoft Expression Web y tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras. Hasta la versión MX, fue duramente criticada por su escaso soporte de los estándares de la web, ya que el código que generaba era con frecuencia sólo válido para Internet Explorer y no validaba como HTML estándar. Esto se ha ido corrigiendo en las versiones recientes.

Se vende como parte de la suite Adobe Creative Suite.

La gran ventaja de este editor sobre otros es su gran poder de ampliación y personalización del mismo, puesto que en este programa, sus rutinas (como la de insertar un hipervínculo, una

---

<sup>40</sup> <http://dev.mysql.com/doc/refman/5.0/es/features.html>

imagen o añadir un comportamiento) están hechas en Javascript-C, lo que le ofrece una gran flexibilidad en estas materias. Esto hace que los archivos del programa no sean instrucciones de C++ sino rutinas de Javascript que hace que sea un programa muy fluido, que todo ello hace, que programadores y editores web hagan extensiones para su programa y lo ponga a su gusto.<sup>41</sup>

### **WampServer**

Wamp Server es un completísimo paquete de software que sirve para instalar y configurar fácilmente en tu sistema el popular servidor Web Apache, juntamente con el lenguaje de programación PHP y el servidor de base de datos MySQL. Todo lo necesario para instalar un servicio de base de datos web, reunido en una única aplicación.<sup>42</sup>

---

<sup>41</sup> [ibidem](#)

<sup>42</sup> <http://www.brothersoft.es/download/47226.html>

**CAPÍTULO II.  
ANÁLISIS DE LOS REQUERIMIENTOS DEL  
SOFTWARE.**

## Capítulo II. Análisis de los requerimientos del software.

En este capítulo se describirán los requerimientos necesarios para cumplir el objetivo de este proyecto, el cual es agilizar el servicio brindado tanto para alumnos como profesores en la Coordinación de Seminarios y Servicio Social DIE (CSYSS DIE), la cual cuenta con un sistema antiguo y presenta ciertas deficiencias como son la captura doble para evitar pérdidas de información, además de que es necesario llevar registros de los trámites que se realizan en dicho lugar.

Para la CSYSS DIE es indispensable llevar una administración eficiente de cierta información como es de profesores, para seleccionar el jurado en los exámenes profesionales, para los temas de tesis, evitando tesis parecidas o idénticas, que cada profesor pueda observar a través de internet sus tesis aprobadas, el cupo y vacantes que tiene y pueda inscribir a sus alumnos por este medio sin necesidad de tener que llevar documentación a la misma.

La base de datos debe almacenar la siguiente información:

Datos personales de alumnos como son; nombre, apellidos, genero, calle y número, colonia, código postal, delegación o municipio, teléfono particular, teléfono de oficina, extensión, observaciones, email, carrera y generación.

Datos del servicio social; número de cuenta, carrera, créditos, promedio, año, mes y número consecutivo de servicio social, fecha de inicio, duración en meses, clave de la dependencia, clave DGOSE, status (alta/baja), fecha de terminación y comentarios.

Datos de los temas de seminario y/o tesis; año mes y número consecutivo del tema, ID del tema el cual se forma con los anteriores, título, cupo para alumnos de las carreras de Ingeniería Eléctrica Electrónica, Computación, Telecomunicaciones e Ingenieros Mecánicos Electricistas, si pertenece o no a la Facultad de Ingeniería, si es un tema del PAT (Programa de apoyo a la titulación), si su inscripción es a seminario y/o tesis, si se le va a pagar el seminario al profesor, si ese seminario es para carga académica, su horario de asesorías, clave del director, clave del codirector, fecha de aprobación y observaciones.

Datos personales de los profesores; grado, nombre, apellidos, genero, fecha de ingreso, egreso, reingreso y RFC, para sus datos particulares; calle y número, colonia, código postal, delegación, teléfono, datos del trabajo; dependencia, calle, colonia, código postal, delegación, teléfono, extensión, tipo de adscripción (Asignatura, carrera y externo) y su email.

Datos para inscripción a seminarios y/o temas de tesis; número de cuenta del alumno, carrera, créditos, promedio, inscripción (ordinaria o extraordinaria), año mes y número consecutivo del tema, ID del tema el cual se forma con los anteriores, fecha de alta de seminario, fecha de baja de seminario, calificación del seminario, fecha de alta de tesis, fecha de baja de tesis, fecha de titulación.

Datos de dependencias del servicio social; nombre de la dependencia, siglas, calle, código postal, colonia, delegación, ciudad, teléfono, tipo de dependencia (1.- UNAM, 2.- Gobierno, 3.- Descentralizada, 4.- Paraestatal, 5.- Asociación civil y 6.- Otro) y si es verdadero o falso que pertenece a la Facultad de Ingeniería.

Datos de programas de servicio social; clave, nombre, tipo de programa (1.- Investigación, 2.- Académico administrativo, 3.- Servicios, 4.- Administración pública y 5.- Docencia), nombre del responsable, cargo y teléfono.

Para trabajar con esta base de datos será necesario realizar lo siguiente: altas, bajas, cambios y consultas, altas permitirá agregar tanto a un alumno, profesor, tema, inscripción y servicio social, las bajas permitirán dar de baja cualquiera de los anteriores, así como modificaciones permitirá actualizar datos en nuestras tablas, mientras las consultas sólo permitirán observar la información que se encuentra en la base de datos, dependiendo de la consulta que se desee realizar, también se podrán imprimir constancias de inscripción a tema de tesis o seminario, datos de alumnos, datos de temas y constancias de temas de tesis de profesores.

Se hará uso de la especificación de requerimientos la cual servirá para saber qué es lo que deberá hacer el sistema en sí y esto se realizará conociendo las funciones principales con las que deberá contar el sistema.

## **II.I Especificación de requerimientos.**

El sistema deberá ser eficiente y óptimo, además de contar con alguna plataforma web para que cualquier usuario final pueda tener acceso a él por medio de internet para disponer de los servicios que éste proporciona.

Las funciones principales que realizará el sistema son:

- 1) Autenticación.
- 2) Altas.
- 3) Bajas.
- 4) Consultas.
- 5) Modificaciones.
- 6) Impresión de constancias.
- 7) Inscripciones de alumnos por medio de internet.

Los requerimientos que debe cumplir cada una de las tablas son:

Alumnos.

- Altas
- Modificaciones
- Buscar alumno por número de cuenta

- Buscar alumno por apellido y nombre(s)
- Imprimir datos personales del alumno
- Ver alumnos registrados

#### Servicio social.

- Altas
- Bajas o modificaciones
- Reportes
- Buscar alumno en servicio social por número de cuenta
- Ver alumnos registrados

#### Dependencias de servicio social.

- Altas
- Bajas
- Modificar dependencias
- Buscar dependencia
- Ver dependencias registradas

#### Programas de servicio social.

- Altas
- Bajas
- Modificar programa
- Buscar programa
- Ver programas registrados

#### Profesores.

- Altas
- Bajas
- Modificar información de profesores
- Buscar profesor por apellidos y nombre
- Buscar profesor por clave
- Imprimir temas de tesis de profesores
- Ver profesores registrados

#### Temas de seminario y/o tesis.

- Altas
- Bajas
- Modificaciones
- Imprimir temas de tesis
- Buscar temas con identificador

- Buscar tema de tesis por título
- Buscar temas por nombre del profesor
- Buscar temas con director y codirector
- Ver temas registrados

Inscripción a seminarios tesis.

- Altas
- Bajas y modificaciones
- Buscar alumno inscrito con número de cuenta
- Buscar alumno inscrito con identificador del tema
- Buscar alumno inscrito con clave del profesor
- Imprimir constancia de alumno inscrito en temas de tesis
- Ver alumnos registrados

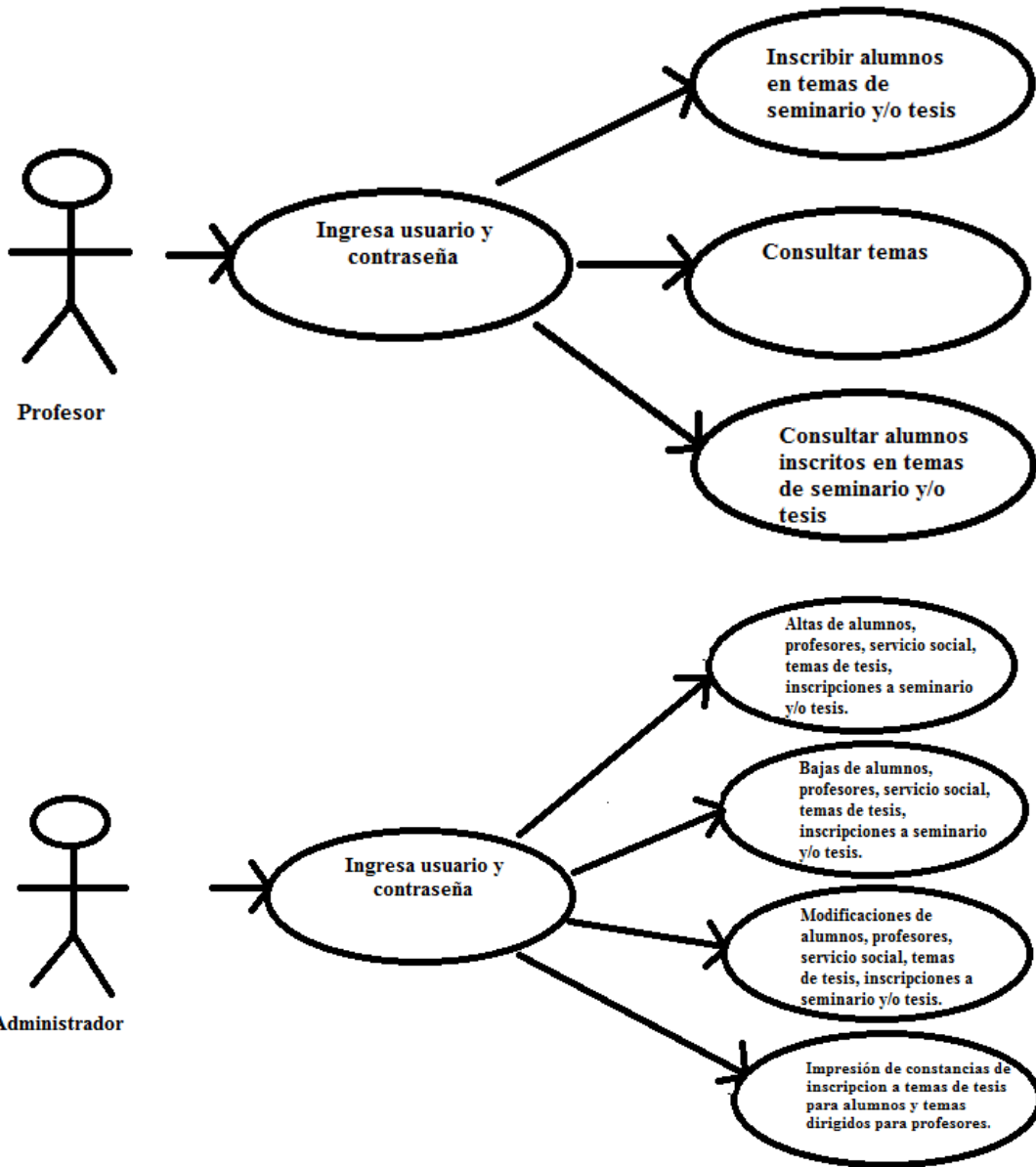
El sistema necesita que se cumplan ciertos requisitos para mejorar la eficiencia y también es necesario que se impriman algunas constancias como son:

- a) Captura de información eficiente y sin pérdidas o fuga de la misma.
- b) Consulta de información óptima y eficiente.
- c) Agilizar inscripciones a seminario y/o tesis.
- d) Agilizar entrega de constancias de inscripción a seminario/tesis de alumnos.
- e) Agilizar entrega de constancias de temas de profesores con límite de fechas.
- f) Consultar los temas de tesis en existencia y con vacantes.

A continuación se verán los casos de uso.



## II.II Casos de uso.



**CAPÍTULO III.  
DISEÑO DE LA BASE DE DATOS DEL  
SISTEMA.**

### Capítulo III. Diseño de la base de datos del sistema.

En esta etapa se especifican los detalles que establecen la forma en que el sistema debe de cumplir con los requerimientos solicitados anteriormente, esta etapa se considera diseño lógico.

Aquí se muestran las tablas y sus atributos además de las características y descripción de ambos.

#### III.I Diccionario de datos.

##### 01. Entidad: tbalumnos (alumnos)

**Objetivo:** Almacenar datos personales de los alumnos de las carreras de ingeniería en computación, eléctrica, telecomunicaciones, mecánica eléctrica.

##### Atributos.

No.	Nombre	Tipo	Long	Descripción
*01	nocuentaz	int	11	Número de cuenta del alumno.
02	apealumno	varchar	100	Apellidos del alumno.
03	nombres	varchar	100	Nombres del alumno.
04	genero	varchar	1	Género del alumno
05	callenumero	varchar	200	Calle y número del hogar del alumno.
06	colonia	varchar	200	Colonia del alumno.
07	cp	int	5	Código postal alumno.
08	delegacion	varchar	100	Delegación del alumno.
09	telparticulat	int	15	Teléfono particular del alumno
10	teloficina	int	15	Teléfono oficina del alumno
11	extension	int	5	Extensión telefónica del alumno.
12	observa	varchar	500	Observaciones del alumno.
13	email	varchar	100	Email del alumno.

**Nota:** - El (\*) indica campo llave.

-

**02. Entidad: tbcarreracursada** (carrera que están cursando los alumnos)

**Objetivo:** Almacenar información referente a la carrera que cursan los alumnos, tales como: generación, clave de la carrera, y su número de cuenta de cada alumno.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	nocuentaz	int	11	Número de cuenta del alumno.
02	clavecarrera	int	3	Clave de la carrera del alumno.
03	generacion	int	4	Generación del alumno.

**Nota:** - El (\*) indica campo llave.

-

**03. Entidad: tbdependencias** (dependencias de servicio social)

**Objetivo:** Almacenar información referente a las distintas dependencias en donde los alumnos realizan su servicio social.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	clavedepend	int	4	Clave de la dependencia.
02	nombredepen	varchar	50	Nombre de la dependencia.
03	siglasdepend	varchar	10	Siglas de la dependencia
04	calledepend	varchar	50	Calle y número de la dependencia.
05	cpdepend	int	5	Código postal de la dependencia.
06	coloniadepend	varchar	30	Colonia de la dependencia.
07	delegdepend	varchar	25	Delegación de la dependencia
08	ciudadepend	varchar	20	Ciudad que pertenece la dependencia.
09	teldepend	int	15	Teléfono de la dependencia
10	tipodepend	int	1	Tipo de dependencia
11	fidepend	varchar	10	Pertenece a la facultad de ingeniería UNAM.

**Nota:** - El (\*) indica campo llave.

-

#### 04. Entidad: tbprofesores (profesores)

**Objetivo:** Almacenar datos personales de los profesores pertenecientes a la división de ingeniería eléctrica.

#### Atributos.

No.	Nombre	Tipo	Long	Descripción
*01	claveprof	int	6	Se generará automáticamente y de manera incremental.
02	gradoprof	varchar	8	Grado de estudios del profesor.
03	nombrefprof	varchar	20	Nombre(s) del profesor.
04	apellidosprof	varchar	50	Apellidos del profesor.
05	generoprof	varchar	1	Género del profesor.
06	ingresoprof	int	8	Fecha de ingreso empezando por el día, mes y año (ddmmaaaa).
07	rfc	varchar	20	RFC del profesor.
08	calleprof	varchar	50	Calle y número del hogar del profesor.
09	coloniaprof	varchar	20	Colonia del hogar del profesor.
10	cpprof	int	5	Código postal del hogar del profesor.
11	delegprof	varchar	30	Delegación del hogar del profesor.
12	telprof	int	15	Teléfono de casa del profesor.
13	deplabprof	varchar	50	Dependencia donde labora el profesor.
14	calleofiprof	varchar	30	Calle de la dependencia en donde labora el profesor.
15	coloniaofiprof	varchar	30	Colonia de la dependencia en donde labora el profesor.
16	cpofiprof	int	5	Código postal de la dependencia en donde labora el profesor.
17	delegofiprof	varchar	30	Delegación de la dependencia en donde labora el profesor.
18	telofiprof	int	15	Teléfono de la oficina del profesor.
19	extofiprof	int	5	Extensión del teléfono de oficina, si existe.
20	tipoadscrip	varchar	1	Tipo de adscripción.
21	emailprof	varchar	50	Email del profesor.

**Nota:** - El (\*) indica campo llave.

-

**05. Entidad: tbprogramass** (programas de servicio social)

**Objetivo:** Almacenar los programas de servicio social existentes, para los alumnos de la división de ingeniería eléctrica.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	claveprog	bigint	20	Clave del programa de servicio social.
02	nombreprog	varchar	300	Nombre del programa de servicio social.
03	tipoprogram	int	1	Tipo de programa de servicio social.
04	responsprog	varchar	100	Responsable del programa de servicio social.
05	cargoresponsprog	varchar	100	Cargo del responsable del programa de servicio social.
06	telresponsprog	int	15	Teléfono responsable del programa de servicio social.

**Nota:** - El (\*) indica campo llave.

-

**06. Entidad: tbreporteseso** (reportes de servicio social)

**Objetivo:** Almacenar los reportes de servicio social de los alumnos de la división de ingeniería eléctrica.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	idreporte	int	12	ID del reporte.
02	año	int	4	Año del reporte.
03	mes	int	2	Mes del reporte.
04	numero	int	6	Número del reporte.
05	numeroreporte	int	2	Número del reporte, ajeno a la entidad anterior (numero).
06	reporte	int	8	Fecha de reporte empezando por el día, mes y año (ddmmaaaa).

**Nota:** - El (\*) indica campo llave.

-

**07. Entidad: tbsemitesis (seminario de tesis)**

**Objetivo:** Almacenar las inscripciones de los alumnos que procedieron a inscribirse al seminario de titulación, de la división de ingeniería eléctrica.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	nocuentaz	int	11	Número de cuenta del alumno.
02	clavcarrera	int	3	Clave de carrera del alumno.
03	creditos	float		Créditos académicos del alumno.
04	promedio	float		Promedio académico del alumno.
05	inscripcion	varchar	1	Tipo de inscripción ordinaria o extraordinaria.
06	idtema	int	10	Id del tema.
07	año	int	4	Año del tema.
08	semestre	int	2	Semestre del tema.
09	numero	int	3	Número de tema.
10	fealtasemi	int	8	Fecha de alta seminario empezando por el día, mes y año (ddmmaaaa).
11	febasemina	int	8	Fecha de baja seminario empezando por el día, mes y año (ddmmaaaa).
12	fecasemina	int	8	Fecha de calificación de seminario empezando por el día, mes y año (ddmmaaaa).
13	califsemina	varchar	2	Calificación.
14	fealtesis	int	8	Fecha de alta tesis empezando por el día, mes y año (ddmmaaaa).
15	febatesis	int	8	Fecha de baja tesis empezando por el día, mes y año (ddmmaaaa).
16	fetitutesis	int	8	Fecha de titulación empezando por el día, mes y año (ddmmaaaa).
17	observacins	int	100	Observaciones.

**Nota:** - El (\*) indica campo llave.

-

**08. Entidad: tbserviciosocial** (servicio social)

**Objetivo:** Almacenar las inscripciones de los alumnos que procedieron a inscribirse al seminario de titulación, de la división de ingeniería eléctrica.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	idseso	int	12	ID del servicio social.
02	añoss	int	4	Año del servicio social.
03	messs	int	2	Mes del servicio social.
04	numeros	int	6	Número del servicio social.
05	nocuentaz	int	11	Número de cuenta del alumno.
06	clavecarrera	int	3	Clave de la carrera.
07	promedio	float		Promedio académico del alumno.
08	creditos	float		Créditos académicos del alumno.
09	inicios	int	8	Fecha de inicio de servicio social empezando por día, mes y año (ddmmaaaa).
10	duracionss	int	2	Duración en meses del servicio social.
11	terminacionss	int	8	Fecha de terminación del servicio social empezando por día, mes y año (ddmmaaaa).
12	claveprog	bigint	50	Clave DGOSE.
13	statusss	varchar	1	Status del servicio social, el alumno puede estar dado de alta o baja.
14	comentarioss	varchar	500	Comentarios.
15	clavedepend	int	6	Clave de la dependencia.

**Nota:** - El (\*) indica campo llave.

-



**09. Entidad: tbtemas** (temas de tesis)

**Objetivo:** Almacenar los temas de tesis que se impartirán en la facultad de ingeniería.

**Atributos.**

No.	Nombre	Tipo	Long	Descripción
*01	aañotema	int	4	Año del tema.
02	semestretema	int	2	Semestre del tema.
03	numerotema	int	3	Número de tema.
04	Id	int	9	ID del tema.
05	titulotema	varchar	255	Título del tema.
06	ime	int	1	Vacantes en ingeniería mecánica eléctrica.
07	comput	int	1	Vacantes en ingeniería en computación.
08	telec	int	1	Vacantes en ingeniería en telecomunicaciones.
09	electric	int	1	Vacantes en ingeniería eléctrica.
10	fi	varchar	10	Fi
11	pat	varchar	10	Pat
12	fechaaprobtema	int	8	Fecha de aprobación de tema empezando por el día, mes y año (ddmmaaaa).
13	luninicio	time		Horario de impartición del tema de tesis.
14	lunfinal	time		Horario de impartición del tema de tesis.
15	marinicio	time		Horario de impartición del tema de tesis.
16	marfinal	time		Horario de impartición del tema de tesis.
17	mierinicio	time		Horario de impartición del tema de tesis.
18	mierfinal	time		Horario de impartición del tema de tesis.
19	jueinicio	time		Horario de impartición del tema de tesis.
20	juefinal	time		Horario de impartición del tema de tesis.
21	vieinicio	time		Horario de impartición del tema de tesis.
22	viefinal	time		Horario de impartición del tema de tesis.
23	sabinicio	time		Horario de impartición del tema de tesis.
24	sabfinal	time		Horario de impartición del tema de tesis.
25	director	time		Horario de impartición del tema de tesis.
26	codirector	time		Horario de impartición del tema de tesis.
27	observacionestema	varchar	500	Observaciones del tema de tesis.
28	pagado	varchar	10	Pagado se escogerá si o no.
29	cargaa	varchar	10	Carga se escogerá si o no.
30	seminario	varchar	10	Seminario se escogerá si o no.
31	tesis	varchar	10	Tesis se escogerá si o no.

