



Benemérita Universidad Autónoma de Puebla



Facultad De Ciencias De La Computación

ADMINISTRACIÓN DE EXALUMNOS

TESIS PROFESIONAL

Que para obtener el Título de:
Licenciada en Ciencias De La Computación

Presenta:
Jessica Marcial Romero

Asesor:
M.C. Pedro Bello López

Puebla, Pue. Junio de 2009

AGRADECIMIENTOS

A mis Padres por el apoyo, que me han brindado, por los principios que me han inculcado y por darme la vida.

A mis hermanos por la paciencia, el cariño, la comprensión y el apoyo que he tenido de cada uno de ustedes.

A mi novio por nunca haber perdido la confianza en mí, por sus palabras de aliento, por su comprensión, ánimo y apoyo en todo momento, que lo he necesitado, por los grandes logros que hemos tenido juntos, sumemos uno más.

A mis amigos por aquellos buenos tiempos de estudiantes, por los que ahora tenemos, por los que vendrán y por la motivación de cada uno de ustedes, deseándome siempre una superación personal, por compartir los logros y lo realmente importante en nuestras vidas, tanto laboral como personal.

A mi asesor el M.C. Pedro Bello López por su paciencia, por el apoyo que me brindó para la realización de la tesis, porque aún cuando todos pensaban que no la terminaría, él me motivaba para continuar y concluir este documento, mostrándose no sólo como mi asesor sino como un amigo.

A "Dios" por todo lo logrado hasta el día de hoy.

ÍNDICE

INTRODUCCIÓN	V
ESTRUCTURA DEL DOCUMENTO	VII
Capítulo 1 . Marco Teórico	I
INGENIERÍA DE SOFTWARE	2
Ingeniería de Software: una definición	2
Evolución del Software ¿Cómo ha cambiado la Ingeniería de Software?	2
LENGUAJE UNIFICADO DE MODELADO (UML)	3
Visión general de UML	3
Vistas de UML	4
Vista Estática	5
Vista de los casos de uso	6
Vista de interacción	6
Diagrama de secuencia	6
Diagrama de colaboración	6
Vista de la máquina de estados	7
Vista de actividades	7
UML Basado en Ingeniería Web (UWE)	7
BASES DE DATOS	7
Niveles Externo, Conceptual e Interno	8
Proceso de Normalización	8
Datos y modelos de los Datos	9
Lenguajes de Bases de Datos	10
Bases de Datos Arquitectura Cliente – Servidor	11
Introducción a las bases de datos para web	12
MYSQL	12
Historia	12
PHP	13
Historia	13
Primeros comienzos de PHP	13
Evolución de PHP	13
APACHE	14
WAMP	14
Capítulo 2 . Análisis Del Sistema	15
Documento de requerimientos	16
Análisis textual del problema	16
Restricciones	17
Descripción de la información	17
Capítulo 3 . Diseño	24
Diagrama entidad – relación	26
Diccionario de datos	28
Modelo relacional	29
Normalización de la Base de Datos	30

Capítulo 4. Implementación	31
Implementación de la interfaz (PHP)	32
Página de Inicio	32
Interfaz de usuario “Exalumno”	36
Interfaz de usuario “Administrador”	41
Conclusiones	43
Perspectivas	43
Apéndice A: Instalación del servidor	44
Bibliografía	46