**Nota:** - El (\*) indica campo llave.

-

### **III.II Definición de módulos.**

#### **Módulo administrador.**

Este módulo será utilizado exclusivamente por el administrador del sistema y es la única persona autorizada para realizar consultas como: altas, bajas, modificaciones o búsquedas en las distintas tablas que existen en la base de datos. También podrá ser capaz de imprimir cualquier tipo de información que requiera, será quien autorice o quite accesos al sistema, podrá realizar respaldos al sistema, exportaciones o importaciones de datos, etc., es decir será quien tenga acceso total al sistema.

#### **Módulo profesores.**

Este módulo será utilizado por los profesores de la División de Ingeniería Eléctrica, podrán ser capaces de inscribir a los alumnos a seminarios de tesis y/o tesis siempre y cuando hayan vacantes, podrán consultar información de sus temas de tesis sin poder modificar la información que ahí se muestra como es el nombre del tema, cupo, los horarios de impartición, información del director o codirector, observaciones, etc.

#### **Modelo entidad relación.**

En el Anexo 1 se muestra el modelo entidad relación de la base de datos.

**CAPÍTULO IV.  
DISEÑO DE PANTALLAS DEL SISTEMA.**

## Capítulo IV. Diseño de pantallas del sistema.

Ya finalizadas las etapas de análisis y diseño del sistema se procederá a la parte del diseño del programa, en la cual se implementará el sistema a través de un administrador de base de datos, la elección del lenguaje de programación y las herramientas para el diseño de la interfaz.

### Acceso al sistema.

Se mostrará un formulario al inicio de la aplicación, la cual le solicitará un nombre de usuario y contraseña, para poder tener acceso al sistema, como se muestra en la figura 1.



The image shows a user authentication interface. At the top, there is a dark blue header with the UNAM logo on the left, the text 'Universidad Nacional Autónoma de México Facultad de Ingeniería' in the center, and 'COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL DIE' on the right. Below the header is a photograph of a modern building at night. The main content area has a light blue background and contains the following elements: a blue instruction line that reads 'Instrucciones: Ingrese el nombre de usuario y la contraseña respetando mayúsculas y minúsculas.', a label 'Ingresa el nombre de usuario:' followed by a text input field, a label 'Ingresa la contraseña:' followed by a password input field, and a grey 'Acceder' button at the bottom.

Figura 1. Autenticación del usuario.

Y en caso de ingresar datos incorrectos mostrará la siguiente pantalla indicando el error y le pedirá al usuario que lo vuelva a intentar, de esta manera sólo personal autorizado será capaz de utilizar este sistema informático, como se muestra en la figura 2.

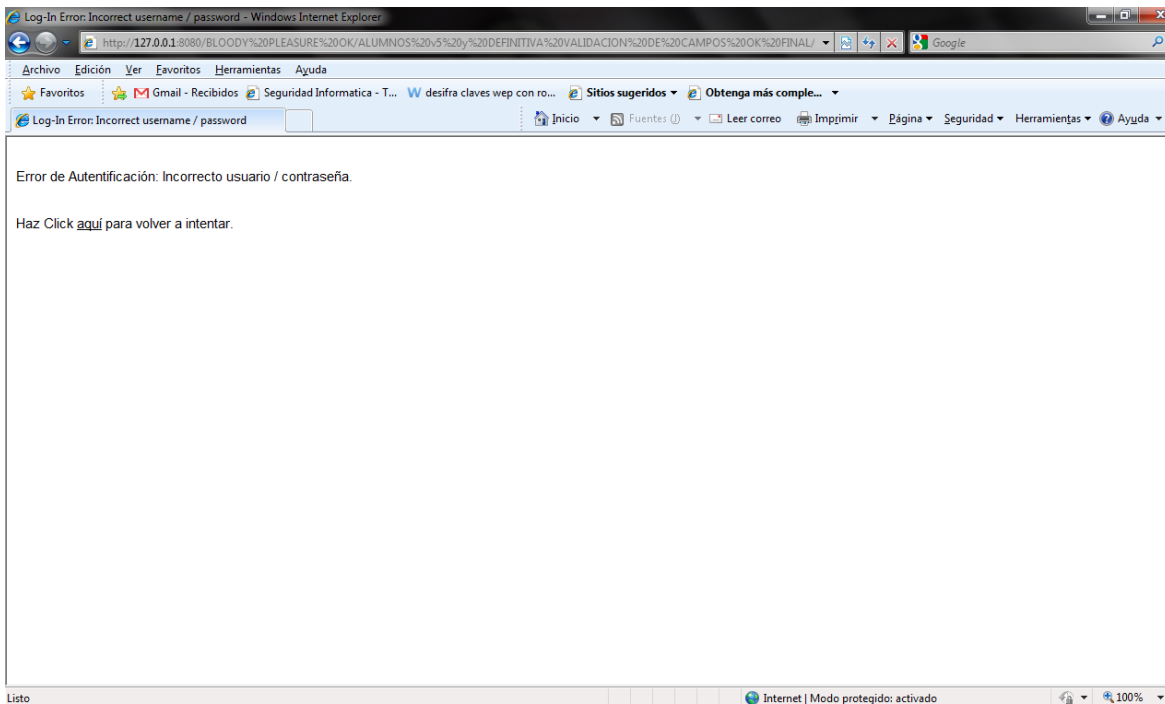


Figura 2. Datos de autenticación incorrectos.

Sin embargo al escribir correctamente el nombre de usuario y contraseña, permitirá visualizar un formulario conformado por un menú de opciones para acceder a cualquier parte que esté disponible en el sistema, contendrá varios submenús conformados como hijos.

Y dependiendo de qué botón el usuario haya escogido, se mostrará alguna de las siguientes pantallas:

Gestión de alumnos de la DIE. Véase figura 3.

Gestión de carreras para alumnos de licenciatura. Véase figura 4.

Gestión de dependencias de servicio social para alumnos de licenciatura. Véase figura 5.

Gestión de profesores para alumnos de licenciatura. Véase figura 6.

Gestión de programas de servicio social para alumnos de licenciatura. Véase figura 7.

Gestión de reportes entregados de servicio social para alumnos de licenciatura. Véase figura 8.

Gestión de seminario tesis para alumnos de licenciatura. Véase figura 9.

Gestión de servicio social para alumnos de licenciatura. Véase figura 10.

Gestión de temas para alumnos de licenciatura. Véase figura 11.

**Funciones de la pantalla correspondiente a la figura 3.**

Existen los siguientes botones:

- **Altas:** Da de alta alumnos de la DIE.
- **Bajas:** Elimina completamente al alumno seleccionado a través del número de cuenta.
- **Modificaciones:** Modifica información de alumnos por medio del número de cuenta.
- **Buscar alumno por número de cuenta:** Búsqueda de alumno por su número de cuenta.
- **Buscar alumno por apellido y nombre(s):** Búsqueda de alumno por nombre completo.
- **Imprimir datos personales del alumno:** Imprime datos del alumno.
- **Imprimir constancia de tesis del alumno:** Impresión de la constancia de tesis.
- **Ver alumnos registrados:** Muestra absolutamente todos los alumnos registrados.



GESTIÓN DE ALUMNOS DE LA DIE.

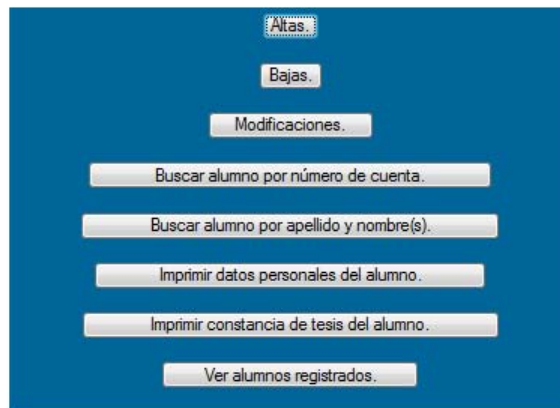


Figura 3. Gestión de alumnos de la DIE.

**Función del botón *Altas*.**

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta, como se muestra en la figura 3.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios a excepción de los *teléfonos, extensión, observaciones y email.*
- Clave de carreras:

Clave Carrera

109 Ingeniería Eléctrica-Electrónica.

110 Ingeniería en Computación.

111 Ingeniería en Telecomunicaciones.

116 Ingeniería Mecánica Eléctrica (Área Electrica-Electrónica).

- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a : [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

---

**BASE DE DATOS ALUMNOS DIE.**

**INFORMACIÓN DEL ALUMNO.**

Número de cuenta:	<input type="text"/>
Apellidos paterno y materno:	<input type="text"/>
Nombre(s):	<input type="text"/>
Género:	<input type="text" value="M"/>

Calle y número:	<input type="text"/>
Colonia:	<input type="text"/>
Código postal:	<input type="text"/>
Delegación o municipio:	<input type="text"/>
Teléfono particular:	<input type="text"/>
Teléfono oficina:	<input type="text"/>
Extensión:	<input type="text"/>
Observaciones:	<input type="text"/>
Email:	<input type="text"/>

**INFORMACIÓN ACADÉMICA.**

Carrera:	<input type="text" value="109"/>
Generación, año de ingreso (aaaa):	<input type="text"/>

Figura 3.1. Formulario de alta para alumnos.

### **Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso del número de cuenta del alumno para poder eliminar su registro, como se muestra en la figura 3.2

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BAJA DE ALUMNO.

Ingrese el número de cuenta del alumno@:

Baja de alumno.

Figura 3.2. Formulario para baja de alumno.

### **Función del botón *Modificaciones*.**

Al dar clic se desplegará un formulario indicando que deberá ser ingresado el número de cuenta del alumno a modificar, en este caso para ejemplificar se utilizó un número de cuenta real, contenido en la base de datos, el cual es: 302042836, como se muestra en la figura 3.3

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

MODIFICACIÓN DE DATOS DEL ALUMNO.

Ingrese el número de cuenta del alumno@:

Modificar datos del alumno.

Figura 3.3. Formulario para modificación de datos.



Y al buscar al alumno para hacer las modificaciones, obtenemos como resultado una pantalla en la cual se podrán modificar los campos que deseemos a excepción del número de cuenta.

Los campos del alumno que el usuario podrá modificar serán algunos como: nombre, apellidos, género, dirección, colonia, código postal, delegación teléfono particular, teléfono de oficina, extensión telefónica, observaciones, etc. Cabe destacar que al modificar los datos de esta pantalla se actualizarán simultáneamente dos tablas contenidas en la base de datos, las tablas actualizadas son las que llevan por nombre tbalumnos y tbcarreracursada, como se muestra en la figura 3.4

The image shows a web form for modifying student data. The form is set against a light pink background and contains the following fields:

- Número de cuenta: 302042836
- Apellidos del alumno: GARCIA MARTINEZ
- Nombre(s): LETICIA
- Género: F
- Calle y número: TOLNEPANTLA 15
- Colonia: PEDREGAL DE SANTO DOMING
- Código postal: 4369
- Delegación: COYOACÁN
- Teléfono particular: 2147483647
- Teléfono de oficina: 0
- Extensión: 0
- Observaciones: TERMINACIÓN: 21/11/2007
- Email: (empty field)
- Clave de carrera: 111
- Generación: 2005

At the bottom center of the form is a button labeled "Actualizar".

Figura 3.4. Formulario de modificación de datos.

**Función del botón *Buscar alumno por número de cuenta.***

Al hacer clic, se muestra un formulario requiriendo el número de cuenta del alumno a buscar, como se muestra en la figura 3.5

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE ALUMNO.

Ingrese el número de cuenta del alumno@: 302042836

Búsqueda de alumno.

Figura 3.5. Formulario de búsqueda de alumno.

Y al ser ingresado el número de cuenta, y posteriormente al hacer clic al botón: Búsqueda de alumno, mostrará la información referente a ese alumno, como se muestra en la figura 3.6

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de alumnos.**

Número de cuenta del alumno	Apellidos paterno y materno:	Nombre(s)	Género	Calle y número	Colonia	Cp
302042836	GARCIA MARTINEZ	LETICIA	F	TOLNEPANTLA 15	PEDREGAL DE SANTO DOMINGO	436

Número de alumnos por método de búsqueda: 1

Figura 3.6. Datos de alumno.

### **Función del botón *Buscar alumno por apellido y nombre(s)*.**

El usuario al hacer clic se mostrará un formulario requiriendo los datos: nombre(s) y apellidos del alumno, para poder iniciar la búsqueda del alumno.

Este método de búsqueda se realizó debido a la problemática que se tenía, al consultar ciertos datos ya que algunas veces el usuario no recordaba el número de cuenta de cierto alumno, pero si recordaba su nombre completo y por ello se decidió implementar un segundo método de búsqueda, como se muestra en la figura 3.7

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE ALUMNOS POR APELLIDOS Y NOMBRE.

Apellidos del alumno: Garcia Martinez

Nombre(s) del alumno: Leticia

Búsqueda de alumno.

Figura 3.7. Formulario de búsqueda de alumno por nombre y apellidos.

Y al hacer clic al botón *búsqueda de alumno*, de la figura 3.6, se obtiene como resultado una pantalla idéntica como la que se mostró en la figura 3.5.

### **Función del botón *imprimir datos personales del alumno*.**

Al hacer clic, se muestra un formulario requiriendo, su número de cuenta, para proceder a imprimir sus datos personales tales como: número de cuenta, apellidos, nombres, dirección, teléfono particular, generación, carrera, etc., como se muestra en las figuras 3.8 y 3.9



**BÚSQUEDA DE ALUMNOS PARA IMPRESIÓN DE DATOS PERSONALES.**

Ingrese el número de cuenta del alumno@:

Figura 3.8. Formulario que solicita impresión de datos del alumno.

Y al hacer clic al botón *búsqueda de alumno* el resultado es la figura 3.9

**INFORMACIÓN DEL ALUMNO.**

**Información personal.**

Número de cuenta:

Apellidos:

Nombre:

Calle y número:

Cólonia:

Código postal:

Delegación:

Teléfono particular:

Teléfono de oficina:

Extensión:

**Carrera(s) cursadas(s)**

Carrera:

Generación:

Figura 3.9. Impresión de datos del alumno.

**Función del botón *Imprimir constancia de tesis del alumno.***

Al dar clic se desplegará un formulario, requiriendo el número de cuenta del alumno, al cual le será impresa su constancia de tesis, cabe señalar que el alumno debe estar inscrito a algún seminario de tesis para poder imprimir este documento, como se muestra en la figura 3.10

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

IMPRESIÓN DE CONSTANCIA DE TESIS .

Ingrese el número de cuenta del alumno@: 30079508

Imprimir constancia del alumno.

Figura 3.10. Formulario solicita impresión de constancia de tesis del alumno.

Y al dar clic en el botón *imprimir constancia del alumno*, se obtiene un formato que contiene datos como número de cuenta, nombre del alumno, director de tesis, fecha, etc., como se muestra en la figura 3.11

**FACULTAD DE INGENIERÍA**  
**DIVISIÓN DE INGENIERÍA ELÉCTRICA**  
**COORDINACIÓN DE SEMINARIOS  
Y SERVICIO SOCIAL**  
**OFICIO FING/DIE/CSSS/1327/2011**  
**ASUNTO: Constancia de inscripción a Tesis**

**A QUIEN CORRESPONDA:**

Por medio del presente, me permito hacer constar que el (la) alumno (a)  
MARISOL [ ] SALGADO RANGEL [ ] nta [ ] 30079508 [ ]  
está inscrito (a) en el tema de tesis: en REDES DE PLC EN MEXICO [ ]  
10194 [ ] y se encuentra bajo la dirección del (la) profesor(a): [ ] M.I [ ]  
DAMIAN FEDERICO [ ] VARGAS SANDOVAL [ ]

Se extiende la presente a petición del(a) interesado(a) para los fines que al(a)  
misma convengan.  
Para cualquier aclaración o duda al respecto, favor de comunicarse a la  
Coordinación de Seminarios y Servicio Social, número telefónico 5622-3111.

**A tentamente.**  
"POR MI RAZA HABLARÁ EL ESPÍRITU"  
Cd.Universitaria,D.F.,a . 27/03/2012 [ ]  
LA COORDINADORA DE SEMINARIOS Y SERVICIO SOCIAL

**LIC.ANGÉLICA GUTIÉRREZ VÁZQUEZ**

Figura 3.11. Impresión de constancia de tema de tesis del alumno.

### **Función del botón *Ver alumnos registrados.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los alumnos registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 3.12

#### **Universidad Nacional Autónoma de México.**

#### **Facultad de Ingeniería.**

#### **Información de alumnos.**

Número de cuenta del alumno	Apellidos paterno y materno:	Nombre(s)	Género	Calle y número	Colonia	Cp	Delegación o
97134747	ABAD MARTINEZ	DAFNE CITLALLI	F	OCOTE L6 MZ.17	EL MOLINO	9830	IZTAPALAP.
90001031	ABAUNZA CASTILLO	JUSTO ALEJANDRO	M	CUCARACHE NO. 130	BENITO JUAREZ	57000	EDO DE ME
89358225	ABDALÁ ROMERO	MARINO OMAR	M	CALLE "H" # 9, MANZ. 12	EDUCACIÓN	4400	COYOACÁN
402012971	ABRAHAM ISRAEL	MONROY CANO	M	MARGARITAS 143	JARDINES DE MORELOS	55070	ECATEPEC
86292494	ABREGO SANTOS	LINA	M	CANO # 182	SAN MIGUEL CHAPULTEPEC	11850	MIGUEL HII
95172653	ABRIZ CABRERA	ALEJANDRO	M	AV. IGNACI COMONFORT MZ. 779 LT 03	CONCEPCIÓN	56615	VALLE DE C SOLIDARID.
302206991	ABUNDIO CRUZ	PABLO	M	PRIVADA LA COMUNIDAD # 19	TECAMACHALCO	56500	LA PAZ
94287882	ABURTO ESTEBANEZ	MARIAN	F	PONCIANO QUIROZ N° 51	LOS CIPRECES	4830	COYOACÁN
302225426	ABURTO LÓPEZ	LORENA	F	AHUEJOTE 250	PEDRAGAL DE SANTO DOMINGO	4369	COYOACÁN
99598785	ACA LINARES	ANA LAURA	F	ARMERIA 30-36 INT 107	MORELOS	15270	VENUSTIAN
302202278	ACALCO RÍOS	NANCY	F	CALLE PALMAS MZ. 2 LT. 56	CAMPESTRE POTRERO	9637	IZTAPALAP.
300029297	ACEVEDO AGUILAR	ULISES JACOBO	M	FLAMINGOS 161	METROPOLITANA 2DA SECCIÓN	57740	NEZAHUAL
95074791	ACEVEDO CAÑEDO	JORGE ARTURO	M	5 DE MAYO MZ.40 LT.8	REVOLUCIÓN	5637	CUAJIMALP
405067767	ACEVEDO DÍAZ	ELOISA	F	CDA. FRESCO MZ. 3 LT. 9	TEMASTITITLA	10640	MAGDALEN
304186547	ACEVEDO MEDINA	EMMA	F	AVENIDA JUAREZ CIÓN.ACUAUHTZINGO #14	LA CONCEPCIÓN TLACOAPA	16000	XOCHIMILC
98511796	ACEVEDO OSORIO	ALEJANDRO	M	PLAZA DE TENEXPA #44	DR. ORTIZ TIRADO	9020	IZTAPALAP.
408090577	ACEVEDO RAMÍREZ	MAURICIO	M	TEJAMANIL NO. 414	PEDREGAL DE STO. DOMINGO	4369	COYOACÁN
97193054	ACEVEDO REYES	ANDRES	M	LOS REYES NO. 22 ALTOS 1	BARRIO LOS REYES IZTACALCO	8620	IZTACALCC
91001687	ACEVEDO SANTAROSA	OLIVER	M	CONGRESO # 412	FEDERAL	15700	VENUSTIAN
302073760	ACEVEDO SOLANO	LINA	F	PROLONGACION CUAUHEMOC # 63	BARRIO SAN JOSE	9000	IZTAPALAP.
303194907	ACEVEDO VALLE	HERMINIO	M	CALLE IXTACALCO NO.127	METROPOLITANA 3RA SECCIÓN	57750	NEZAHUAL

Figura 3.12. Alumnos registrados en el sistema.

### **Funciones de la pantalla correspondiente a la figura 4.**

Nota: Aunque el módulo anterior de nombre: *Gestión de alumnos de la DIE*, se puedan llevar a cabo las tareas que se desarrollan en este módulo, aun así se decidió desarrollar debido a que, existía una fuerte necesidad por parte de la Coordinación de Seminarios y Servicio Social de la DIE, ya que muchas veces necesitaba consultar única y exclusivamente la carrera y generación de ciertos alumnos y por ende la forma más cómoda para poderlo hacer es a través de este módulo, el cual se explicará a detalle en la figura 4.

Existen los siguientes botones:

- **Altas:** Da de alta número de cuenta, clave carrera y generación de alumnos de la DIE.
- **Bajas:** Elimina completamente al alumno seleccionado a través del número de cuenta.
- **Buscar:** Modifica información de alumnos por medio del número de cuenta.

- **Ver listado completo:** Muestra absolutamente todos registros.

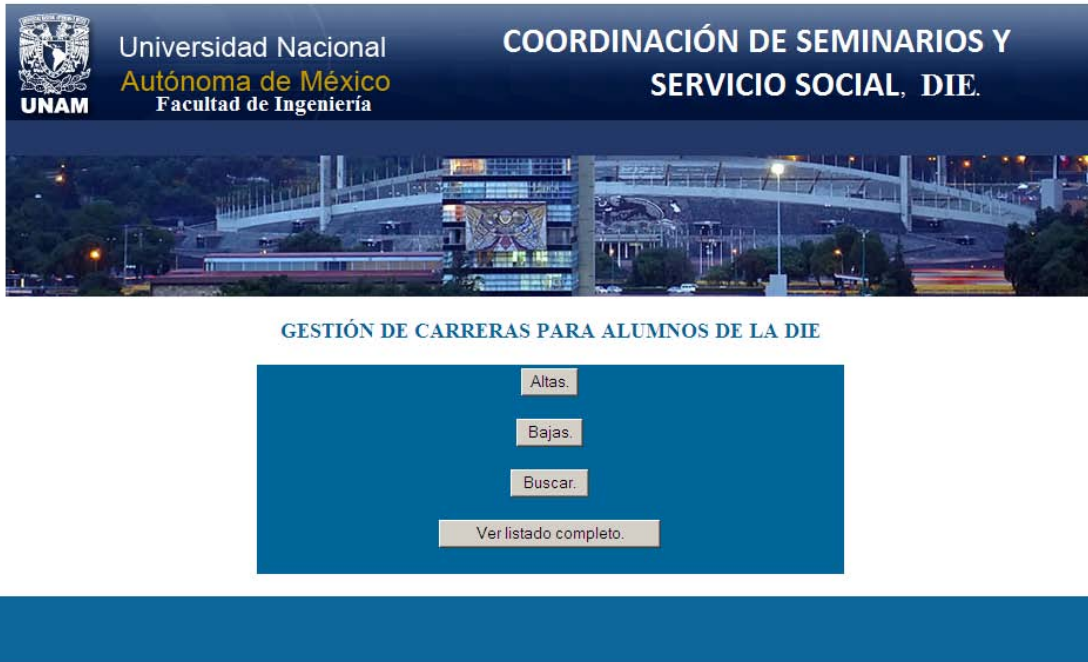


Figura 4. Formulario de gestión de carreras de alumnos.

#### **Función del botón *Altas.***

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta, como se muestra en la figura 4.1





**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- Clave de carreras:  
Clave Carrera  
109 Ingeniería Eléctrica-Electrónica.  
110 Ingeniería en Computación.  
111 Ingeniería en Telecomunicaciones.  
116 Ingeniería Mecánica Eléctrica (Área Eléctrica-Electrónica ).
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a : posdead@yahoo.com.mx o karisscoketa@hotmail.com.

**BASE DE DATOS CARRERAS DE LA DIE.**

Número de cuenta:

Clave de la carrera:

Generación, año de ingreso (aaaa):

Figura 4.1. Formulario de alta de carrera a alumno.

**Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso del número de cuenta del alumno para poder eliminar su registro, como se muestra en la figura 4.2



**BAJA DE CARRERAS DE LA DIE.**

Ingrese el número de cuenta del alumno@:

Figura 4.2. Formulario para baja de carrera de alumno.

**Función del botón *Buscar*.**

Al hacer clic, se muestra un formulario requiriendo el número de cuenta del alumno a buscar, en este caso se utilizó el número de cuenta 300078989, contenido en la base de datos, como se muestra en la figura 4.3

Figura 4.3. Formulario de búsqueda de carrera del alumno.

Y al ser ingresado el número de cuenta y posteriormente al hacer clic al botón: *búsqueda de carrera*, mostrará la información referente a ese alumno, como se muestra en la figura 4.4

## Universidad Nacional Autónoma de México.

### Facultad de Ingeniería.

#### Información de carreras de la DIE.

Número de cuenta	Clave de la carrera	Generación
307171634	110	2010

**Número de registros: 1**

Figura 4.4. Formulario que muestra las carreras del alumno.

#### **Función del botón *Ver listado completo.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los alumnos registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 4.5

**Universidad Nacional Autónoma de México.**  
**Facultad de Ingeniería.**  
**Información de carreras de la DIE.**

Número de cuenta	Clave de la carrera	Generación
307097123	110	2010
307677088	110	2010
410003983	110	2010
410055322	110	2010
410073458	110	2010
410087239	110	2010
307160672	110	2010
307171634	110	2010
300163050	110	2009
302090466	110	2009
303080370	110	2009
303109936	109	2009
304029701	109	2009
304190359	109	2009
305147648	109	2009
305339265	110	2009
306007659	109	2009
306027394	110	2009
306030336	110	2009
306037634	110	2009
306092314	110	2009
306102417	109	2009
306127580	110	2009
306130429	110	2009
306181713	110	2009

Figura 4.5. Formulario de alumnos con carrera (s) registrados en el sistema.

**Funciones de la pantalla correspondiente a la figura 5.**

Existen los siguientes botones:

- **Altas:** Da de alta dependencias de servicio social.
- **Bajas:** Elimina completamente a la dependencia seleccionada a través de la clave de la dependencia.
- **Modificar Dependencia:** Modifica información de la dependencia por medio de la clave de la dependencia.
- **Buscar Dependencia:** Búsqueda de dependencias mediante la clave de la dependencia.
- **Ver dependencias registradas:** Muestra todas las dependencias registradas.



GESTIÓN DE DEPENDENCIAS DE SERVICIO SOCIAL PARA ALUMNOS DE LA DIE.



Figura 5. Formulario de gestión de dependencias de Servicio Social.

**Función del botón *Altas.***

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta, como se muestra en la figura 5.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- Tipos de dependencias:

Clave	Nombre dependencia.
1	UNAM.
2	Gobierno.
3	Descentralizada.
4	Paraestatal.
5	Asociación Civil.
6	Otro.

- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

**BASE DE DATOS DEPENDENCIAS DE SERVICIO SOCIAL.**

Nombre de la dependencia:

Siglas:

Calle:

Código postal:

Colonia:

Delegación:

Ciudad:

Teléfono:

Tipo de dependencia:

Facultad de Ingeniería:

Figura 5.1. Formulario para alta de dependencia.

**Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso de la clave de la dependencia para poder eliminar su registro, como se muestra en la figura 5.2



#### BAJA DE DEPENDENCIAS DE SERVICIO SOCIAL.

Ingrese la clave de la dependencia:

Baja de dependencia.

Figura 5.2. Formulario de baja de dependencia.

#### **Función del botón *Modificar***

Al dar clic se desplegará un formulario indicando que deberá ser ingresada la clave de la dependencia a modificar, en este caso para ejemplificar se utilizó la clave: 168, contenida en la base de datos, como se muestra en la figura 5.3



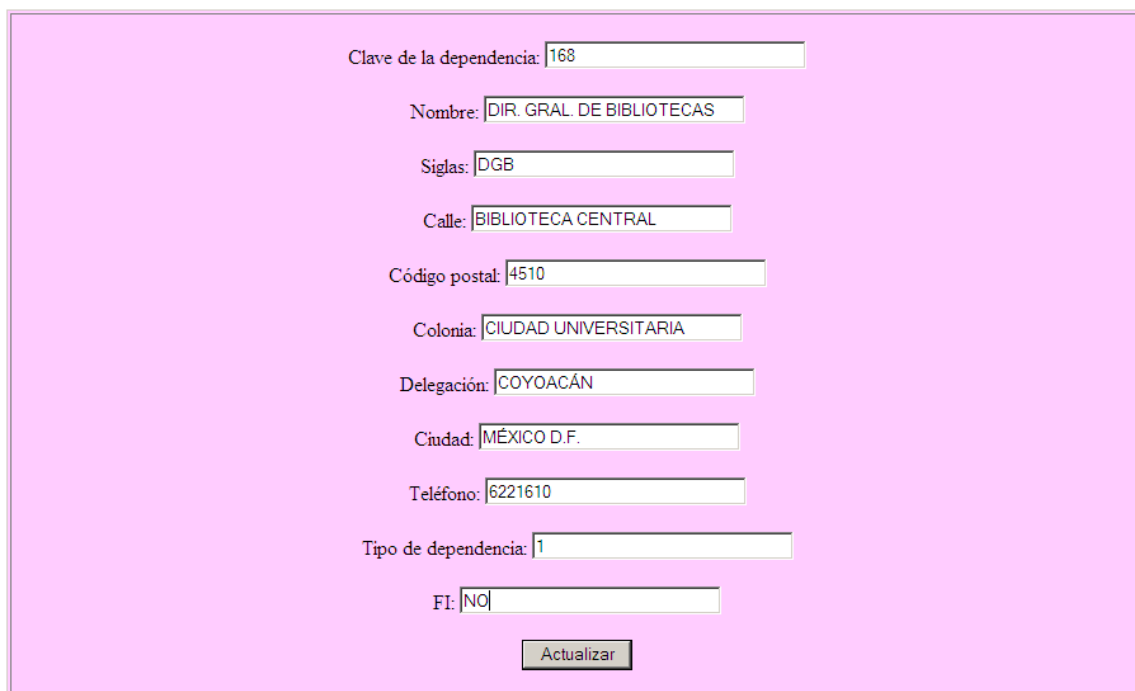
#### MODIFICACIÓN DE DEPENDENCIAS DE SERVICIO SOCIAL.

Ingrese la clave de la dependencia:

Modificar dependencias.

Figura 5.3. Formulario para solicitar modificaciones en datos de dependencia.

Y al buscar la dependencia para hacer las modificaciones, se tiene como resultado una pantalla en la cual se podrán modificar los campos que se deseen a excepción de la clave de la dependencia, como se muestra en la figura 5.4



Clave de la dependencia: 168

Nombre: DIR. GRAL. DE BIBLIOTECAS

Siglas: DGB

Calle: BIBLIOTECA CENTRAL

Código postal: 4510

Colonia: CIUDAD UNIVERSITARIA

Delegación: COYOACÁN

Ciudad: MÉXICO D.F.

Teléfono: 6221610

Tipo de dependencia: 1

FI: NO

Actualizar

Figura 5.4. Formulario de modificación de datos de la dependencia.

#### **Función del botón *Buscar dependencia*.**

Al hacer clic, se muestra un formulario requiriendo la clave de la dependencia a buscar, en este caso se utilizó la clave 168, contenida en la base de datos, como se muestra en la figura 5.5



Universidad Nacional  
Autónoma de México  
UNAM Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y  
SERVICIO SOCIAL, DIE.

BÚSQUEDA DE DEPENDENCIAS DE SERVICIO SOCIAL.

Ingrese la clave de la dependencia: 168

Búsqueda de dependencias.

Figura 5.5. Formulario para búsqueda de dependencia.

Y al ser ingresado la clave de la dependencia, y posteriormente al hacer clic al botón: *búsqueda de dependencia*, mostrará la información referente a esa dependencia de servicio social, como se muestra en la figura 5.6

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de dependencias de servicio social.**

Clave de la dependencia	Nombre	Siglas	Calle	Código postal	Colonia
168	DIR. GRAL. DE BIBLIOTECAS	DGB	BIBLIOTECA CENTRAL	4510	CIUD

Número de dependencias de servicio social por método de búsqueda: 1

Figura 5.6. Formulario que muestra la información de la dependencia.

**Función del botón *Ver dependencias registradas*.**

Al dar clic se desplegará una lista la cual contendrá absolutamente todas las dependencias registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 5.7

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de dependencias de servicio social.**

Clave de la dependencia	Nombre	Siglas	Calle	Código postal	Colonia
808	ACADEMIA MEXICANA DE CIENCIAS		RISCO 255		
132	ADMON. ESCOLAR DE LA FAC. DE ING.	AEFI			
45	AEREOPUERTOS Y SERVICIOS AUXILIARES DE LA SCT	ASA			
800	AEROPUERTO INTERNACIONAL DE LA CIUDAD DE MÉXICO		AV. CAP. CARLOS LEÓN GONZÁLEZ		
359	ANTES CENTRO DE INSTRUMENTOS	CIADET	CU A.P. 70-186		
902	APOYOS Y SERVICIOS A LA COMERCIALIZACIÓN AGROPECUA	ASERCA	JOSÉ MA IBARRAGAN 84 P.B		
169	ASEGURADORA HIDALGO S.A.	AH	PRESIDENTE MASARIK 111		
98	ASOC. DE INGENIEROS UNIVERSITARIOS MEC. ELEC.	AIUM			
102	ASOC. MEXICANA DE CIENCIA Y TECNOLOGIA	AMCT			
251	ASOCIACION AUTONOMA DEL PERSONAL ACADEMICO	APAUNAM	AV. CERRO DEL AGUA NO. 120		
292	ASOCIACION MEXICANA DE CONTADORES PUBLICOS A.C.	AMCP	ESPERANZA NO. 765		
371	ASOCIACION MEXICANA DE EMPRESAS DE INSTALACIONES P	AMERIC	V. SAN ANTONIO NO. 319 - 101		
96	ASOCIACIONES CIVILES	AC			
847	AUDITORIA SUPERIOR DE LA FEDERACIÓN		AV. COYOACAN NUM.1501		
76	BANCO DE MEXICO	B.M.			
915	BANCO DEL AHORRO NACIONAL Y SRVICIOS FINANCIEROS S	BANSEFI	RIO MAGDALENA NO. 115		
182	BANCO NACIONAL DE COMERCIO EXTERIOR	BNCE	PERIFERICO SUR 4333		
249	BANCO NACIONAL DE CREDITO RURAL	BANRURAL	AV. MEXICO COYOACAN NO. 318		
366	BANCO NACIONAL DEL EJERCITO	BANJERCITO	AVENIDA INDUSTRIA MILITAR No. 1055		
313	BANCOMEXT		PERIFERICO SUR NO. 4333		
214	C.A.A.M.T.R.O.H	C.A.A.M.T.	IGNACIO COMONFORT ESQ. IGNACIO RAMÍREZ		
15	CAMARA DE DIPUTADOS				
221	CÁMARA NACIONAL DE EMPRESAS DE CONSULTORÍA	CNEC	CÓRDOBA # 127		

Figura 5.7. Formulario que contiene todas las dependencias registradas.



## Funciones de la pantalla correspondiente a la figura 6.

Existen los siguientes botones:

- **Altas:** Da de alta profesores de la DIE.
- **Bajas:** Elimina completamente el registro del profesor mediante la clave del profesor.
- **Modificar información de profesores:** Modifica la información del profesor mediante la clave del profesor.
- **Buscar profesor:** Búsqueda de profesor mediante la clave del profesor.
- **Buscar profesor por apellidos y nombre:** Búsqueda de profesor mediante su nombre completo.
- **Ver profesores registrados:** Muestra por completo todos los profesores registrados.



### GESTIÓN DE PROFESORES PARA ALUMNOS DE LA DIE.

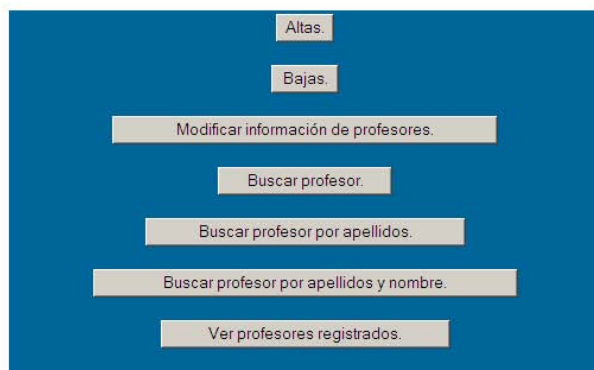


Figura 6. Formulario de gestión de profesores.

### Función del botón *Altas*.

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta. Algunos campos a llenar son: grado, apellidos, nombre, género, RFC, nombre de la dependencia donde trabaja, tipo de adscripción, email, etc., como se muestra en la figura 6.1



Instrucciones:

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios a excepción de los campos: *ingreso, egreso, reingreso, teléfonos, extensión y email*.
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

**BASE DE DATOS PROFESORES.**

Grado:	<input type="text"/>
Nombre(s):	<input type="text"/>
Apellidos:	<input type="text"/>
Género:	<input type="text" value="Hombre"/>
Ingreso (ddmmaaaa):	<input type="text"/>
Egreso (ddmmaaaa):	<input type="text"/>
Reingreso (ddmmaaaa):	<input type="text"/>
RFC:	<input type="text"/>

**DATOS PARTICULARES.**

Calle y número:	<input type="text"/>
Colonia:	<input type="text"/>
Código postal:	<input type="text"/>
Delegación:	<input type="text"/>
Teléfono particular:	<input type="text"/>

**DATOS DEL TRABAJO.**

Dependencia:	<input type="text"/>
Calle:	<input type="text"/>
Colonia:	<input type="text"/>
Código postal:	<input type="text"/>
Delegación:	<input type="text"/>
Teléfono de oficina:	<input type="text"/>
Extensión:	<input type="text"/>
Tipo de adscripción:	<input type="text"/>
Email:	<input type="text"/>

Figura 6.1. Formulario para altas de profesores.

### **Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso de la clave del profesor para poder eliminar su registro, como se muestra en la figura 6.2

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BAJA DE PROFESORES.

Ingrese la clave del profesor:

Baja de profesor.

Figura 6.2. Formulario de baja de profesores.

### **Función del botón *Modificaciones*.**

Al dar clic se desplegará un formulario indicando que deberá ser ingresada la clave del profesor a modificar, en este caso para ejemplificar se utilizó la clave: 711, contenida en la base de datos, como se muestra en la figura 6.3

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

MODIFICAR INFORMACIÓN DE PROFESORES.

Ingrese la clave del profesor: 711

Modificación de profesores.

Figura 6.3. Formulario que solicita modificación de datos del profesor.

Y al buscar al profesor para hacer las modificaciones, se tiene como resultado una pantalla en la cual se podrán modificar los campos que deseen a excepción del número de cuenta.

Los campos del alumno que el usuario podrá modificar serán algunos como: nombre, apellidos, género, dirección, colonia, código postal, delegación, teléfono particular, teléfono de oficina, extensión telefónica, etc., como se muestra en la figura 6.4

Clave: 711

Grado: ING.

Nombre(s): NIKOLAY

Apellidos: SHTIN

Género: M

Ingreso: 0

Egreso: 0

Reingreso: 0

RFC: SINI780718

Calle y número: 8° ORIENTE 20-1

Colonia: ISIDRO FABELA

Código postal: 14030

Delegación: TLALPAN

Teléfono particular: 50339002

Dependencia donde trabaja: FACULTAD DE INGENIERÍA

Calle donde trabaja:

Colonia donde trabaja:

Código postal del trabajo: 4510

Delegación del trabajo:

Teléfono de oficina: 56223071

Extensión: 0

Tipo de adscripción: E

Email:

Actualizar

Figura 6.4. Formulario de modificaciones para datos de profesor.

### **Función del botón *Buscar profesor.***

Al hacer clic, se muestra un formulario requiriendo la clave del profesor a buscar, en este caso se utilizó la clave 711, contenida en la base de datos, como se muestra en la figura 6.5

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE PROFESORES.

Ingrese la clave del profesor: 711

Búsqueda de profesores.

Figura 6.5. Formulario de búsqueda de profesores por clave.

Y al ser ingresado la clave del profesor y posteriormente al hacer clic al botón: *búsqueda de profesor*, mostrará la información referente a ese profesor, como se muestra en la figura 6.6

### **Universidad Nacional Autónoma de México.**

#### **Facultad de Ingeniería.**

#### **Información de profesores.**

Clave del profesor	Grado	Nombre	Apellidos	Género	Ingreso
666	ING.	MARIAN	ABURTO ESTEBANEZ	M	1082005

Número de profesores por método de búsqueda: 1

Figura 6.6. Formulario de información del profesor.

### **Función del botón *Buscar profesor por apellido y nombre(s).***

El usuario al hacer clic se mostrará un formulario requiriendo los datos: nombre(s) y apellidos del profesor, para poder iniciar la búsqueda del alumno.

Este método de búsqueda se realizó debido a la problemática que se tenía, al consultar ciertos datos ya que algunas veces el administrador no recordaba la clave de profesor, pero si recordaba su nombre completo y por ello se implementará un segundo método de búsqueda, como se muestra en la figura 6.7



**BÚSQUEDA DE PROFESORES POR APELLIDO Y NOMBRE.**

Apellidos del profesor:

Nombre(s) del profesor:

Figura 6.7. Formulario de búsqueda de profesores por apellidos y nombre.

***Función del botón Ver profesores registrados.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los profesores registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 6.8

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de profesores.**

Clave del profesor	Grado	Nombre	Apellidos	Género
666	ING.	MARIAN	ABURTO ESTEBANEZ	M
155	DR.	PEDRO JESUS	ACEVEDO CONTLA	M
550	DR.	ESPINOSA BAUTISTA	ADRIÁN	M
514	M.C.	EDGAR BALDEMAR	AGUADO CRUZ	M
610	ING.	CRUZ SERGIO	AGUILAR DÍAZ	M
613	DR.	HERIBERTO DE JESÚS	AGUILAR JUÁREZ	M
531	ING.	HERNÁNDEZ QUINTO	AIDA ALICIA	F
479	M.C	REYNALDO	ALANIS CANTU	M
436	MI	EDUARDO	ALARCÓN AVILA	M
567	ING.	CORTEZ MONDRAGÓN	ALBERTO	M
445	ING.	ALBERTO A.	ALCANTARA MELENDEZ	M
345	DR.	ROGELIO	ALCANTARA SILVA	M
47	ING.	EUGENIO	ALMANZA CASTRO	M
390	DR.	RAFAEL	ALMANZA SALGADO	M
400	ING.	ELEUTERIA SILVINA	ALONSO SALINAS	F
105	M. I.	JESÚS	ALVAREZ CASTILLO	M
470	DR.	LUIS AGUSTIN	ALVAREZ ICAZA LONGORIA	M
87	ING.	JESÚS MANUEL	ÁLVAREZ LÓPEZ	M
577	ING.	GABRIEL MAURICIO	ALVAREZ MEDINA	M
903	ING.	PABLO	ALVAREZ WATKINS	M
653	DR.	PABLO	ALVAREZ WITKINS	M
234	ING.	SERGIO	AMBRIZ MAGUEY	M
407	ING.	MARCO	AMBRIZ MAGUEY	M
454	M. I.	RUBEN	ANAYA GARCIA	M
887	ING.	JESÚS MAURICIO	ANDRADE GUZMÁN	M

Figura 6.8. Formulario de profesores registrados en el sistema.