ÍNDICE DE FIGURAS

<i>Figura 1-1 Vistas y diagramas de UML.....</i>	<i>5</i>
<i>Figura 2-1 Diagrama de casos de uso general</i>	<i>17</i>
<i>Figura 2-2 Diagrama de casos de uso usuario</i>	<i>18</i>
<i>Figura 2-3 Diagrama de casos de uso exalumno.....</i>	<i>18</i>
<i>Figura 2-4 Diagrama de casos de uso administrador.....</i>	<i>19</i>
<i>Figura 3-1 Diagrama de actividad y estados.....</i>	<i>25</i>
<i>Figura 3-2 Diagrama de navegación.....</i>	<i>26</i>
<i>Figura 3-3 Diagrama entidad - relación.....</i>	<i>27</i>
<i>Figura 4-1 Página principal SIADEX.....</i>	<i>32</i>
<i>Figura 4-2 Objetivo.....</i>	<i>33</i>
<i>Figura 4-3 Beneficios.....</i>	<i>33</i>
<i>Figura 4-4 Búsquedas Carrera o Nombre.....</i>	<i>34</i>
<i>Figura 4-5 Búsquedas Generación.....</i>	<i>34</i>
<i>Figura 4-6 Resultados de búsqueda.....</i>	<i>34</i>
<i>Figura 4-7 Anuncios.....</i>	<i>35</i>
<i>Figura 4-8 Obtener una nueva cuenta</i>	<i>35</i>
<i>Figura 4-10 Olvide mi contraseña</i>	<i>36</i>
<i>Figura 4-9 Usuario registrado.....</i>	<i>36</i>
<i>Figura 4-12 Menú de exalumno</i>	<i>37</i>
<i>Figura 4-11 Sesión iniciada.....</i>	<i>37</i>
<i>Figura 4-13 Modificar exalumno</i>	<i>38</i>
<i>Figura 4-14 Agregar anuncio</i>	<i>38</i>
<i>Figura 4-15 Agregar anuncio</i>	<i>39</i>
<i>Figura 4-16 Modificar anuncio.....</i>	<i>40</i>
<i>Figura 4-17 Modificación anuncio.....</i>	<i>40</i>
<i>Figura 4-18 Eliminar anuncios.....</i>	<i>40</i>
<i>Figura 4-19 Administrador</i>	<i>41</i>
<i>Figura 4-20 Eliminar usuarios.....</i>	<i>41</i>
<i>Figura 4-21 Todos los anuncios.....</i>	<i>42</i>
<i>Figura 4-22 SIADEX.....</i>	<i>42</i>
<i>Figura 4-23 Icono de wamp</i>	<i>44</i>
<i>Figura 4-24 Pantalla de phpMyAdmin.....</i>	<i>44</i>
<i>Figura 4-25 Creación de tablas</i>	<i>45</i>
<i>Figura 4-26 Insertando campos en tablas.....</i>	<i>45</i>

INTRODUCCIÓN

Actualmente el sector empresarial reclama alumnos preparados con experiencia laboral. Sin embargo los egresados de computación se vinculan con las empresas hasta el momento de terminar sus créditos en la mayoría de los casos, lo que muchas veces impide que los estudiantes sean contratados debido a que carecen de experiencia laboral.

Dadas estas circunstancias, surge el planteamiento de generar un sistema web que proporcione información de las actividades de los egresados mediante una red de contacto de éstos con los alumnos activos para favorecer la vinculación estudiante-empresa esperando satisfacer la mayoría de las demandas actuales del sector empresarial.

Se tiene el objetivo de crear un espacio de interacción entre la Facultad de Ciencias de la Computación (FCC), sus alumnos y exalumnos para que estos fortalezcan sus relaciones profesionales y personales.

En donde se desarrolle un sistema de información basado en web que administre el seguimiento de los alumnos que han concluido sus créditos en las carreras de Licenciatura e Ingeniería en Ciencias de la Computación, así como alumnos de Maestría en Ciencias de la Computación y establezca un medio de vinculación entre alumnos activos y egresados. El sistema desarrollado incluye las siguientes características:

- Altas de usuarios, permiten acceder al sistema.
- Altas de anuncios, información general de congresos, conferencias y talleres.
- Altas de estadísticas, información gráfica de datos obtenidos.
- Eliminación de usuarios, cuentas inactivas.
- Eliminación de anuncios, con invalidez a la fecha actual.
- Eliminación de estadísticas.
- Modificaciones de usuarios, actualización del perfil.
- Modificaciones de anuncios, actualización de anuncios.
- Modificaciones de estadísticas, actualización de estadísticas.
- Consultas de anuncios, e información general.
- Búsquedas de exalumnos.