## Funciones de la pantalla correspondiente a la figura 7.

Existen los siguientes botones:

- **Altas:** Da de alta programa de servicio social.
- **Bajas:** Elimina completamente el registro del programa del servicio social mediante la clave del programa.
- **Modificar información de profesores:** Modifica la información del profesor mediante la clave del programa.
- **Buscar programa:** Búsqueda de programa de servicio social mediante la clave del programa.
- **Ver programas registrados:** Muestra por completo todos los programas de servicio social registrados.



Figura 7. Formulario de gestión de programas de Servicio Social.

### Función del botón *Altas*.

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta. Algunos campos a ser llenados son: clave del programa, nombre del programa, tipo de programa, etc., como se muestra en la figura 7.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

**BASE DE DATOS PROGRAMAS DE SERVICIO SOCIAL.**

Clave del programa:	<input type="text"/>
Nombre del programa:	<input type="text"/>
Tipo de programa:	<input type="text"/>

**RESPONSABLE ADMINISTRATIVO.**

Nombre:	<input type="text"/>
Cargo:	<input type="text"/>
Telefono:	<input type="text"/>

Figura 7.1. Formulario para alta de programas de Servicio Social.

**Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso de la clave del programa del servicio social, para poder eliminar su registro, como se muestra en la figura 7.2



Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BAJA DE PROGRAMAS DE SERVICIO SOCIAL.

Ingrese la clave del programa:

Baja de programa.

Figura 7.2. Formulario que solicita baja de programa de Servicio Social.

#### **Función del botón *Modificar***

Al dar clic se desplegará un formulario indicando que deberá ser ingresada la clave del programa del servicio social a modificar, en este caso para ejemplificar se utilizó la clave: 20000120354825, contenida en la base de datos, como se muestra en la figura 7.3

Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

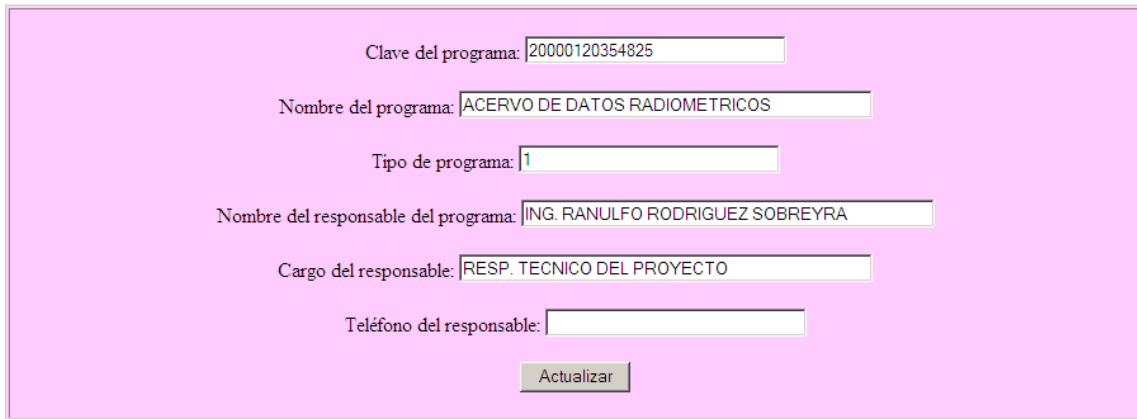
MODIFICACIÓN DE PROGRAMAS DE SERVICIO SOCIAL.

Ingrese la clave del programa: 20000120354825

Modificar programa.

Figura 7.3. Formulario para solicitar modificación de programa de Servicio Social.

Y darle clic al botón *Modificar programa* para hacer las modificaciones, se tendrá como resultado una pantalla en la cual se podrán modificar los campos que se deseen a excepción de la clave del programa del servicio social, como se muestra en la figura 7.4



Clave del programa: 20000120354825

Nombre del programa: ACERVO DE DATOS RADIOMETRICOS

Tipo de programa: 1

Nombre del responsable del programa: ING. RANULFO RODRIGUEZ SOBREYRA

Cargo del responsable: RESP. TECNICO DEL PROYECTO

Teléfono del responsable:

Actualizar

Figura 7.4. Formulario para modificar datos del programa de Servicio Social.

#### **Función del botón *Buscar programa*.**

Al hacer clic, se muestra un formulario que requerirá la clave del programa del servicio social a buscar, en este caso se utilizó la clave: 20000120354825, contenida en la base de datos, como se muestra en la figura 7.5



Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE PROGRAMAS DE SERVICIO SOCIAL.

Ingrese la clave del programa: 20000120354825

Búsqueda de programa.

Figura 7.5. Formulario de búsqueda de programa de Servicio Social.

Y al ser ingresada la clave del programa del servicio social y posteriormente al hacer clic al botón: *búsqueda de programa*, mostrará la información referente a ese programa de servicio social, como se muestra en la figura 7.6

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de programas de servicio social.**

Clave del programa	Nombre del programa	Tipo de programa	Nombre del responsable
20000120354825	ACERVO DE DATOS RADIOMETRICOS	1	ING. RANULFO RODRIGUEZ SOBREYRA

Número de programas de servicio social por método de búsqueda: 1

Figura 7.6. Formulario que muestra información del Servicio Social.

**Función del botón *Ver programas registrados*.**

Al dar clic se desplegará una lista la cual contendrá absolutamente todos programas de servicio social registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 7.7

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de programas de servicio social.**

Clave del programa	Nombre del programa	Tipo de programa	Nombre del responsable
20111020052686	Proyecto de reingeniería de procesos	2	LIC. MARIA ELENA MALACARA DOMINGUEZ
20110520312856	CREATIVIDAD INFORMATICA PARA LA SEGURIDAD DEL PACIENTE Y LA PREVENCION DE CONFLICTOS EN LA ATENCION A LA SALUD	3	DRA. ESTHER MAHUINA CAMPOS CASTOLO
20110220312949	DESARROLLO, INSTRUMENTACIÓN Y ACTUALIZACIÓN DE LOS SISTEMAS, PROGRAMAS Y PROYECTOS EN EL IMSS	4	C. GILBERTO POBLETE BEDOLLA
20110120992614	INSTALACION Y ADMINISTRACION DE LOS SERVICIOS DE RED	5	IVETTE DE LUNA BONILLA
20110120950432	PROYECTO DE INGENIERÍA	2	JOSÉ LUIS FERNANDEZ ZAYAS
20110120950378	FUNDAMENTACIÓN TÉCNICA DE PROYECTO DE DESALACIÓN CON ENERGIAS RENOVABLES	1	OCEAN. LIUS JOSÉ TREVIÑO ACUÑA
20110120810526	APOYO DOCENTE A LA CARRERA DE INGENIERIA EN TELECOMUNICACIONES	1	MIGUEL MOCTEZUMA FLORES
20110120810202	DESARROLLO E IMPLEMENTACIÓN DE UN CURSO A DISTANCIA SOBRE LA ASIGNATURA CURRICULAR DE SISTEMAS DE CONTROL	1	M.I. RICARDO GARIBAY JIMÉNEZ
20110120432642	REALIZACIÓN Y PRODUCCIÓN DE MATERIALES TELEVISIVOS	3	LIC. MARÍA ISABEL ARMENTA SOSA
20110120292447	SERVICIO SISMOLÓGICO NACIONAL	3	DR. CARLOS VALDÉS GONZÁLEZ
20110120230184	DESARROLLO DE SISTEMAS PARA EL APOYO ADMINISTRATIVO, DOCENTE Y DE INVESTIGACIÓN	3	DRA. MARIANA GUTIERREZ LARA
20110120033227	APOYO A LA ELABORACIÓN DE MATERIAL DIDÁCTICO PARA LA ENSEÑANZA DE LA QUÍMICA EN LA FACULTAD DE INGENIERÍA	1	DRA. MARÍA DEL PILAR CARREÓN CASTRO
20070120811865	MODOS DESLIZANTES	1	RAFAEL IRIARTE VIVARBALDERRAMA
20070120810633	APOYO ACADÉMICO ADMINISTRATIVO DE LA COORDINACIÓN DE SEMINARIOS Y TITULACIÓN DE LA DIVISIÓN DE INGENIERÍA MECÁNICA INDUS	2	M.I. H'CTOR RAUL MEJÍA RAMÍREZ
20040520602491	INTEGRACIÓN DE LA SALUD CON LAS TECNOLOGÍAS DE LA INGENIERÍA BIOMÉDICA	3	DR. SAÚL RENÁN LEÓN HERNÁNDEZ
20034210024622	APOYO EN EL MANTENIMIENTO A LAS REDES LAN Y WAN DEL TRIBUNAL ELECTORAL DEL PODER JUDICIAL DE LA FEDERACION	4	ING. FERNANDO RESENDIZ FLORES
20032640032780	MEDIDAS PARA EL AHORRO DE ENERGIA	3	ING. LUIS ACOSTA DE LAGO

Figura 7.7. Formulario que muestra los programas de Servicio Social registrados en el sistema.  
**Funciones de la pantalla correspondiente a la figura 8.**

Existen los siguientes botones:

- **Alta de reporte:** Da de alta un reporte de servicio social.
- **Baja de reporte:** Elimina completamente el reporte de servicio social, mediante el ID del reporte y el número de reporte.
- **Buscar reporte:** Búsqueda de un reporte de servicio social, mediante el ID del reporte.
- **Buscar reporte por número de cuenta:** Búsqueda de un reporte de servicio social, mediante el número de cuenta del alumno.
- **Ver listado completo de reportes:** Muestra por completo todos los reportes de servicio social registrados.



GESTIÓN DE REPORTES ENTREGADOS DE SERVICIO SOCIAL PARA ALUMNOS DE LA DIE.

Alta de reporte.

Baja de reporte.

Buscar reporte.

Buscar reporte por número de cuenta.

Ver listado completo de reportes.

Figura 8. Formulario de gestión de reportes de Servicio Social.

**Función del botón *Alta de reporte.***

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta, como se muestra en la figura 8.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

**BASE DE DATOS REPORTE ENTREGADOS.**

ID Reporte (aammnn):	<input type="text"/>
Año (aa):	<input type="text"/>
Mes (mm):	<input type="text"/>
Número (nn):	<input type="text"/>
Número de reporte:	<input type="text"/>
Fecha de entrega de reporte (ddmmaaaa):	<input type="text"/>

Figura 8.1. Formulario para alta de reporte entregado.

**Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso del ID del reporte y el número de reporte, como se muestra en la figura 8.2



#### BAJA DE REPORTES.

Ingrese el ID del reporte (aammnnn):

Ingrese el número del reporte:

Figura 8.2. Formulario de baja de reporte de Servicio Social.

#### **Función del botón *Buscar reporte.***

Al hacer clic, se muestra un formulario requiriendo el ID del reporte a buscar, en este caso se utilizó el ID: 764, contenido en la base de datos, como se muestra en la figura 8.3



#### BÚSQUEDA DE REPORTES ENTREGADOS.

Ingrese el ID del reporte:

Figura 8.3. Formulario de búsqueda de reporte entregado.

Y al ser ingresado el ID, y posteriormente al hacer clic al botón: *búsqueda de reporte*, mostrará la información referente a ese reporte de servicio social, en este caso mostrará tres resultados, cabe señalar que no significa que este duplicado, este resultado se debe a que todos los alumnos que estuvieron o estén realizando su servicio social debieron haber entregado por lo menos tres reportes, en caso de que su servicio social haya tenido una duración mínima de seis meses, como fue en este caso y en otros tal vez habrá hasta doce reportes, como se muestra en la figura 8.4

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de alumnos.**

Número de cuenta del alumno	ID Servicio social	Promedio	Créditos	Inicio	Duración
305094250	114	7.3	37.25	18012010	6
305094250	114	7.3	37.25	18012010	6
305094250	114	7.3	37.25	18012010	6

Número de alumnos por método de búsqueda: 3

Figura 8.4. Formulario que muestra los reportes entregados.

**Función del botón *Buscar reporte por número de cuenta*.**

Al hacer clic, se muestra un formulario requiriendo el número de cuenta, en este caso se utilizó el número de cuenta siguiente: 300012295, contenido en la base de datos, como se muestra en la figura 8.5

Figura 8.5. Formulario para búsqueda de reportes de Servicio Social por número de cuenta del alumno.



Y al ser ingresado el número de cuenta del alumno, y posteriormente al hacer clic al botón: *búsqueda de reportes*, mostrará la información requerida, como se muestra en la Figura 8.6

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de alumnos.**

Número de cuenta del alumno	ID Servicio social	Promedio	Créditos	Inicio	Duración
305094250	114	7.3	37.25	18012010	6
305094250	114	7.3	37.25	18012010	6
305094250	114	7.3	37.25	18012010	6

Número de alumnos por método de búsqueda: 3

Figura 8.6. Formulario de información de reportes del alumno de Servicio Social.

**Función del botón *Ver listado completo de reportes.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los reportes de servicio social registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 8.7

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de carreras de la DIE.**

ID reporte	Año	Mes	Número	Número de reporte	Reporte
0	0	0	0	0	0
114	1	1	4	1	20100324
114	1	1	4	2	20100519
114	1	1	4	3	20100813
731	7	3	1	1	20070522
731	7	3	1	2	20070706
731	7	3	1	3	20070919
732	7	3	2	1	20070504
732	7	3	2	2	20070706
732	7	3	2	3	20070911
733	7	3	3	1	20070524
733	7	3	3	2	20070803
733	7	3	3	3	20070925
733	7	3	3	4	20071127
733	7	3	3	5	20080125
733	7	3	3	6	20080401
735	7	3	5	1	20070517
735	7	3	5	2	20070803
735	7	3	5	3	20070917
736	7	3	6	1	20070522
736	7	3	6	2	20070803
736	7	3	6	3	20071015
737	7	3	7	1	20070503
737	7	3	7	2	20070704
737	7	3	7	3	20070904

Figura 8.7. Formulario de reportes registrados en el sistema.

### Funciones de la pantalla correspondiente a la figura 9.

Existen los siguientes botones:

- **Altas:** Da de alta un alumno en un seminario de tesis.
- **Bajas:** Elimina completamente el registro del alumno inscrito en un seminario de tesis mediante su número de cuenta.
- **Modificaciones del alumno:** Modifica la información del alumno mediante su número de cuenta.
- **Buscar alumno:** Búsqueda de alumno mediante su número de cuenta.
- **Buscar por ID de tema:** Búsqueda de alumno mediante su ID de tema.
- **Ver alumnos registrados:** Muestra por completo todos los registros actuales en la base de datos.



### GESTIÓN DE SEMINARIOS Y/O TESIS PARA ALUMNOS DE LA DIE.

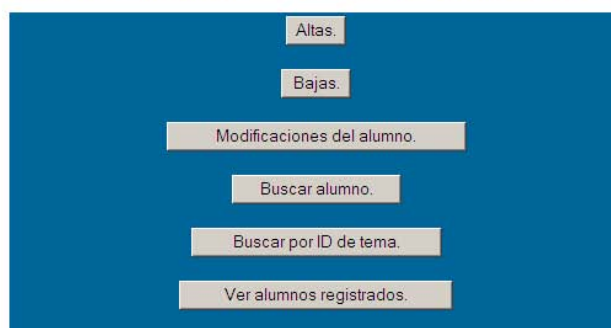


Figura 9. Formulario de gestión de seminarios de tesis.

### Función del botón *Altas*.

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta. Algunos campos a ser llenados son: número de cuenta, clave carrera, créditos, promedio, tipo de inscripción, ID del tema, observaciones, etc., como se muestra en la figura 9.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- Clave de carreras:  
Clave Carrera  
109 Ingeniería Eléctrica-Electrónica.  
110 Ingeniería en Computación.  
111 Ingeniería en Telecomunicaciones.  
116 Ingeniería Mecánica Eléctrica (Área Eléctrica-Electrónica).
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

**BASE DE DATOS SEMINARIO DE TESIS.**

**INFORMACIÓN DEL ALUMNO.**

Número de cuenta:	<input type="text"/>
Clave de carrera:	<input type="text" value="109"/>
Créditos:	<input type="text"/>
Promedio:	<input type="text"/>
Inscripción:	<input type="text" value="0"/>

**TEMA.**

ID (aasn):	<input type="text"/>
Año (aa):	<input type="text"/>
Semestre (s):	<input type="text"/>
Número (nn):	<input type="text"/>

**SEMINARIO DE TESIS.**

Fecha de alta seminario (ddmmaaaa):	<input type="text"/>
Fecha de baja seminario (ddmmaaaa):	<input type="text"/>
Fecha de calificación seminario (ddmmaaaa):	<input type="text"/>
Calificación:	<input type="text" value="Sin cursar seminario"/>
Fecha de alta tesis (ddmmaaaa):	<input type="text"/>
Fecha de baja tesis (ddmmaaaa):	<input type="text"/>
Fecha de titulación (ddmmaaaa):	<input type="text"/>
Observaciones:	<input type="text"/>

Figura 9.1. Formulario para alta de alumno a seminario de tesis.

### **Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá del número de cuenta, para poder eliminar su registro, como se muestra en la figura 9.2



Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BAJA DE ALUMNOS DE SEMINARIOS Y/O TESIS.

Ingrese el número de cuenta del alumno@:

Baja de alumno.

Figura 9.2. Formulario para solicitar baja de alumno en el sistema.

### **Función del botón *Modificaciones del alumno*.**

Al dar clic se desplegará un formulario indicando que deberá ser ingresado el número de cuenta a modificar, en este caso para ejemplificar se utilizó el número de cuenta siguiente: 30004258, contenido en la base de datos, como se muestra en la figura 9.3



Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

MODIFICACIÓN DE DATOS DE ALUMNOS.

Ingrese el número de cuenta del alumno@:

Modificar alumno.

Figura 9.3. Formulario que solicita modificación de datos del alumno.

Y al hacer clic al botón: *modificar alumno*, obtenemos como resultado una pantalla en la cual se podrán modificar los campos que deseen a excepción del número de cuenta.

Los campos del alumno que el usuario podrá modificar serán algunos como: clave carrera, créditos, promedio, tipo de inscripción, ID del tema, observaciones, etc., es decir los campos que fueron ingresados al registrarse al seminario de tesis, a excepción del número de cuenta del alumno, como se muestra en la figura 9.4

Formulario para modificar datos de alumno en inscripción a seminario. El formulario contiene los siguientes campos de entrada:

- Número de cuenta: 30004258
- Clave de carrera: 110
- Créditos: 100.22
- Promedio: 8.91
- Inscripción: E
- ID: 82111
- Año: 8
- Semestre: 2
- Número: 111
- Fecha de alta seminario: 2008-04-15
- Fecha de baja seminario: 0000-00-00
- Fecha de calificación Seminario: 0000-00-00
- Calificación: 0
- Fecha de alta tesis: 0000-00-00
- Fecha de baja tesis: 0000-00-00
- Fecha de titulación: 0000-00-00
- Observaciones:

Botón: Actualizar

Figura 9.4. Formulario para modificar datos de alumno en inscripción a seminario.

#### **Función del botón *Buscar alumno*.**

Al hacer clic, se muestra un formulario requiriendo la clave del profesor a buscar, en este caso se utilizó el número de cuenta: 30004258, contenido en la base de datos, como se muestra en la figura 9.5

Figura 9.5. Formulario para búsqueda de alumno en seminario de tesis.

Y al ser ingresado el número de cuenta y posteriormente al hacer clic al botón: *búsqueda de alumno*, mostrará la información referente a ese alumno, como se muestra en la figura 9.6

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de seminarios y/o tesis.**

Número de cuenta del alumno	Clave de la carrera	Créditos	Promedio	Inscripción	ID Tema
88309255	111	100	8	E	11
88309255	111	100	8	E	11
92294833	110	101.92	8.62	O	12
92294833	110	101.92	8.62	O	12
91545132	116	102.9	8.53	E	13
91545132	116	102.9	8.53	E	13
88400778	116	100.96	8.17	E	13
88400778	116	100.96	8.17	E	13
89338009	116	100.96	7.7	E	13
89338009	116	100.96	7.7	E	13
85350904	110	99	8	E	14
85350904	110	99	8	E	14
85145218	110	99	8	E	14
85145218	110	99	8	E	14
81000353	110	99	8	E	15
81000353	110	99	8	E	15
86302755	110	99	8	E	15
86302755	110	99	8	E	15
88402538	116	100.96	8.64	E	16
88402538	116	100.96	8.64	E	16
91275673	110	96.63	8.21	O	17
91275673	110	96.63	8.21	O	17
91356499	110	96.87	8.29	O	17
91356499	110	96.87	8.29	O	17
89267963	110	99.03	7.9	O	18

Figura 9.6. Información de seminarios de tesis.

### **Función del botón *Buscar por ID de tema.***

Al hacer clic, se muestra un formulario requiriendo el ID del tema a buscar, en este caso se utilizó el ID: 111106, contenido en la base de datos, como se muestra en la figura 9.7

Y al ser ingresado el ID del tema, y posteriormente al hacer clic al botón: *búsqueda de alumno*, mostrará la información referente a ese tema. Y cabe destacar que en este ejemplo se encuentran inscritos dos alumnos en ese seminario de tesis, ya que es el número de vacantes permitido por el profesor, como se muestra en la figura 9.8

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE ALUMNOS EN SEMINARIOS DE TESIS.

Ingrese ID del tema:

Figura 9.7. Búsqueda de alumnos en seminarios de tesis.

### **Universidad Nacional Autónoma de México.**

#### **Facultad de Ingeniería.**

#### **Información de seminarios y/o tesis.**

Número de cuenta del alumno	Clave de la carrera	Créditos	Promedio	Inscripción	ID Tema	Año	Semestre	Número
68151627	116	85.44	7.34	E	111106	11	1	106
95317939	109	99.1	7.53	E	111106	11	1	106

Número de alumnos en seminarios y/o tesis: 2

Figura 9.8. Información de la búsqueda en seminario de tesis.

### **Función del botón *Ver alumnos registrados.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los alumnos registrados en seminarios de tesis y a su vez indicando el total de registros que se tienen, como se muestra en la figura 9.9

Universidad Nacional Autónoma de México.

Facultad de Ingeniería.

Información de seminarios y/o tesis.

Número de cuenta del alumno	Clave de la carrera	Créditos	Promedio	Inscripción	ID Tema
88309255	111	100	8	E	11
88309255	111	100	8	E	11
92294833	110	101.92	8.62	O	12
92294833	110	101.92	8.62	O	12
91545132	116	102.9	8.53	E	13
91545132	116	102.9	8.53	E	13
88400778	116	100.96	8.17	E	13
88400778	116	100.96	8.17	E	13
89338009	116	100.96	7.7	E	13
89338009	116	100.96	7.7	E	13
85350904	110	99	8	E	14
85350904	110	99	8	E	14
85145218	110	99	8	E	14
85145218	110	99	8	E	14
81000353	110	99	8	E	15
81000353	110	99	8	E	15
86302755	110	99	8	E	15
86302755	110	99	8	E	15
88402538	116	100.96	8.64	E	16
88402538	116	100.96	8.64	E	16
91275673	110	96.63	8.21	O	17
91275673	110	96.63	8.21	O	17
91356499	110	96.87	8.29	O	17
91356499	110	96.87	8.29	O	17
89267963	110	99.03	7.9	O	18

Figura 9.9. Información de seminario de tesis.

**Funciones de la pantalla correspondiente a la figura 10.**

Existen los siguientes botones:

- **Altas:** Da de alta un servicio social.
- **Bajas:** Elimina completamente el registro de un servicio social mediante: el número de cuenta del alumno, el ID del servicio social y el status del servicio social.
- **Modificaciones del servicio social:** Modifica la información del alumno mediante: el número de cuenta del alumno, el ID del servicio social y el status del servicio social.
- **Buscar servicio social:** Búsqueda de un servicio social mediante el número de cuenta del alumno.
- **Ver servicios sociales:** Muestra por completo todos los servicios sociales registrados.





### GESTIÓN DE SERVICIO SOCIAL PARA ALUMNOS DE LA DIE.

Altas.

Bajas.

Modificaciones del servicio social.

Buscar servicio social.

Ver servicios sociales.

Figura 10. Formulario de gestión de servicio social.

#### **Función del botón *Altas.***

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta. Algunos campos a ser llenados son: grado, apellidos, nombre, género, RFC, nombre de la dependencia donde trabaja, tipo de adscripción, email, etc., como se muestra en la figura 10.1



Instrucciones:

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios a excepción de comentarios, fecha de terminación y clave de la dependencia.
- Clave de carreras:  
Clave Carrera  
109 Ingeniería Eléctrica-Electrónica.  
110 Ingeniería en Computación.  
111 Ingeniería en Telecomunicaciones.  
116 Ingeniería Mecánica Eléctrica (Área Eléctrica-Electrónica).
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: [posdead@yahoo.com.mx](mailto:posdead@yahoo.com.mx) o [karisscoketa@hotmail.com](mailto:karisscoketa@hotmail.com).

BASE DE DATOS SERVICIO SOCIAL.

Número de cuenta:	<input type="text"/>
Clave de carrera:	<input type="text" value="109"/>
Créditos:	<input type="text"/>
Promedio:	<input type="text"/>

SERVICIO SOCIAL.

ID del servicio social (aammnn):	<input type="text"/>
Año del servicio social (aa):	<input type="text"/>
Mes del servicio social (mm):	<input type="text"/>
Número del servicio social (nnn):	<input type="text"/>
Fecha de inicio (ddmmaaaa):	<input type="text"/>
Duración en meses:	<input type="text" value="6"/>
Clave de la dependencia:	<input type="text"/>
Clave DGOSE:	<input type="text"/>
Status (alta o baja):	<input type="text" value="A"/>
Fecha de terminación (ddmmaaaa):	<input type="text"/>
Comentarios:	<input type="text"/>

Figura 10.1. Formulario de altas para servicio social.

### **Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá el ingreso del número de cuenta del alumno, el ID del servicio social y el status del servicio social, para poder eliminar su registro, como se muestra en la figura 10.2

The image shows a web interface for the 'BAJA DE SERVICIO SOCIAL' (Social Service Cancellation) process. At the top, there is a header with the UNAM logo and the text 'Universidad Nacional Autónoma de México Facultad de Ingeniería' on the left, and 'COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.' on the right. Below the header is a photograph of a modern building at night. The main content area is titled 'BAJA DE SERVICIO SOCIAL.' and contains a blue-bordered form with three input fields: 'Ingrese el número de cuenta del alumn@:', 'Ingrese el ID del servicio social:', and 'Ingrese el status del servicio social:'. Below these fields is a button labeled 'Baja de servicio social.'

Figura 10.2. Baja de servicio social.

### **Función del botón *Modificaciones del servicio social***

Al dar clic se desplegará un formulario indicando que deberá ser ingresado el número de cuenta del alumno, el ID del servicio social y el status del servicio social, en este caso para ejemplificar se utilizarán:

Número de cuenta: 81617625

ID del servicio social: 2001749

Status del servicio social: B

Los datos anteriores son reales y están contenidos en la base de datos, como se muestra en la figura 10.3



### MODIFICACIÓN DE DATOS DE ALUMNOS.

Ingrese el número de cuenta del alumn@:

Ingrese el ID del servicio social:

Ingrese el status del servicio social:

Figura 10.3. Modificación de servicio social.

Y al dar clic al botón modificar datos del alumno se obtendrá la siguiente pantalla, como se aprecia en la figura 10.4

Número de cuenta:

Clave de carrera:

Créditos:

Promedio:

ID del servicio social:

Año del servicio social:

Mes del servicio social:

Número del servicio social:

Fecha de inicio:

Duración en meses:

Clave de la dependencia:

Clave DGOSE:

Status:

Fecha de terminación:

Comentarios:

Figura 10.4. Actualización de servicio social.

### **Función del botón Buscar *servicio social*.**

Al hacer clic, se mostrará un formulario requiriendo el número de cuenta del alumno, en este caso se utilizó el siguiente: 81617625, contenido en la base de datos, como se muestra en la figura 10.5



Figura 10.5. Búsqueda de servicio sociales.

Y al ser ingresado el número de cuenta, y posteriormente al hacer clic al botón: *búsqueda de servicio social*, mostrará la información referente a ese alumno y su servicio social, como se muestra en la figura 10.6

### **Universidad Nacional Autónoma de México.**

#### **Facultad de Ingeniería.**

#### **Información de servicios sociales.**

Número de cuenta	Clave de carrera	Créditos	Promedio	ID	Año del servicio social	Mes del servicio social
81617625	116	98.56	7.83	2001749	2001	7

Número de servicios sociales por método de búsqueda: 1

Figura 10.6. Información solicitada en la búsqueda de servicio social.

### **Función del botón *Ver servicios sociales*.**

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los servicios sociales registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 10.7

Universidad Nacional Autónoma de México.

Facultad de Ingeniería.

Información de servicio social.

Número de cuenta	Clave de carrera	Créditos	Promedio	ID	Año del servicio social	Mes del servicio social
300135770	110	70.83	8.88	2006122006	2006	12
97110327	110	77	8.3	2003102003	2003	10
407083077	110	408	8.9	123120314	2012	3
303098111	109	417	8.18	123120310	2012	3
300251184	109	324	8.83	123120309	2012	3
305651840	109	397	9	123120308	2012	3
305205618	110	199	7.04	121120131	2012	1
300026492	110	100	8.07	20081121	2008	11
99043924	109	93.08	7.51	20081011	2008	10
300278471	109	98.88	8.07	20078105	2007	8
99085126	110	83.82	7.54	20078102	2007	8
96022458	110	100	7.69	20078101	2007	8
99055842	110	96.2	7.96	20078100	2007	8
99018748	110	90.17	8.17	20071215	2007	12
97005078	109	98.88	8.25	20071214	2007	12
301303970	110	80.58	9.22	20071212	2007	12
300304273	109	93.43	8.18	20071210	2007	12
302222535	110	37.74	7.9	20071120	2007	11
99135537	109	87.72	8.06	20071119	2007	11
403052330	110	45	7.95	20071118	2007	11
404103789	109	76.15	8.05	20071117	2007	11
99259703	109	84.59	8	20071115	2007	11
301116004	111	79.02	9.3	20071114	2007	11
99195401	110	89.7	7.88	20071113	2007	11
300246595	110	97.05	8.89	20071030	2007	10

Figura 10.7. Información de servicio sociales.

**Funciones de la pantalla correspondiente a la base de datos para la gestión de temas de tesis.**

Existen los siguientes botones:

- **Altas:** Da de alta un tema de tesis.
- **Bajas:** Elimina completamente el registro de un tema de tesis, mediante el ID del tema.
- **Buscar temas registrados:** Busca el ID del tema que se requiera consultar.
- **Buscar tema con director y codirector:** Busca el tema requerido junto con sus correspondientes director y codirector.
- **Buscar alumnos inscritos con profesores:** Despliega una lista de los alumnos inscritos con el profesor que deseemos consultar.
- **Buscar alumnos inscritos en temas:** Muestra por completo todos los alumnos inscritos en un tema de tesis.
- **Buscar directores y codirectores vía apellido y nombre:** Búsqueda de director o codirector por medio de su nombre completo o su apellido, en caso de no saber su clave del mismo.
- **Ver temas registrados:** Muestra todos los registros de temas de tesis, que se tengan en la base de datos.

Como se muestran en la figura 11.



GESTIÓN DE TEMAS PARA ALUMNOS DE LA DIE.

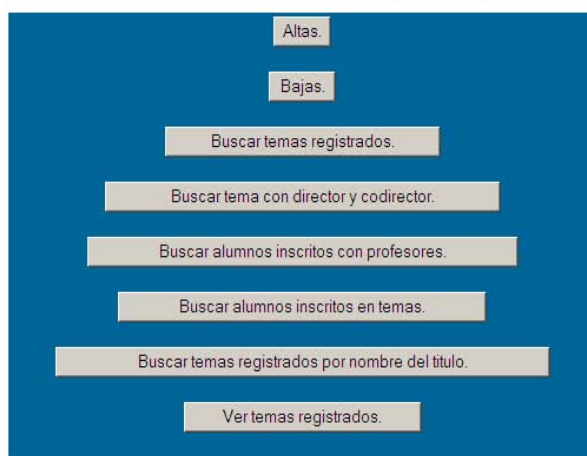


Figura 11. Gestión de temas de tesis para alumnos de licenciatura.

**Función del botón *Altas.***

Al dar clic se mostrará un formulario, el cual deberá ser llenado exactamente como se indica en las instrucciones, en caso contrario no procederá la alta, como se muestra en la figura 11.1



**Instrucciones:**

- Leer y contestar cuidadosamente cada sección del registro.
- Todos los campos son obligatorios.
- En caso de detectar algún acto fraudulento, se actuará conforme a la Legislación Universitaria.
- En caso de detectarse alguna falla en el sistema enviar correo a: posdead@yahoo.com.mx o karisscoketa@hotmail.com.

BASE DE DATOS TEMAS.

Año del tema (aa):

Semestre del tema (s):

Número del tema (nnn):

ID del tema (aasnnn):

Título:

CUPOS EN DIVISIONES.

Cupo en Ingeniería Mecánica Eléctrica (Área Eléctrica-electrónica):

Cupo en Ingeniería en Computación:

Cupo en Ingeniería en Telecomunicaciones:

Cupo en Ingeniería Eléctrica-Electrónica:

ELIGE UNA OPCIÓN.

FI:

PAT:

Pagado:

Carga:

Seminario:

Tesis:

Observaciones:

HORARIO.

**Inicia:** Lunes (00:00:00):  Martes (00:00:00):  Miércoles (00:00:00):  Jueves (00:00:00):   
Viernes (00:00:00):  Sábado (00:00:00):

**Termina:** Lunes (00:00:00):  Martes (00:00:00):  Miércoles (00:00:00):  Jueves (00:00:00):   
Viernes (00:00:00):  Sábado (00:00:00):

INFORMACIÓN DEL DIRECTOR REVISOR.

Director (clave):

Codirector (clave):

Fecha de aprobación (ddmmaaaa):

Figura 11.1. Formulario para temas de tesis.



### **Función del botón *Bajas*.**

Al dar clic se mostrará un formulario, el cual requerirá ID del tema de tesis, para poder eliminar su registro, como se muestra en la figura 11.2



UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BAJA DE TEMAS.

Número ID del tema (aasnmm):

Baja de temas.

Figura 11.2. Baja de temas de tesis.

### **Función del botón *Búsqueda de temas*.**

Al hacer clic, se muestra un formulario requiriendo el ID del tema a buscar, en este caso se utilizó el ID: 22, contenido en la base de datos, como se muestra en la figura 11.3



UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE TEMAS.

Número ID del tema: 22

Búsqueda de temas.

Figura 11.3. Búsqueda de temas de tesis.

Y al ser ingresado el ID del tema y posteriormente al hacer clic al botón: *búsqueda de temas*, mostrará la información referente a ese tema de tesis, como por ejemplo el nombre del tema de tesis, el horario de impartición del seminario de tesis, la clave del director, fecha de aprobación, etc., como se muestra en la figura 11.4

Universidad Nacional Autónoma de México.

Facultad de Ingeniería.

Información de temas.

Número ID	Año del tema	Semestre del tema	Número del tema	Título
22	0	2	2	SISTEMA DE INFORMACION INTEGRAL DEL CENTRO COAPA

Número de alumnos en temas por método de búsqueda: 1

Figura 11.4. Información de temas de tesis.

#### **Función del botón *Buscar tema con director y codirector.***

Al hacer clic, se mostrará un formulario casi idéntico al que aparece en la figura 11.3 y el cual requerirá el ID del tema a buscar, en este caso se volverá a utilizar el ID: 22, contenido en la base de datos, como se muestra en la figura 11.5

La diferencia con la captura de pantalla correspondiente a la figura 11.4, es que en este caso muestra el nombre completo del director y codirector más no su clave de cada uno de ellos, como se muestra en la figura 11.6



#### **BÚSQUEDA DE TEMAS.**

Número ID del tema (aamnnn):

Búsqueda de temas.

Figura 11.5. Búsqueda de temas de tesis.

Universidad Nacional Autónoma de México.

Facultad de Ingeniería.

Información de temas.

Viernes inicia	Viernes termina	Sábado inicia	Sábado termina	Nombre	Apellidos
12:00:00	14:00:00			GERMÁN	SANTOS JAIMES
12:00:00	14:00:00			SC	SIN CODIRECTOR

Figura 11.6. Resultado de la búsqueda de temas de tesis, dando los nombres del director y codirector de tesis.

### **Función del botón *Buscar alumnos inscritos con profesores.***

Al hacer clic, se mostrará un formulario requiriendo la clave del profesor a buscar, en este caso se utilizó la clave 22, contenida en la base de datos, como se muestra en la figura 11.7

UNAM Universidad Nacional Autónoma de México Facultad de Ingeniería

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE.

BÚSQUEDA DE ALUMNOS INSCRITOS CON PROFESORES.

Clave del profesor: 22

Búsqueda de alumnos inscritos con profesores.

Figura 11.7. Buscar alumnos inscritos con profesores.

Y al ser ingresada la clave del profesor y posteriormente al hacer clic al botón: *búsqueda de temas*, mostrará todos los alumnos que el profesor, ha tenido inscritos en los seminarios de tesis, como se muestra en la figura 11.8

**Universidad Nacional Autónoma de México.  
Facultad de Ingeniería.  
Información de temas.**

Clave del profesor	ID Tema	Número de cuenta del alumno
22	97134	79589589
22	97134	79589589
22	97134	79589589
22	97134	79589589
22	97134	79589589
22	97134	79589589
22	932135	81161197
22	932135	81161197
22	99222	85003686
22	99222	85003686
22	99222	85003686
22	99222	85003686
22	99222	85003686
22	99222	85003686
22	99222	85003686
22	98292	85016790
22	98292	85016790
22	119	85181803
22	119	85181803
22	119	85181803
22	119	85181803
22	119	85181803
22	119	85181803
22	2193	85313264
22	2193	85313264
22	3175	85375840

Figura 11.8. Información de alumnos inscritos en temas del profesor.

**Función del botón *Buscar alumnos en temas.***

Al hacer clic, se muestra un formulario requiriendo el ID del tema tal y como se mostró en las figuras 11.3 y 11.5, en este caso se siguió utilizando el ID= 22, contenido en la base de datos.

Y al ser ingresada la clave del tema de tesis, y posteriormente al hacer clic al botón: *búsqueda de temas*, mostrará única y exclusivamente los alumnos inscritos en ese tema de tesis, en este caso se podrá apreciar que el tema de tesis con ID=22, tiene únicamente dos alumnos inscritos, como se muestra en la figura 11.9

**Universidad Nacional Autónoma de México.  
Facultad de Ingeniería.  
Información de temas.**

Año del tema	Semestre del tema	Número del tema	Título del tema	Número de cuenta del alumno	Clave de la carrera	Apellido
2000	2	2	SISTEMA DE INFORMACION INTEGRAL DEL CENTRO COAPA	90258314	110	OLIVA
2000	2	2	SISTEMA DE INFORMACION INTEGRAL DEL CENTRO COAPA	91329721	110	DAVA

Número de alumnos: 2

Figura 11.9. Resultado de los alumnos inscritos en temas.

**Función del botón *Buscar profesor vía apellido y nombre.***

Búsqueda de director o codirector por medio de su nombre completo o apellido, en caso de no saber su clave del mismo, como se aprecia en las figuras 11.10 y 11.11



The screenshot shows a dark blue header with the UNAM logo on the left and the text "Universidad Nacional Autónoma de México Facultad de Ingeniería" and "COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE." on the right. Below the header is a photograph of a modern building at night. Underneath the photo, the text "BÚSQUEDA DE PROFESORES POR APELLIDO Y NOMBRE." is centered. The search form consists of two input fields: "Apellidos del profesor:" and "Nombre(s) del profesor:". Below these fields is a button labeled "Búsqueda de profesor."

Figura 11.10. Búsqueda por apellido y nombre.



The screenshot shows a dark blue header with the UNAM logo on the left and the text "Universidad Nacional Autónoma de México Facultad de Ingeniería" and "COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL, DIE." on the right. Below the header is a photograph of a modern building at night. Underneath the photo, the text "BÚSQUEDA DE PROFESORES POR APELLIDOS." is centered. The search form consists of one input field: "Apellidos del profesor:". Below this field is a button labeled "Búsqueda de profesor."

Figura 11.11. Búsqueda por apellido.

**Función del botón *Ver temas de tesis registrados.***

Al dar clic se desplegará una lista la cual contendrá absolutamente todos los temas de tesis registrados y a su vez indicando el total de registros que se tienen, como se muestra en la figura 11.12

**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Información de temas.**

Número ID	Año del tema	Semestre del tema	Número del tema	Título
92119	92	1	19	CONOCIMIENTO DE LOS PROCEDIMIENTOS DE CALCULO, ESPECIFICACIONES Y DISEÑO DE LAS INSTALACIONES ELECTRI REGLAMENTACION OFICIAL .
92119	92	1	19	CONOCIMIENTO DE LOS PROCEDIMIENTOS DE CALCULO, ESPECIFICACIONES Y DISEÑO DE LAS INSTALACIONES ELECTRI REGLAMENTACION OFICIAL .
92119	92	1	19	CONOCIMIENTO DE LOS PROCEDIMIENTOS DE CALCULO, ESPECIFICACIONES Y DISEÑO DE LAS INSTALACIONES ELECTRI REGLAMENTACION OFICIAL .
95298	95	2	98	EVALUACION DE CONDICIONES DE OPERACION DEL EQUIPO ELECTRICICO DE POTENCIA.
952102	95	2	102	DISEÑO E IMPLANTACION DE UN SISTEMA DE CONTROL, MONITOREO Y REGISTRO DE UNA BANDA DE ESFUERZO PARA AF
952101	95	2	101	DISEÑO Y CONSTRUCCION DE UN TABLERO PARA EXPERIMENTACION EN ELECTRONICA DE POTENCIA.
952105	95	2	105	PROGRAMA PARA LA CAPTURA DE ECUACIONES EN FORMA GRAFICA.
952100	95	2	100	DISEÑO Y CONSTRUCCION DE UNA INTERFACE PARA LA ADQUISICION DE DATOS PARA UN SISTEMA DE DETECCION DE
95299	95	2	99	ESTUDIO DEL EFECTO DE LAS CORRIENTES ARMONICAS EN LAS MAQUINAS SINCRONAS
952115	95	2	115	DISEÑO Y CONSTRUCCION DE UN SISTEMA DE VIDEOPORTERO CON TECNOLOGIA DISPOSITIVO ACOPLADO POR CARGA (
952114	95	2	114	DISEÑO Y CONSTRUCCION DE UN INCLIONOMETRO BIDIRECCIONAL PARA MEDICION DE DEFORMACIONES EN PRESAS DI
952112	95	2	112	DISEÑO E INTEGRACION DE UN SISTEMA DE MEDICION DE RADIACION SOLAR CON UN RADIOMETRO DE BANDA GIRATOR
952124	95	2	124	HERRAMIENTAS FINANCIERAS PARA EVALUAR ESTRATEGIAS DE INVERSION.
952126	95	2	126	SISTEMA TUTORIAL DE APOYO DIDACTICO (PROGRAMACION DINAMICA)
952117	95	2	117	OPTIMIZACION DE PARAMETROS DE ENLACE PARA COMUNICACIONES VIA SATELITE.
952123	95	2	123	GENERADOR DE APLICACIONES DE MANTENIMIENTO EN AMBIENTECLIENTE SERVIDOR UTILIZANDO MICROSOFT VISUAL
952119	95	2	119	APLICACIONES DE REDES Y MULTIMEDIOS EN EL SECTOR SALUD.
952118	95	2	118	SISTEMA DE ADQUISICION Y CONTROL DE PROPOSITO GENERAL.
952130	95	2	130	LOS METODOS DE ANALISIS DE IMAGENES DIGITALES POR SEPARACION DE CONTORNOS Y TEXTURAS PARA CODIFICACI
952116	95	2	116	ENLACE DE UN SISTEMA DE ADQUISICION DE DATOS USANDO ETHERNET.
952128	95	2	128	CODIFICACION DE IMAGENES POR MEDIO DE FILTROS ESPEJO EN CUADRATURA Y CUANTIZACION VECTORIAL.
952127	95	2	127	ANALISIS Y CODIFICACION DE IMAGENES ASTRONOMICAS

Figura 11.12. Temas de tesis registrados.