Nota: El sistema web "Administración de Exalumnos" se encuentra temporalmente alojado en la siguiente liga <http://mail.cs.buap.mx/~jessicam/SAE2/index.php>

JUSTIFICACIÓN

Los exalumnos forman parte medular de la comunidad universitaria ya que en el ejercicio de su profesión refrendan a diario el proyecto universitario: excelencia académica y compromiso social. Son conciencia crítica y a través de su experiencia, crean y recrean la ciencia y la cultura.

Con el fin de obtener una herramienta para fortalecer los vínculos sociales y profesionales de los exalumnos con la FCC y entre ellos, se desarrolla un sistema web con acceso a base de datos que servirá para que exista una relación entre alumnos activos, y exalumnos, mediante una página web que permitirá:

Funcionar vía web, acceder como usuario (común), exalumno o administrador.

Un usuario común: podrá registrarse para obtener una nueva cuenta y, hacer una consulta de la página en general.

Un exalumno: realizará consultas, tendrá una cuenta previamente registrada con la cual podrá acceder a la página y dar de alta anuncios o hacer modificaciones de su cuenta y de sus anuncios publicados.

Los tipos de anuncios que serán dados de alta o publicados son:

- Congreso
- Conferencia
- Taller

Un administrador tiene todos los privilegios del sistema es decir realizará registros, consultas, tendrá acceso al sistema, dar altas, modificaciones y eliminación tanto de cuentas como de anuncios, generará estadísticas y enviará información relevante a todos los usuarios registrados.

ESTADO DEL ARTE

Actualmente la Facultad de Ciencias de la Computación en su página www.cs.buap.mx apartado "información" cuenta con un enlace a egresados, donde refiere que son parte muy importante de la FCC e invitan a enviar un correo y escribir datos personales.

Uno de los grandes modelos para el diseño de la página fue basado en el sistema web exalumnos.com este sistema cuenta con una amplia gama de localización de exalumnos de toda la república mexicana, para ello un exalumno debe registrarse ingresando datos personales, obteniendo una cuenta, y este servicio es gratuito, permite hacer búsquedas de exalumnos aún cuando ellos no sean del mismo nivel educativo, escuela o estado. Éste sistema fue programado bajo el lenguaje de java y Microsoft Visual Studio.

En las páginas web de la mayoría de las escuelas existen enlaces o información para exalumnos, donde te piden registrarte e ingresar datos personales, por ejemplo <http://www.uanl.mx/exalumnos/> o http://exalumnos.itam.mx/exitam_bienvenida.php, estos sitios son con la finalidad de enviar información a aquellos que se encuentren inscritos en el sistema web, incluyen por lo general una bolsa de trabajo de empresas o lugares donde haya vacantes que sean acordes a su perfil académico, noticias relevantes o de interés a exalumnos en general.

ESTRUCTURA DEL DOCUMENTO

Capítulo 1. - En esta parte del documento se presenta un breve bosquejo del marco teórico que se utilizará para definir los conceptos primordiales con los cuales se trabajó para la realización del sistema web.

Capítulo 2. - Describe brevemente los requerimientos del problema y da una posible solución en el análisis de éste, mostrando los casos de uso básicos con los que cuenta el sistema.

Capítulo 3. - Se detalla la forma de implementación del sistema mostrando diversos diagramas así como la base de datos utilizada, las relaciones que existen entre los procesos, las reglas de normalización para determinar que la base de datos utilizada se encuentra de forma normalizada.

Capítulo 4. - La implementación descrita en esta sección se enfoca en un desarrollo gráfico con ayuda de herramientas de software que facilitan la elaboración de sistemas, tanto en programación y diseño, se describe mediante pantallas la funcionalidad del sistema.

Capítulo 5. - Se realizan las pruebas del sistema, realizadas sobre el funcionamiento del sistema web en la página donde se aloja.

Capítulo 1. Marco Teórico

En este capítulo se inicia por dar los conceptos teóricos referentes a la ingeniería de software y base de datos utilizados para la elaboración de este sistema de administración de exalumnos de la FCC.

INGENIERÍA DE SOFTWARE

Ingeniería de Software: una definición

Los ingenieros del software usan conocimiento de las computadoras y de la computación para ayudar a resolver problemas. A menudo el problema bajo análisis está relacionado con una computadora o con un sistema de computación existente; pero a veces las dificultades subyacentes del problema nada tienen que ver con las computadoras. Por consiguiente es esencial que en primer término se entienda la naturaleza del problema. En particular debemos ser muy cuidadosos para no imponer técnicas de computación en todo problema que surja. Primero debe resolverse el problema. Después, si resulta necesario, se puede usar la tecnología como herramienta para implementar nuestra solución [1].

El marco, que incluye un proceso, un conjunto de métodos y una serie de herramientas se llama ingeniería de software [6].

En la sociedad moderna el papel de la ingeniería es proporcionar sistemas y productos que mejoren los aspectos materiales de la vida humana, para que así la vida sea más fácil, segura y placentera [6].

La ingeniería del software es el establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente en máquinas reales [6].

Evolución del Software ¿Cómo ha cambiado la Ingeniería de Software?

En el desarrollo del software se pueden utilizar diversas metodologías, éstas son elegidas de acuerdo a los objetivos que se tienen dentro de un sistema, entre los modelos clásicos podemos citar los siguientes: incremental, espiral, cascada. Algunos de estos modelos permiten revisar si el sistema resultante satisface las necesidades del cliente, en caso contrario se habrá desperdiciado el tiempo y esfuerzo de todos los involucrados. Por esa razón es esencial que nuestras técnicas y herramientas de ingeniería del software se utilicen con la mira puesta en flexibilidad. Se debe reconocer que la mayoría de los sistemas no existen por sí mismos. En realidad, tienen interfaces con otros sistemas, ya sea porque reciben o proporcionan información.

El software de computadora evoluciona a través del tiempo, sin importar su dominio de aplicación, tamaño o complejidad. Manny Lehman ha analizado en forma detallada la industria del software y los sistemas en un esfuerzo dirigido a desarrollar una teoría unificada para la evolución del software. Las leyes subyacentes derivadas de su estudio son dignas de mencionarse.

La ley del cambio continuo (1974).- Los sistemas de tipo electrónico deben adaptarse en forma continua, de lo contrario se volverán menos satisfactorios a través del tiempo.

La ley de la complejidad creciente (1974).- Cuando un sistema de tipo electrónico está en evolución, su complejidad se incrementa a menos que se realice el trabajo necesario para mantenerla o reducirla.

La ley de la autorregulación (1974).- El proceso de evolución de un sistema de tipo electrónico se autorregula con la distribución del producto y las mediciones del proceso cercanas a la normal.

La ley de la conservación de la familiaridad (1980).- Cuando un sistema de tipo electrónico está en evolución y se quiere tener un desarrollo satisfactorio, todos los involucrados con el sistema, deben mantener el dominio sobre su contenido y comportamiento. El crecimiento excesivo disminuye ese dominio.

La ley del crecimiento continuo (1980).- El contenido funcional de los sistemas de tipo electrónico debe incrementarse en forma continua para mantener la satisfacción del usuario a lo largo del periodo de vida del sistema.

La ley de la calidad decreciente (1996).- La calidad de los sistemas de tipo electrónico parecerá declinar a menos que éstos se mantengan y adapten en forma rigurosa de acuerdo con los cambios en su ambiente operacional.

La ley del sistema de retroalimentación (1996).- Los procesos de evolución de los sistemas de tipo electrónico constituyen sistemas de retroalimentación con niveles, ciclos y agentes múltiples, y deben tratarse de forma que se obtengan mejorías significativas sobre cualquier base razonable [6].