# **CAPÍTULO V. CODIFICACIÓN.**

## Capítulo V. Codificación.

En esta etapa de desarrollo se llevó a cabo la traducción de la elaboración del diseño del sistema a código fuente, logrando como objetivo la facilidad de entendimiento del código para posteriores modificaciones facilitando así la depuración y la corrección de errores en caso de que se requieran.

En los anexos 2, 3, 4, 5, 6, 7, 8, 9 y 10 se muestran ejemplos del código del sistema, que se desarrollaron para la mayor parte de la fase de desarrollo, a continuación se muestran los anexos 2 y 3.

### V.I Anexo 2. Conexión a la base de datos.

En el siguiente código se muestra la conexión con la base de datos llamada ayumi, para el ejemplo se utilizó la tabla carrera cursada.

```
<center>
    <?php
error_reporting(E_PARSE);

$conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con el
servidor");
mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de datos");
$result=mysql_query("SELECT * FROM tbcarreracursada ORDER BY nocuentaz", $conexion);

    $num=mysql_num_rows($result);

    ?>
<h2>Universidad Nacional Autónoma de México.</h2>
<h2>Facultad de Ingeniería.</h2>
<h2>Información de carreras de la DIE.</h2>

<table border="1" width="50%">

    <tr>

<td bgcolor="#FF9999">Número de cuenta</td>
<td bgcolor="#FFFFCC">Clave de la carrera</td>
<td bgcolor="#FFFFCC">Generación</td>

    </tr>
<?php

    while($fila=mysql_fetch_array($result)) {

        echo "<tr>";

echo "<td bgcolor='#FF9999'>".$fila[nocuentaz]."</td>";
```



```

echo "<td bgcolor='#FFFFCC'>".$fila[clavecarrera]."</td>";
echo "<td bgcolor='#FFFFCC'>".$fila[generacion]."</td>";
echo "</tr>";

    }

    echo "</table>";
    echo "<hr /><h3>Número de registros: ".$num."</h3>";
    mysql_close($conexion);

?>

</center>

```

V.II Anexo 3. Insertar un registro nuevo en una tabla.

En el siguiente código se muestra como insertar un registro en una tabla, lo que en el sistema es dar una alta, para el ejemplo se utilizó la tabla programas de servicio social.

```

<?php
error_reporting(E_PARSE);

if($_POST) {

    $claveprog=$_POST["claveprog"];

    $nombreprog=$_POST["nombreprog"];

    $tipoprogram=$_POST["tipoprogram"];

    $responsprog=$_POST["responsprog"];

    $cargoresponsprog=$_POST["cargoresponsprog"];

    $telresponsprog=$_POST["telresponsprog"];

    if($claveprog=="" || $nombreprog=="" || $tipoprogram=="" || $responsprog=="" ||
    $cargoresponsprog=="" || $telresponsprog==""){

        die("Debe completar todos los campos que se le indicaron en las instrucciones.");
    }
}

```

```
}
```

```
    $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con  
el servidor");
```

```
    mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de  
datos");
```

```
        $result=mysql_query("INSERT INTO tbprogramass VALUES  
('.$claveprog.', ".$nombreprog.", ".$tipoprog.", ".$responsprog.", ".$cargoresponsprog.",  
".$stelresponsprog.)", $conexion);
```

```
        if($result) {  
            echo "Su inscripción ha sido registrada  
satisfactoriamente.";
```

```
        } else {
```

```
            $numErr=mysql_errno($conexion);
```

```
            $descErr=mysql_error($conexion);
```

```
            echo "No se pudo insertar el registro<br />";
```

```
            echo "Nº de error: ".$numErr." * Descripción: ".$descErr;
```

```
        }
```

```
    mysql_close($conexion);
```

```
}
```

```
?>
```

Los anexos 4, 5, 6, 7, 8, 9 y 10 están organizados de la siguiente manera:

Anexo 4. Insertar registros simultáneamente a dos tablas distintas en la base de datos.

Anexo 5. Eliminación de un registro de una tabla.

Anexo 6. Modificación de datos de un registro en una tabla.

Anexo 7. Consultar datos de un registro en una tabla.

Anexo 8. Combinar registros de dos o más tablas en una base de datos.

Anexo 9. Impresión de datos de un registro de una tabla.

Anexo 10. Mostrar todos los datos de los registros que se realizaron en una tabla.

**Capítulo VI.**  
**Descripción de la fase de pruebas.**

## Capítulo VI. Descripción de la fase de pruebas.

Las pruebas no son una fase aislada, sino constituyen en si un proyecto con su propio ciclo de vida, los beneficios de aplicar las técnicas de verificación y validación cumplen con el compromiso de plazo, costo y calidad, por consecuencia facilitan la transición de desarrollo a producción.<sup>43</sup>

Para conocer la importancia del proceso de pruebas es necesario conocer los siguientes conceptos:

- **Pruebas de calidad:** Es el nivel de fiabilidad robustez y eficiencia del software referido a todo su comportamiento en su periodo de vigencia. La facilidad de uso y mantenimiento son otros factores a tener en cuenta, así como la escalabilidad. Otro parámetro de calidad que también hay que tener presente es el cumplimiento de los requisitos especificados, la concordancia y el rendimiento explícitamente establecido, así como los estándares documentados y las características implícitas que se esperan del software.<sup>44</sup>
- **Metodología:** Conjunto de acciones y procedimiento realizados en el desarrollo de un software.
- **Entorno:** Es el conjunto de hardware y software sobre los que se efectúa una acción concreta.
- **Control de calidad:** Es el conjunto de procedimientos que se realizan para comprobar que se cumple la metodología de calidad del producto en parámetros aceptables para la entrega al cliente y su entorno de producción.
- **Revisiones de código:** Las más utilizadas son las revisiones técnicas formales cuyo objetivo principal es descubrir errores en la función, la lógica o la implementación de cualquier producto del software, verificar que satisface sus especificaciones, que se ajusta a los estándares establecidos, señalando las posibles desviaciones detectadas. Es un proceso de revisión riguroso, su objetivo es llegar a detectar lo antes posible, los posibles defectos o desviaciones en los productos que se van generando a lo largo del desarrollo. Esta característica fuerza a que se adopte esta práctica únicamente para productos que son de especial importancia, porque de otro modo podría frenar la marcha del proyecto.

---

<sup>43</sup> [enfoca.emagenia.info/enfoca.php?seccio=5&id=662](http://enfoca.emagenia.info/enfoca.php?seccio=5&id=662)

<sup>44</sup> <http://www.telefonica.es/sociedaddeinformacion/pdf/publicaciones/imagenio/capitulos/imageniocap8.pdf>

- **Fiabilidad del software:** La fiabilidad del software se define en términos estadísticos como la probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y durante un tiempo específico.
- **Fallo:** Es una falta de concordancia con los requisitos del software, el fallo puede ser asumible, como un error ortográfico, o puede ser catastrófico, como el fallo de un sistema de vuelo en tipo real, tales como el área de monitoreo de tráfico aéreo de un aeropuerto.
- **Verificación:** Es el conjunto de actividades que aseguran que el software implementa correctamente una función específica.<sup>45</sup>
- **Validación:** Conjunto de actividades que aseguran que el software construido se ajusta a los requisitos del cliente.

#### VI.I Pruebas al sistema.

Se realizaron distintos tipos de pruebas, para verificar la funcionalidad y así poderlo implementar en la Coordinación de Seminarios y Servicio Social de la DIE.

- **Pruebas unitarias:** Estas pruebas consistieron en verificar que el código fuente no trajera errores sintácticos y ortográficos, Dreamweaver cuenta con un revisor ortográfico y un depurador de código, los cuales al ser utilizados depuraron cada línea del código evitando errores de tipo sintáctico u ortográficos.
- **Pruebas integrales:** Su finalidad es comprobar que cada módulo sea capaz de funcionar interactuando con el resto de los módulos sin afectar el valor de los datos en las funciones.  
Se realizaron pruebas en línea, en donde el sistema se montó en red y a partir de distintas computadoras se ingresaron a los diferentes módulos del sistema y en ellos se realizaron altas, bajas, modificaciones, búsquedas, impresiones de documentos, etc., y se observó que todos los datos se guardaban correctamente.
- **Pruebas de volumen:** Las pruebas de volumen son necesarias ya que se puede identificar la forma en que se comporta un sistema si gradualmente la información aumenta, en la base de datos y en este sistema que desarrollamos se identificó si éste se comportaba estable cada vez que se le ingresaba información, como se muestra en la tabla 1.

---

<sup>45</sup> *Ibidem*

Proceso	Comportamiento del sistema	Prueba aprobada
Migración de 12859 registros, a la tabla tbreportseso, posteriormente se realizaron varios tipos de consulta.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.
Migración de 9309 registros, a la tabla tbalumnos, posteriormente se realizaron varios tipos de consulta.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.
Migración de 6205 registros, a la tabla tb serviciossocial, posteriormente se realizaron varios tipos de consulta.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.
Migración de 7127 registros, a la tabla tbsemitesis, posteriormente se realizaron varios tipos de consulta.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.
Registrar 40 profesores en la tabla tbprofesores, posteriormente se realizaron varios tipos de consulta.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.
Registrar 30 profesores más, distribuidos en todas las computadoras conectadas a la base de datos.	El sistema no presentó ningún cambio, se mantuvo estable.	Si.

Tabla 1. Comportamiento de pruebas.

- **Pruebas de usuario:** Este tipo de pruebas sirven para poder detectar las posibles fallas que una persona pueda generar ya que no está familiarizada con el sistema. Las partes principales de este tipo de pruebas son:
  - Responsable de la prueba.
  - Fecha y hora de la prueba.
  - El usuario quien realizará la prueba.
  - El puesto del usuario.

- Descripción de la prueba.
- Observaciones.
- Errores encontrados.
- Las correcciones de los errores encontrados.
- Firma del responsable de la prueba y del usuario quien realizó la prueba.

***A continuación se muestra un ejemplo de esta prueba.***



**Universidad Nacional Autónoma de México.**

**Facultad de Ingeniería.**

**Responsable:** Jesús Bucio Delgado.

**Fecha y hora:** 23 de febrero de 2012 11:00 a.m

**Usuario responsable:** Lic. Angélica Gutiérrez Vázquez

**Puesto:** Coordinadora de Seminarios y Servicio Social de la División de Ingeniería Eléctrica.

**Descripción de la prueba.**

Se deberán realizar los siguientes procesos:

- Registrar 3 dependencias de servicio social.
- Consultar los registros ingresados.
- Modificar el tipo de dependencia de alguno de los registros anteriores a "Paraestatal"
- Eliminar cualquiera de los registros ingresados

Observaciones: Ninguna.

**Error**

Sin error alguno.

**Corrección.**

-----

---

Usuario responsable

---

Responsable.

# **CAPÍTULO VII. IMPLANTACIÓN DEL SISTEMA.**

## Capítulo VII. Implantación del sistema.

En esta sección se mostrará el procedimiento para la implantación de este sistema. Como primer paso y dirigido al administrador del sistema, se debe tener en cuenta que la base de datos quedará centralizada, de esta manera se deberá elegir a la computadora que realizará la función de “Servidor”, entonces copiar el archivo “ayumi” a la carpeta “www”, situada en la unidad de almacenamiento “c”, cabe señalar que es necesario tener en cuenta los requerimientos de software y hardware.

### VII.I Requerimientos mínimos de software y hardware.

#### Requerimientos para computadora servidor

Dispositivo	Características mínimas (1)	Características recomendadas (2)
Procesador	Pentium III a 600 MHz	Pentium 4 a 1.5 GHz
Memoria	256 Mb	3 Gb.
Bus	PCI 2.2 a 66 MHz	PCI-X a 133 MHz
Disco Duro	80 Gb	100 Gb o mayor dependiendo de la cantidad de información que se desea manejar.
CD-ROM	CD-ROM 10 x	CD-ROM 24x
Interfaz gráfica y Monitor	Super VGA Color	Super VGA Color de 0.28.
Puertos	1 Serial, 1 Paralelo, 2 USB	2 Seriales, 2 Paralelo, 5 USB
Unidad de respaldo	de -	Unidad de respaldo acorde con la capacidad de almacenamiento instalada. Ejemplo: Iomega Zip 100. Iomega Jazz 2 Gb.
Tarjeta de red	Ethernet 10/100	Ethernet 10/100/1000.
Ratón	2 botones. Compatible con el estándar MS.	2 botones. Compatible con el estándar MS.

<b>Software básico</b>	<b>Características mínimas</b>	<b>Características recomendadas</b>
Sistema Operativo	Windows XP Service Pack 2	Windows XP Service Pack 3
Manejador de Base de Datos	MySQL, versión 5.2.36	MySQL, versión 5.5.20
Servidor http	Apache, versión 2.2.11	Apache, versión 2.4.1
Lenguaje de programación interpretado	PHP, versión 5.1.36	PHP, versión 5.3.8
Software para la edición y creación de contenidos we.	Adobe® Dreamweaver®CS4	Adobe® Dreamweaver® CS5.5

#### **Requerimientos para computadora cliente.**

<b>Dispositivo</b>	<b>Características mínimas (1)</b>	<b>Características recomendadas (2)</b>
Procesador	Pentium III a 500 MHz	Pentium III4 a 550 MHz
Memoria	128 Mb	256 Mb mínimo.
Bus	PCI 2.2 a 66 MHz	PCI 2.3
Disco Duro	990 MB libres	1 Gb.
CD-ROM	CD-ROM 10 x	CD-ROM 24x
Interfaz gráfica y Monitor	Super VGA Color	Super VGA Color de 0.28.
Puertos	1 Serial, 1 Paralelo, 1 USB	2 Seriales, 1 Paralelo, 2 USB
Unidad de respaldo	-	-
Tarjeta de red	Ethernet 10/100	Ethernet 10/100/.
Ratón	2 botones. Compatible con el estándar MS.	2 botones. Compatible con el estándar MS.

Software básico	Características mínimas	Características recomendadas
Sistema Operativo	Windows XP Service Pack 1	Windows XP Service Pack 2
Explorador de HTTP/HTML	Internet Explorer, versión 8.0	Internet Explorer, versión 9.0.5

Bajar la última versión del software “WampServer”, de la siguiente dirección web: <http://www.wampserver.com/en/> e instalarlo.

Posteriormente dirijirnos a la carpeta “www”, ubicada en la siguiente ruta: C:\wamp y arrastrar nuestro proyecto de nombre “ayumi” a esa carpeta. Como se muestra en la figura 1.

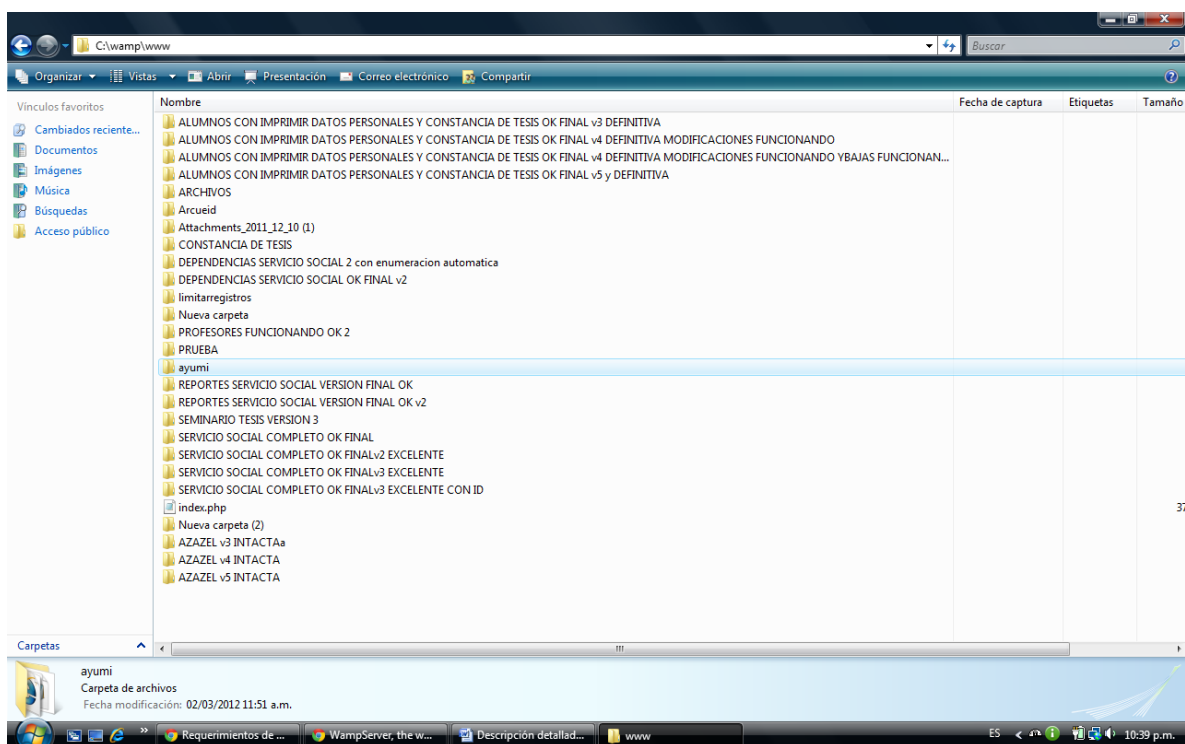


Figura 1. Ubicación de la carpeta “ayumi”

Se procedió a crear la base de datos y en ella crear las tablas correspondientes, para que finalmente se migraran los datos. En este ejemplo se utilizó la tabla “tbdependencias”, que tuvo la siguiente estructura:

```
CREATE TABLE IF NOT EXISTS `tbdependencias` (  
  `clavedepend` int(4) NOT NULL DEFAULT '0',  
  `nombredepen` varchar(800) NOT NULL,  
  `siglasdepend` varchar(10) NOT NULL,  
  `calledepend` varchar(50) NOT NULL,  
  `cpdepend` int(5) NOT NULL,  
  `coloniadepend` varchar(30) NOT NULL,  
  `delegdepend` varchar(25) NOT NULL,  
  `ciudadepend` varchar(20) NOT NULL,  
  `teldepend` int(15) NOT NULL,  
  `tipodepend` int(1) NOT NULL,  
  `fidepend` varchar(2) NOT NULL  
PRIMARY KEY (`clavedepend`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=933 ;
```

Y esta estructura se escribió en el editor de MySQL.

Como se vió en la figura 2.

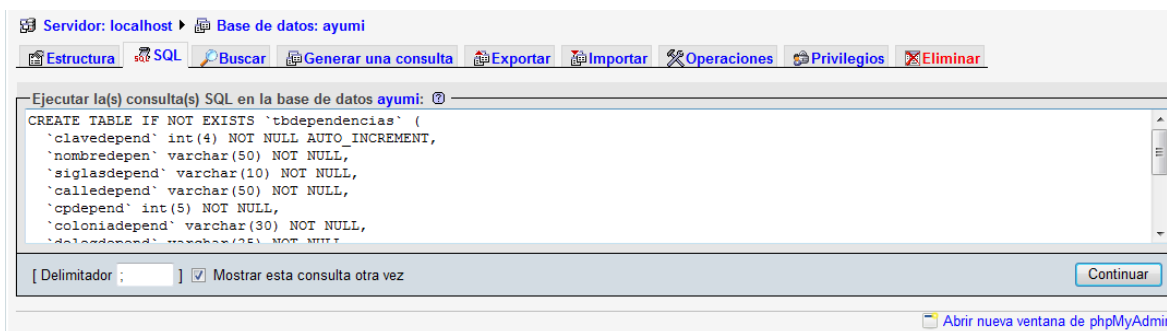


Figura 2. Estructura de la tabla tbdependencias en el editor de MySQL.

Para migrar los datos a la tabla “tbdependencias”, se introdujo en el editor de MySQL, un código como el siguiente:

```
INSERT INTO `tbdependencias` (`clavedepend`, `nombredepen`, `siglasdepend`, `calledepend`,
`cpdepend`, `coloniadepend`, `delegdepend`, `ciudadepend`, `teldepend`, `tipodepend`,
`fidepend`) VALUES
(1, 'DEPARTAMENTO DEL DISTRITO FEDERAL (DDF)', 'D.D.F.', 'DESCONOCIDO', 0, "", "", 0, 2, 'NO'),
(2, 'DIR. GRAL. DE SERVICIOS URBANOS DEL DDF', 'D.G.S.U.', 'AV. RIO CHURUBUSCO #1155', 0, "",
", 0, 2, 'NO'),
(3, 'DIR. GRAL. DE OBRAS PUBLICAS DE DDF', 'D.G.O.P.', "", 0, "", "", 0, 2, 'NO'),
(4, 'DIR. GRAL. DE GOBERNACION, DELEGACIONES POLITICAS', "", 0, "", "", 0, 2, 'NO'),
(5, 'DIR. GRAL. DE REORDENACION URBANA Y ECOLOGIA DEL D', "", 0, "", "", 0, 2, 'NO'),
(6, 'COMISION COORD. PARA EL DESARROLLO RURAL DEL DDF', "", 0, "", "", 0, 2, 'NO'),
(7, 'SERVICO DE TRANSPORTES ELECTRICOS DEL DF', 'S.T.E.', "", 0, "", "", 0, 2, 'NO'),
(8, 'COMISION DE VIALIDAD Y TRANSPORTE URBANO DEL DDF', "", 0, "", "", 0, 2, 'NO'),
(9, 'SRIA. DE SEGURIDAD PUBLICA', 'S.S.P.', "", 0, "", "", 0, 2, 'NO'),
(10, 'CONTRALORIA GRAL. DEL DDF', "", 0, "", "", 0, 2, 'NO'),
(11, 'PLANTA DE ASFALTO DEL DF', "", 0, "", "", 0, 2, 'NO'),
(12, 'SRIA. DEL MEDIO AMBIENTE DE DDF', 'SMA', "", 0, "", "", 0, 2, 'NO'),
(13, 'PRESIDENCIA DE LA REPUBLICA', "", 0, "", "", 0, 2, 'NO'),
(14, 'SUPREMA CORTE DE JUSTICIA DE LA NACION', "", 0, "", "", 0, 2, 'NO'),
(15, 'CAMARA DE DIPUTADOS', "", 0, "", "", 0, 2, 'NO'),
(16, 'PALACIO LEGISLATIVO', "", 0, "", "", 0, 2, 'NO'),
(17, 'PALACIO NACIONAL', "", 0, "", "", 0, 2, 'NO'),
(18, 'SRIA. DE SALUD', 'S.S.', "", 0, "", "", 0, 2, 'NO'),
(19, 'SRIA. DE RELACIONES EXTERIORES', 'S.R.E.', "", 0, "", "", 0, 2, 'NO'),
(20, 'CONTADURIA MAYOR DE HACIENDA', "", 0, "", "", 0, 2, 'NO');
```

Dando como resultado la figura 3.

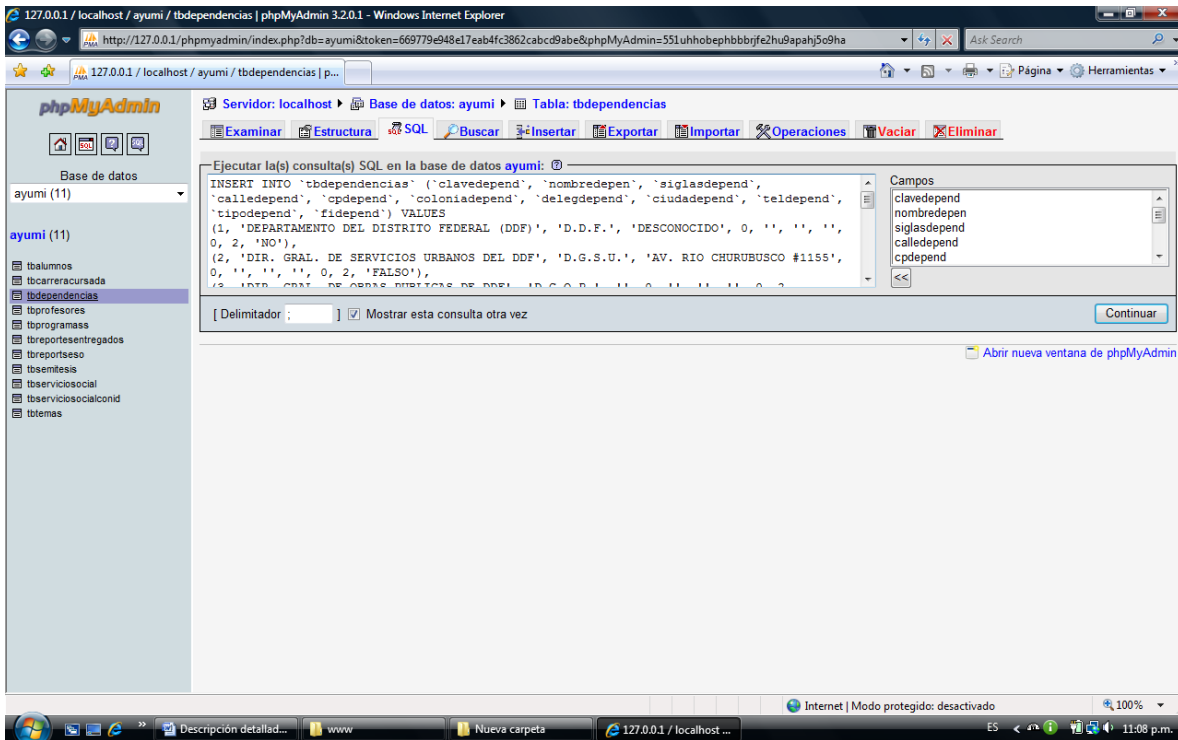


Figura 3. Migración de datos a la tabla de MySQL.

Los procedimientos anteriores se hicieron con cada una de las tablas, cabe señalar que cada una de ellas tuvo una estructura distinta.



Como siguiente paso, se crearon las tablas y en cada una de ellas se migraron los datos y se obtuvo como resultado la figura 4.

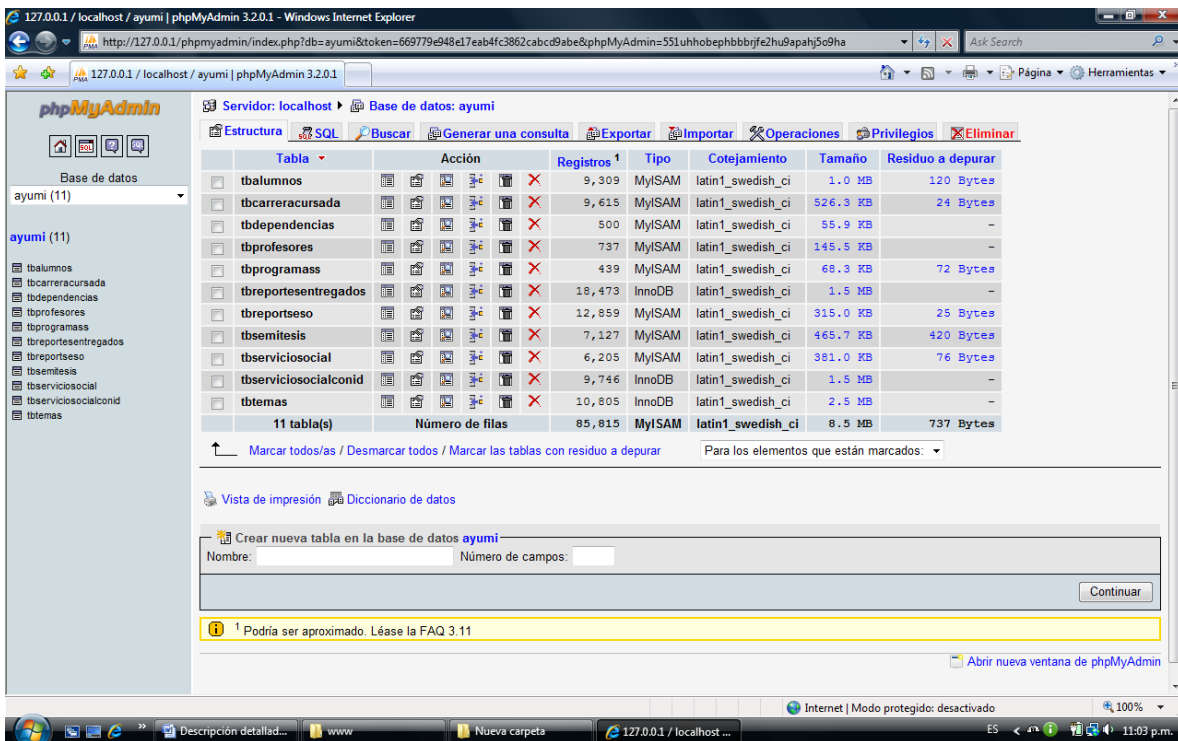


Figura 4. Tablas con datos migrados.

Finalmente para utilizar el sistema se siguieron los siguientes pasos:

- Iniciar WampServer, haciendo doble clic en su icono. Figura 5.
- Dar click al icono CSYSS\_DIE con el cual se inició el sistema Figura 6.
- Ingresamos a la interfaz del sistema y al dar click en el escudo de la Facultad de Ingeniería, nos pidió ingresar el nombre de usuario y contraseña para comenzar a usar el sistema Figura 7.
- Ingresó al menú del sistema. Figura 8.

- Iniciar WampServer.

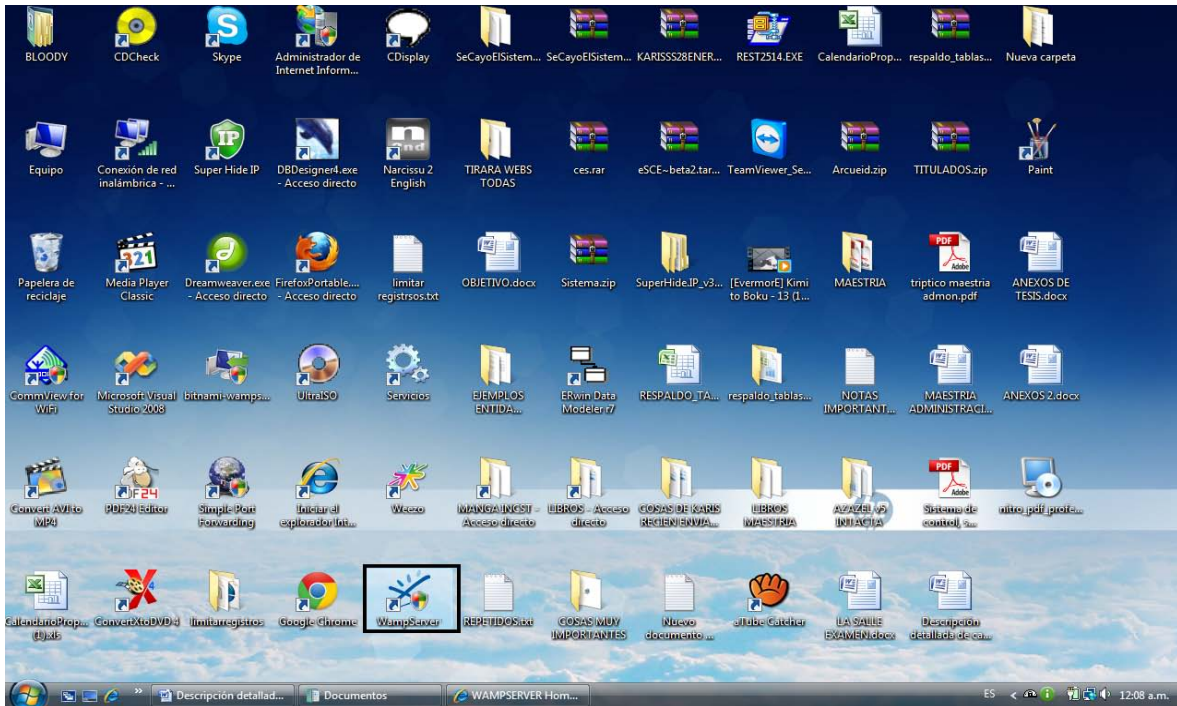


Figura 5. Ícono de Wampserver.

- Click en CSYSS\_DIE.html.

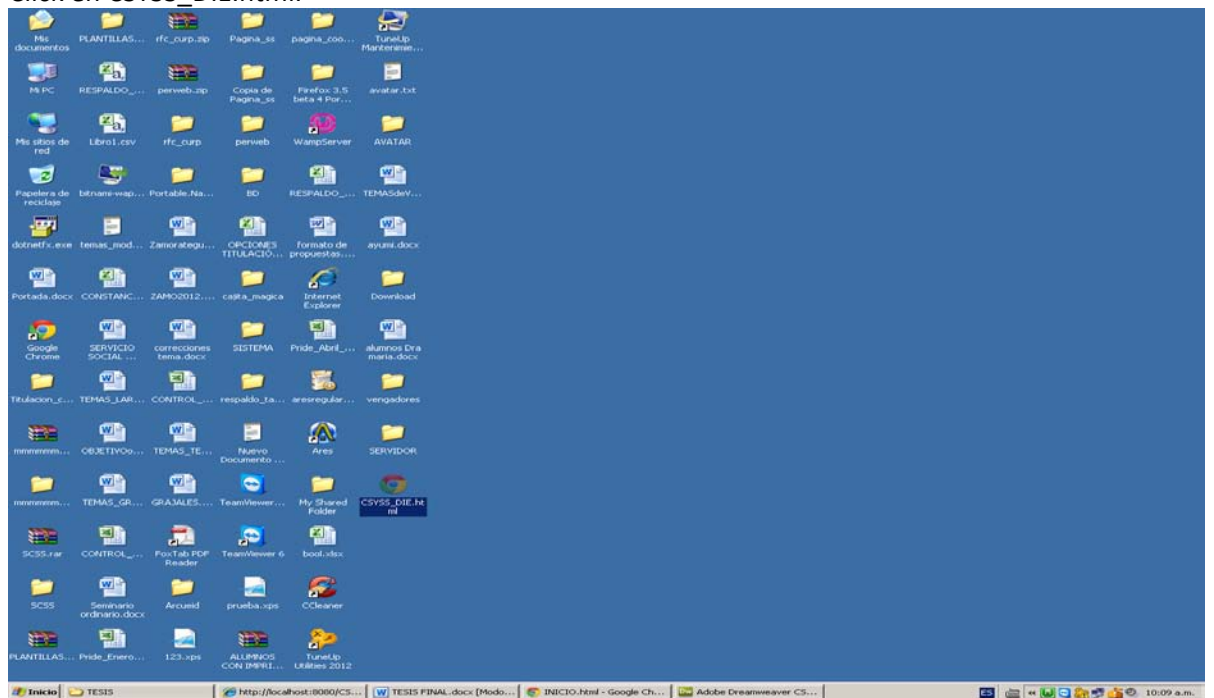


Figura 6. Ícono del sistema.

- Dar click en el icono de la Facultad de Ingeniería.



Figura 7. Ingreso al sistema.

- Ingresar el nombre de usuario y contraseña, e iniciar a usar el sistema como se ve a continuación.



Figura 8. Menú del sistema.

## **Conclusiones.**

El proyecto realizado contribuyó a agilizar los trámites en la Coordinación de Seminarios y Servicio Social de la DIE, haciendo énfasis de un manejo ordenado y responsable de la información, con esto se ven beneficiados todos los alumnos y personal que labora en la CSYSS de la DIE, cumpliendo el objetivo planteado en el inicio de la presente tesis.

Algunas pruebas realizadas fueron las siguientes; prueba de la funcionalidad, en esta prueba se corroboró que efectivamente el sistema cumplió con las características del diseño, prueba de la eficiencia en donde con previa capacitación del usuario fue posible observar cuanto tiempo le llevó realizar captura de información y consultas de la misma, prueba de usabilidad, esta prueba se realizó para observar las deficiencias que pudiera tener el sistema en cuanto a comprensión de la captura de los datos o de alguna otra petición como consultas o modificaciones en datos.

La metodología en cascada utilizada para el desarrollo de este software, fue la más adecuada porque se orienta al desarrollo de proyectos a corto plazo, definitivos y detallados además de que permitió una planificación sencilla y además dio como resultado un software de alta calidad, ya que cumplió satisfactoriamente con cada una de las pruebas realizadas, el software es adaptable a nuevos entornos de hardware, tiene una gran capacidad para soportar los cambios y sus características operativas son muy altas.

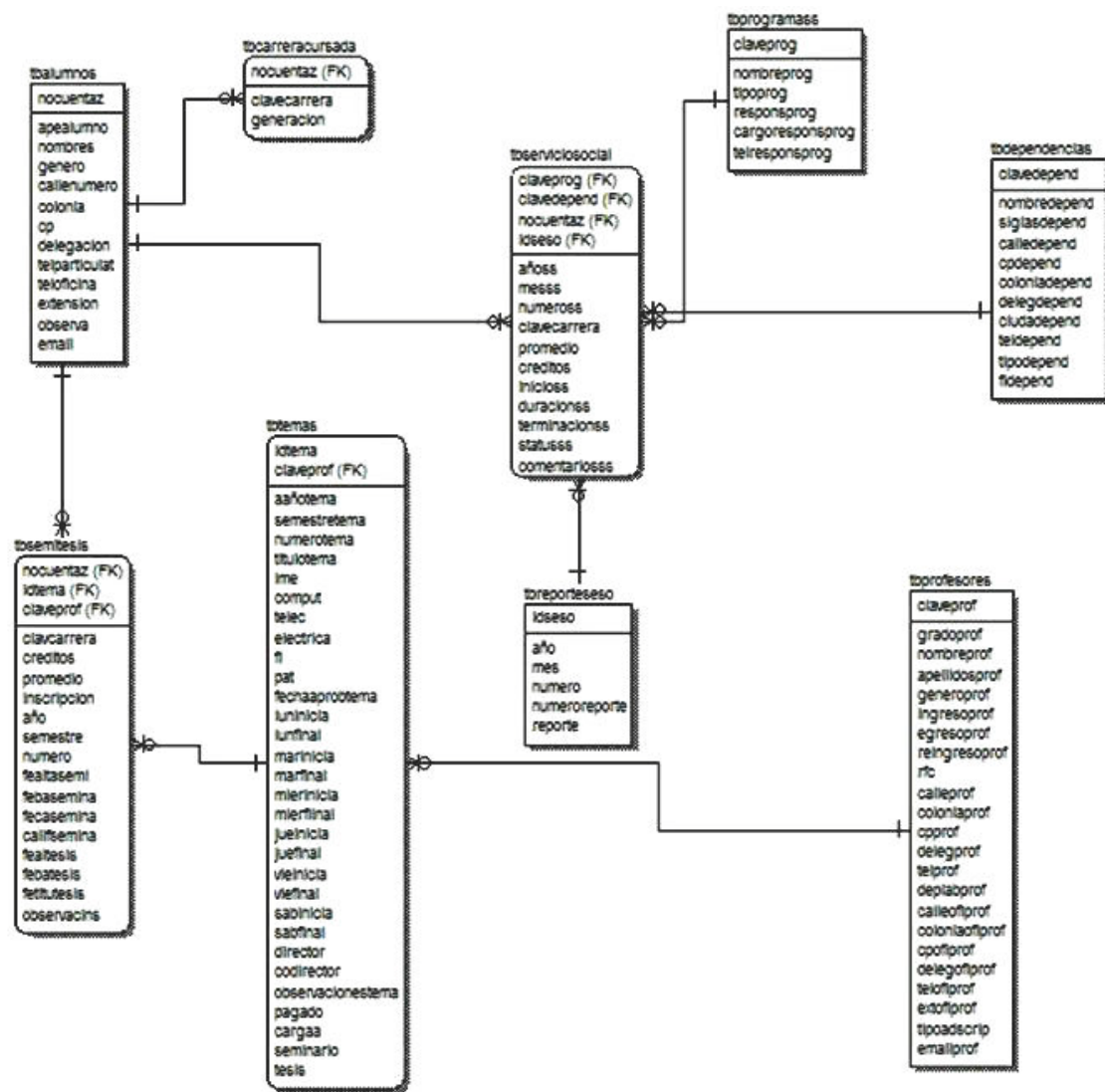
El sistema desarrollado, es de fácil instalación amigable, flexible y fácil de usar, dado que permite al usuario diferentes opciones como son: altas, bajas, cambios, modificaciones, impresiones de comprobantes, etc.

Para el desarrollo de este sistema se tuvieron que pasar por muchas etapas, algunas presentaron muchos problemas y otras no tanto, tal vez una etapa un poco problemática fue a la hora de migrar los datos, el cual fue un proceso muy laborioso y tardado debido a que se migró información de dos bases de datos diferentes.

También se trabajó para que el sistema fuera de gran portabilidad, que consumiera el mínimo de recursos del sistema, que fuera rápido, seguro, de fácil instalación y manipulación para los usuarios finales.

Los conocimientos adquiridos en la carrera y la aplicación de ellos, así como la investigación de otros fueron pieza fundamental para el desarrollo de este software y aunado con las experiencias con que se contaba al desarrollar software, facilitó completar este sistema.

**Anexo 1.**  
**Ejemplo del Diagrama.**  
**Modelo Entidad Relación.**



**Anexo 4.**  
**Ejemplo de código.**  
**Insertar registros simultáneamente a dos tablas  
distintas en la base de datos.**

### Insertar registros simultáneamente a dos tablas distintas en la base de datos.

El siguiente código inserta una alta en dos tablas diferentes como es el caso de tabla alumnos y carrera cursada.

```
<?php
```

```
error_reporting(E_PARSE);
```

```
if($_POST) {
```

```
    $nocuentaz=$_POST["nocuentaz"];
```

```
    $apealumno=$_POST["apealumno"];
```

```
    $nombres=$_POST["nombres"];
```

```
    $genero=$_POST["genero"];
```

```
    $callennumero=$_POST["callennumero"];
```

```
    $colonia=$_POST["colonia"];
```

```
    $cp=$_POST["cp"];
```

```
    $delegacion=$_POST["delegacion"];
```

```
    $telparticulat=$_POST["telparticulat"];
```

```
    $teloficina=$_POST["teloficina"];
```

```
    $extension=$_POST["extension"];
```

```
    $observa=$_POST["observa"];
```

```
    $email=$_POST["email"];
```

```
    $clavecarrera=$_POST["clavecarrera"];
```

```
    $generacion=$_POST["generacion"];
```



```
        if($nocuentaz==" || $apealumno==" || $nombres==" || $genero==" ||
$callennumero==" || $colonia==" || $cp==" || $delegacion==" || $clavecarrera==" ||
$generacion==""){
```

```
        die("Debe completar todos los campos que se le indicaron en las instrucciones.");
```

```
}
```

```
        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con
el servidor");
```

```
        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de
datos");
```

```
function execute($nocuentaz, $apealumno, $nombres, $genero, $callennumero, $colonia, $cp,
$delegacion, $telparticulat, $teloficina, $extension, $observa, $email, $clavecarrera, $generacion)
```

```
{
```

```
$q1= "INSERT INTO tbalumnos (nocuentaz, apealumno, nombres, genero, callennumero, colonia,
cp, delegacion, telparticulat, teloficina, extension, observa, email) values ('$nocuentaz',
'$apealumno', '$nombres', '$genero', '$callennumero', '$colonia', '$cp', '$delegacion',
'$telparticulat', '$teloficina', '$extension', '$observa', '$email)";
```

```
$resultquery = mysql_query($q1);
```

```
$errores = mysql_errno();
```

```
$q2= "INSERT INTO tbcarreracursada (nocuentaz, clavecarrera, generacion) values
('$nocuentaz', '$clavecarrera', '$generacion)";
```

```
$resultquery = mysql_query($q2);
```

```
$errores = $errores + mysql_errno();
```

```
return $errores;
```

```
}
```

```
$result = execute($nocuentaz, $apealumno, $nombres, $genero, $callennumero, $colonia, $cp,  
$delegacion, $telparticulat, $teloficina, $extension, $observa, $email, $clavecarrera, $generacion);
```

```
if ($result == 0) {
```

```
    echo "Su inscripción ha sido registrada satisfactoriamente.";
```

```
}
```

```
else
```

```
{
```

```
    echo "Error al insertar datos.";
```

```
}
```

```
        mysql_close($conexion);
```

```
    }
```

```
?>
```

**Anexo 5.**  
**Ejemplo de código.**  
**Eliminación de un registro de una tabla.**

## Eliminación de un registro de una tabla.

El siguiente código muestra la eliminación de un registro de la tabla alumnos.

```
<?php
    if($_POST) {
        $nocuentaz=$_POST["nocuentaz"];

        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con
el servidor");

        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de
datos");

        $result=mysql_query("DELETE tbalumnos, tbcarreracursada FROM
tbalumnos INNER JOIN tbcarreracursada ON tbalumnos.`nocuentaz` =
tbcarreracursada.`nocuentaz` WHERE tbalumnos.nocuentaz='$nocuentaz'") or die(mysql_error());

        if($result) {
            echo "REGISTRO ELIMINADO!";
        } else {
            $numErr=mysql_errno($conexion);
            $descErr=mysql_error($conexion);
            echo "No se pudo eliminar el registro<br />";
            echo "Nº de error: ".$numErr." * Descripción: ".$descErr;
        }

        mysql_close($conexion);
    }
?>
```

**Anexo 6.**  
**Ejemplo de código.**  
**Modificación de datos de un registro en una tabla.**

### Modificación de datos de un registro en una tabla.

El siguiente código muestra como se lleva a cabo la modificación de información en una tabla, para el ejemplo se utilizó la tabla alumnos.

```
<?php
```

```
    if($_POST) {  
        $nocuentaz=$_POST["nocuentaz"];  
        $apealumno=$_POST["apealumno"];  
        $nombres=$_POST["nombres"];  
        $genero=$_POST["genero"];  
        $callennumero=$_POST["callennumero"];  
        $colonia=$_POST["colonia"];  
        $cp=$_POST["cp"];  
        $delegacion=$_POST["delegacion"];  
        $telparticulat=$_POST["telparticulat"];  
        $teloficina=$_POST["teloficina"];  
        $extension=$_POST["extension"];  
        $observa=$_POST["observa"];  
        $email=$_POST["email"];  
        $clavecarrera=$_POST["clavecarrera"];  
        $generacion=$_POST["generacion"];  
  
        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con el  
servidor");  
  
        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de  
datos");
```

```

$result=mysql_query("UPDATE tbalumnos INNER JOIN tbcarreracursada ON
tbalumnos.`nocuentaz` = tbcarreracursada.`nocuentaz` SET tbalumnos.nocuentaz=".$nocuentaz.",
tbalumnos.apealumno=".$apealumno.", tbalumnos.nombres=".$nombres.",
tbalumnos.genero=".$genero.", tbalumnos.callennumero=".$callennumero.",
tbalumnos.colonia=".$colonia.", tbalumnos.cp=".$cp.", tbalumnos.delegacion=".$delegacion.",
tbalumnos.telparticulat=".$telparticulat.", tbalumnos.teloficina=".$teloficina.",
tbalumnos.extension=".$extension.", tbalumnos.observa=".$observa.",
tbalumnos.email=".$email.", tbcarreracursada.`clavecarrera`=".$clavecarrera.",
tbcarreracursada.`generacion`=".$generacion." WHERE tbalumnos.nocuentaz='$nocuentaz'") or
die(mysql_error());

```

```

if($result) {

```

```

    $numFilas=mysql_affected_rows($conexion);

```

```

    echo $numFilas." FILAS ACTUALIZADAS CORRECTAMENTE.";

```

```

} else {

```

```

    $numErr=mysql_errno($conexion);

```

```

    $descErr=mysql_error($conexion);

```

```

    echo "No se pudo actualizar el registro<br />";

```

```

    echo "Nº de error: ".$numErr." * Descripción: ".$descErr;

```

```

}

```

```

mysql_close($conexion);

```

```

}

```

```

?>

```

**Anexo 7.**  
**Ejemplo de código.**  
**Consultar datos de un registro en una tabla.**



### Consultar datos de un registro en una tabla.

En el siguiente código muestra como realizar una consulta de información de una tabla, en el ejemplo la tabla que se utilizó es la tabla alumnos.

```
<center>

<?php
error_reporting(E_PARSE);

    if($_POST) {

        $nocuentaz=$_POST["nocuentaz"];

        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con
el servidor");

        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de
datos");

        $result=mysql_query("SELECT tbalumnos.`nocuentaz` , tbalumnos.`apealumno` ,
tbalumnos.`nombres` , tbalumnos.`genero` , tbalumnos.`callennumero` , tbalumnos.`colonia` ,
tbalumnos.`cp` , tbalumnos.`delegacion` , tbalumnos.`telparticulat` , tbalumnos.`teloficina` ,
tbalumnos.`extension` , tbalumnos.`observa` , tbalumnos.`email` , tbcarreracursada.`clavecarrera`
, tbcarreracursada.`generacion`

FROM tbalumnos

INNER JOIN tbcarreracursada ON tbalumnos.`nocuentaz` = tbcarreracursada.`nocuentaz` WHERE
tbalumnos.nocuentaz='$nocuentaz'") or die(mysql_error());

        $num=mysql_num_rows($result);

    ?>

    <h2>Universidad Nacional Autónoma de México.</h2>

    <h2>Facultad de Ingeniería.</h2>

    <h2>Información de alumnos.</h2>
```

```

<table border="1" width="200%">
<tr>
<td bgcolor="#FF9999">Número de cuenta del alumno</td>
<td bgcolor="#FFFFCC">Apellidos paterno y materno:</td>
<td bgcolor="#FFFFCC">Nombre(s)</td>
<td bgcolor="#FFFFCC">Género</td>
<td bgcolor="#FFFFCC">Calle y número</td>
<td bgcolor="#CCFFFF">Colonia</td>
<td bgcolor="#CCFFFF">Cp</td>
<td bgcolor="#CCFFFF">Delegación o municipio</td>
<td bgcolor="#CCFFFF">Teléfono particular</td>
<td bgcolor="#FFCCCC">Teléfono oficina:</td>
<td bgcolor="#FFCCCC">Extensión</td>
<td bgcolor="#FFCCCC">Observaciones</td>
<td bgcolor="#FFCCCC">Email</td>
<td bgcolor="#FFCCCC">Clave de la carrera</td>
<td bgcolor="#FFCCCC">Generación</td>

</tr>

<?php

while($fila=mysql_fetch_array($result)) {

```

```

echo "<tr>";

    echo "<td bgcolor='#FF9999'>".$fila[nocuentaz]."</td>";
    echo "<td bgcolor='#FFFFCC'>".$fila[apealumno]."</td>";
    echo "<td bgcolor='#FFFFCC'>".$fila[nombres]."</td>";
    echo "<td bgcolor='#FFFFCC'>".$fila[genero]."</td>";
    echo "<td bgcolor='#FFFFCC'>".$fila[callennumero]."</td>";
    echo "<td bgcolor='#CCFFFF'>".$fila[colonia]."</td>";
    echo "<td bgcolor='#CCFFFF'>".$fila[cp]."</td>";
    echo "<td bgcolor='#CCFFFF'>".$fila[delegacion]."</td>";
    echo "<td bgcolor='#CCFFFF'>".$fila[telparticulat]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[teloficina]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[extension]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[observa]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[email]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[clavecarrera]."</td>";
    echo "<td bgcolor='#FFCCCC'>".$fila[generacion]."</td>";

    echo "</tr>";

}

}

echo "</table>";

echo "<hr /><h3>Número de alumnos por método de búsqueda: ".$num."</h3>";

mysql_close($conexion);

?>

```

</center>

**Anexo 8.**  
**Ejemplo de código.**  
**Combinar registros de dos o más tablas en una base  
de datos.**

### Combinar registros de dos o más tablas en una base de datos.

El siguiente código muestra cómo combinar registros de varias tablas, en el ejemplo las tablas utilizadas son alumnos, seminarios, temas y profesores.

```
<center>
```

```
<?php
```

```
error_reporting(E_PARSE);
```

```
    if($_POST) {
```

```
        $claveprof=$_POST["claveprof"];
```

```
        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con el servidor");
```

```
        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de datos");
```

```
        $result=mysql_query("SELECT claveprof, id, nocuentaz, apealumno, nombres, febasemina, febatesis, inscripcion
```

```
FROM tbalumnos
```

```
INNER JOIN tbsemitesis ON tbalumnos.nocuentaz = tbsemitesis.nocuenta
```

```
INNER JOIN tbtemas ON tbsemitesis.idtema = tbtemas.id
```

```
INNER JOIN tbprofesores ON tbtemas.director = tbprofesores.claveprof WHERE claveprof =".$_SERVER["REQUEST_URI"].$claveprof , $conexion);
```

```
        $num=mysql_num_rows($result);
```

```
    ?>
```

```
<h2>Universidad Nacional Autónoma de México.</h2>
```

```
<h2>Facultad de Ingeniería.</h2>
```

```
<h2>Información de temas.</h2>
```

```
<table border="1" width="300%">
```

```
<tr>
```

```
<td bgcolor="#FF9999">Clave del profesor</td>
```

```
<td bgcolor="#FFFFCC">ID del tema</td>
```

```
<td bgcolor="#FFCCFF">Número de cuenta del alumno</td>
```

```
<td bgcolor="#FFCCFF">Apellidos del alumno</td>
```

```
<td bgcolor="#FFCCFF">Nombre del alumno</td>
```

```
<td bgcolor="#FFCCFF">Fecha de baja de seminario</td>
```

```
<td bgcolor="#FFCCFF">Fecha de baja de tesis</td>
```

```
<td bgcolor="#FFCCFF">Inscripción</td>
```

```
</tr>
```

```
<?php
```

```
while($fila=mysql_fetch_array($result)) {
```

```
    echo "<tr>";
```

```
        echo "<td bgcolor='#FF9999'>".$fila[claveprof]."</td>";
```

```
        echo "<td bgcolor='#FFFFCC'>".$fila[id]."</td>";
```

```
        echo "<td bgcolor='#FFCCFF'>".$fila[nocuentaz]."</td>";
```

```
        echo "<td bgcolor='#FFCCFF'>".$fila[apealumno]."</td>";
```

```
        echo "<td bgcolor='#FFCCFF'>".$fila[nombres]."</td>";
```

```

        echo "<td bgcolor='#FFCCFF'>".$fila[febasemina]."</td>";
        echo "<td bgcolor='#FFCCFF'>".$fila[febatesis]."</td>";
        echo "<td bgcolor='#FFCCFF'>".$fila[inscripcion]."</td>";

        echo "</tr>";
    }
}

echo "</table>";

echo "<hr /><h3>Número de alumnos en temas por método de búsqueda:
".$num."</h3>";

mysql_close($conexion);

?>
</center>

```

**Anexo 9.**  
**Ejemplo de código.**  
**Impresión de datos de un registro de una tabla.**



### Impresión de datos de un registro de una tabla.

El siguiente código muestra la impresión de información contenida en un registro de la tabla, para el ejemplo se utilizó la tabla alumno.

```
<center>
```

```
    <?php
error_reporting(E_PARSE);

    if($_POST) {

        $nocuentaz=$_POST["nocuentaz"];

        $conexion=mysql_connect("localhost", "root", "") or die("No se pudo
conectar con el servidor");

        mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la
base de datos");

        $result=mysql_query("SELECT tbalumnos.`nocuentaz`,
tbalumnos.`apealumno`, tbalumnos.`nombres`, tbalumnos.`callennumero`, tbalumnos.`colonia`,
tbalumnos.`cp`, tbalumnos.`delegacion`, tbalumnos.`telparticulat`, tbalumnos.`teloficina`,
tbalumnos.`extension`, tbcarreracursada.`clavecarrera`, tbcarreracursada.`generacion`

FROM tbalumnos

INNER JOIN tbcarreracursada ON tbalumnos.`nocuentaz` = tbcarreracursada.`nocuentaz` WHERE
tbalumnos.nocuentaz='$nocuentaz'") or die(mysql_error());

        if($result) {

            while($fila=mysql_fetch_array($result)) {

                ?>
```

```

height="741">
                                <table border="1" bgcolor="#FFFFFF" width="838"
                                <tr>
                                <td width="982" height="735"
align="center" valign="middle">
                                <form action="ProcesaModif.php"
method="POST" target="inferior">
<h3>INFORMACI&Oacute;N DEL ALUMNO. </h3>
<h4 align="left">&nbsp;</h4>
<p align="left"><strong>Informaci&oacute;n personal. </strong></p>
<p align="left">
<input type="hidden" name="codigoid" value="<?php echo $fila[codigoid]; ?>" />
<br />
Número de cuenta:
<input type="text" name="nocuentaz" size="30" value="<?php echo $fila[nocuentaz]; ?>" />
</p>
<p align="left">Apellidos:
<input type="text" name="apealumno" size="30" value="<?php echo $fila[apealumno]; ?>" />
<br />
<br />

Nombre:
<input type="text" name="nombres" size="30" value="<?php echo $fila[nombres]; ?>" />
<br />
<br />

Calle y n&uacute;mero:

```

```
<input type="text" name="callnumero" size="30" value="<?php echo $fila[callnumero]; ?>" />
```

```
</p>
```

```
<p align="left">C&oacute;lonia:
```

```
    <input type="text" name="colonia" size="30" value="<?php echo $fila[colonia]; ?>" />
```

```
</p>
```

```
<p align="left">C&oacute;digo postal:
```

```
<input type="text" name="cp" size="30" value="<?php echo $fila[cp]; ?>" />
```

```
    <br />
```

```
    <br />
```

```
Delegaci&oacute;n:
```

```
    <input type="text" name="delegacion" size="30" value="<?php echo $fila[delegacion]; ?>" />
```

```
    <br />
```

```
    <br />
```

```
Tel&eacute;fono particular:
```

```
    <input type="text" name="telparticulat" size="30" value="<?php echo $fila[telparticulat]; ?>" />
```

```
    </p>
```

```
<p align="left"> Tel&eacute;fono de oficina:
```

```
<input type="text" name="teloficina" size="30" value="<?php echo $fila[teloficina]; ?>" />
```

```
</p>
```

```
<p align="left">Extensi&oacute;n:
```

```
    <input type="text" name="extension" size="30" value="<?php echo $fila[extension]; ?>" />
```

```
    </p>
```

```
<p align="left">&nbsp;  </p>
```

```

<p align="left">&nbsp;</p>
<p align="left"><strong>Carrera(s) Cursadas(s) </strong></p>
<p align="left">Carrera:
<input type="text" name="clavecarrera" size="30" value="<?php echo $fila[clavecarrera]; ?>" />
</p>
<p align="left">Generaci&oacute;n:
<input type="text" name="generacion" size="30" value="<?php echo $fila[generacion]; ?>" /></p>
</form>
</td>
</tr>
</table>
<br /><br />
<?php
}
}
mysql_close($conexion);
}
?>
</center>

```

**Anexo 10.**  
**Ejemplo de código.**  
**Mostrar todos los datos de los registros que se  
realizaron en una tabla.**

### Mostrar todos los datos de los registros que se realizaron en una tabla.

El siguiente código muestra todos los registros de información contenidos en una tabla, para el ejemplo se utilizó la tabla alumnos.

```
<center>
```

```
<?php
```

```
error_reporting(E_PARSE);
```

```
    $conexion=mysql_connect("localhost", "root", "") or die("No se pudo conectar con  
el servidor");
```

```
    mysql_select_db("ayumi", $conexion) or die("No se pudo conectar con la base de  
datos");
```

```
    $result=mysql_query("SELECT tbalumnos.`nocuentaz` , tbalumnos.`apealumno` ,  
tbalumnos.`nombres` , tbalumnos.`genero` , tbalumnos.`callennumero` , tbalumnos.`colonia` ,  
tbalumnos.`cp` , tbalumnos.`delegacion` , tbalumnos.`telparticulat` , tbalumnos.`teloficina` ,  
tbalumnos.`extension` , tbalumnos.`observa` , tbalumnos.`email` , tbcarreracursada.`clavecarrera`  
 , tbcarreracursada.`generacion`
```

```
FROM tbalumnos
```

```
INNER JOIN tbcarreracursada ON tbalumnos.`nocuentaz` = tbcarreracursada.`nocuentaz` ORDER  
BY nocuentaz", $conexion);
```

```
    $num=mysql_num_rows($result);
```

```
?>
```

```
<h2>Universidad Nacional Autónoma de México.</h2>
```

<h2>Facultad de Ingeniería.</h2>

<h2>Información de alumnos.</h2>

```
<table border="1" width="200%">
```

```
<tr>
```

```
<td bgcolor="#FF9999">Número de cuenta del alumno</td>
```

```
<td bgcolor="#FFFFCC">Apellidos paterno y materno:</td>
```

```
<td bgcolor="#FFFFCC">Nombre(s)</td>
```

```
<td bgcolor="#FFFFCC">Género</td>
```

```
<td bgcolor="#FFFFCC">Calle y número</td>
```

```
<td bgcolor="#CCFFFF">Colonia</td>
```

```
<td bgcolor="#CCFFFF">Cp</td>
```

```
<td bgcolor="#CCFFFF">Delegación o municipio</td>
```

```
<td bgcolor="#CCFFFF">Teléfono particular</td>
```

```
<td bgcolor="#FFCCCC">Teléfono de oficina:</td>
```

```
<td bgcolor="#FFCCCC">Extensión</td>
```

```
<td bgcolor="#FFCCCC">Observaciones</td>
```

```
<td bgcolor="#FFCCCC">Email</td>
```

```
<td bgcolor="#FFCCCC">Clave de la carrera</td>
```

```
<td bgcolor="#FFCCCC">Generación</td>
```

```
</tr>
```

```
<?php
```

```

while($fila=mysql_fetch_array($result)) {
    echo "<tr>";

        echo "<td bgcolor='#FF9999'>".$fila[nocuentaz]."</td>";
        echo "<td bgcolor='#FFFFCC'>".$fila[apealumno]."</td>";
        echo "<td bgcolor='#FFFFCC'>".$fila[nombres]."</td>";
        echo "<td bgcolor='#FFFFCC'>".$fila[genero]."</td>";
        echo "<td bgcolor='#FFFFCC'>".$fila[callennumero]."</td>";
        echo "<td bgcolor='#CCFFFF'>".$fila[colonia]."</td>";
        echo "<td bgcolor='#CCFFFF'>".$fila[cp]."</td>";
        echo "<td bgcolor='#CCFFFF'>".$fila[delegacion]."</td>";
        echo "<td bgcolor='#CCFFFF'>".$fila[telparticulat]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[teloficina]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[extension]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[observa]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[email]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[clavecarrera]."</td>";
        echo "<td bgcolor='#FFCCCC'>".$fila[generacion]."</td>";

    echo "</tr>";

}

echo "</table>";

echo "<hr /><h3>Número de alumnos por método de búsqueda: ".$num."</h3>";

```



```
mysql_close($conexion);
```

```
?>
```

```
</center>
```

## **BIBLIOGRAFÍA.**

Oktaba, H. (Octubre 2007). *Software Process Improvement: The Compesisoft Project*. (Vol. 40) Computer Society.

Salinas, A. (2007) *Obstáculos en la gestión de proyectos de tecnologías de información y comunicación*. Bucaramanga: ACIS.

Niazi, M., Wilson, D., & Zowghi, D. (2003). *A maturity model for the implementation of software process improvement: an empirical study*. *The Journal of Systems and Software* , 74 (2), 155--172.

DeMarco, T., & Boehm, B. (2002). *The Agile Methods Fray*. *IEEE Computer* , 31 (6), 90–92.

Boehm, B., & Turner, R. (May de 2004). Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods. *Proceedings of the 26th international Conference on Software Engineering* , 718--719.

Hernández, Fernández Baptista (1994). *Metodología de la Investigación*. Colombia: McGraw Hill.

Cohen, D. y Asín, E. *Sistemas de Información para los Negocios* (3ª. Edición). (pp. 3-174). México. McGraw-Hill.

Peña Ayala, A. (2006). *Ingeniería de Software: Una Guía para Crear Sistemas de Información*.

Medina Peralta, M. (2006-2008). *Sistema de Información de los Servicios de Atención Primaria*. Instituto Català de la Salut, *Revista de Innovación Sanitaria y Atención Integrada*: Vol. 1: Iss. 1, Article 7.

Wiley, J. *Introduction to Client / Server Systems: A Practical Guide for Systems Professionals*.

Tanenbaum, A. S. (2000). *Sistemas Operativos Distribuidos*. México: Prentice Hall Hispanoamericana, S.A.

Tanenbaum, A. S. (2000). *Redes de Computadoras*. México : Prentice Hall Hispanoamericana S. A.

## **Mesografía.**

Todas las ligas fueron revisadas por última vez el 23 de junio de 2012.

### **Aquitectura cliente-servidor**

<http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>

### **Bases de datos**

[http://es.wikipedia.org/wiki/Base\\_de\\_datos](http://es.wikipedia.org/wiki/Base_de_datos)

<http://www.monografias.com/trabajos34/base-de-datos/base-de-datos.shtml>

<http://www.basesdedatos.org/>

<http://es.kioskea.net/contents/bdd/bddintro.php3>

<http://aceproject.org/main/espanol/et/etg03.htm>

### **Apache**

[http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache)

### **MySQL**

<http://dev.mysql.com/doc/refman/5.0/es/features.html>

### **Wamp Server**

<http://www.brothersoft.es/download/47226.html>

### **Sistemas de información**

<http://www.gestiopolis.com/Canales4/mkt/simparalas.htm>

<http://www.telefonica.es/sociedaddelainformacion/pdf/publicaciones/imagenio/capitulos/imageniicap8.pdf>

### **Modelo de desarrollo por prototipos**

<http://rguerrero334.blogspot.es/1192897080/>