En la actualidad debido al uso del internet y a la gran difusión con la que cuenta, existe lo que se denomina ingeniería de la web en la cual se hace referencia a las metodologías, técnicas y herramientas que se utilizan en el desarrollo de aplicaciones web en las que se apoya la evaluación, diseño, desarrollo, implementación y evolución de dichas aplicaciones [15].

El término de ingeniería de la web ha sido muy controversial, se podría pensar que se trata de un clon o un subconjunto de la ingeniería de software, sin embargo se incluyen nuevos enfoques, métodos, herramientas. El software a utilizar se clasificará por su complejidad, por ejemplo: utiliza un formulario, interactúa con una base de datos, se encuentra estático.

LENGUAJE UNIFICADO DE MODELADO (UML)

UML es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema de software. UML proporciona una forma estándar de describir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas, tales como los casos escritos en un lenguaje de programación específico, esquemas de bases de datos y componentes software reutilizables.

Las cinco vistas de una arquitectura son:

1. Vista de casos de uso (que muestre los requisitos del sistema).
2. Vista de diseño (que capture el vocabulario del espacio del problema y del espacio de la solución).
3. Vista de procesos (que modele la distribución de los procesos e hilos del sistema).
4. Vista de implementación (que se ocupe de la realización física del sistema).
5. Vista de distribución (que se centre en cuestiones de ingeniería del sistema) [9].

Visión general de UML

UML es un lenguaje para

- ❖ Visualizar.
- ❖ Especificar.

- ❖ Construir
- ❖ Documentar.

Los artefactos de un sistema con gran cantidad de software.

UML es un lenguaje

Un lenguaje proporciona un vocabulario y reglas para combinar palabras de ese vocabulario con el objetivo de posibilitar la comunicación. Un lenguaje de modelado es un lenguaje cuyo vocabulario y reglas se centran en la representación conceptual y física de un sistema. Un lenguaje de modelado como UML es por tanto un lenguaje estándar para planos de software.

UML es un lenguaje para visualizar

Para muchos programadores, la distancia entre pensar en una implementación y transformarla en código es casi cero. Lo piensas, lo codificas. De hecho, algunas cosas se modelan mejor directamente en código. El texto es un medio para escribir expresiones y algoritmos de forma concisa y directa.

UML es un lenguaje para especificar

Es este contexto, especificar significa construir modelos precisos, completos y no ambiguos. En particular, UML cubre la especificación de todas las decisiones de análisis, diseño e implementación que deben realizarse al desarrollar y desplegar un sistema con gran cantidad de software.

UML es un lenguaje para construir

UML no es un lenguaje de programación visual, pero sus modelos pueden conectarse de forma directa a gran variedad de lenguajes de programación. Las cosas que se expresan mejor gráficamente también se representan gráficamente en UML, mientras que las cosas que se expresan mejor textualmente se plasman con el lenguaje de programación.

UML es un lenguaje para documentar

Una organización de software que trabaje bien produce toda clase de artefactos además de código ejecutable. Estos artefactos incluyen (aunque no se limitan a):

1. Requerimientos.
2. Arquitectura.
3. Diseño.
4. Código fuente.
5. Planificación de proyectos.
6. Pruebas.
7. Prototipos.
8. Versiones.

Vistas de UML

No hay ninguna línea entre los diferentes conceptos y las construcciones en UML, pero, por conveniencia, nosotros los dividimos en varias vistas. Una vista es simplemente un subconjunto de UML que modela construcciones que representan un aspecto de un sistema. La división en diversas vistas es algo arbitraria, pero esperamos que sea intuitiva.

Área	Vista	Diagramas	Conceptos Principales
estructural	Vista estática	Diagrama de clases	Clase, asociación, generalización, dependencia, realización, interfaz.
	Vista de casos de uso	Diagrama de casos de uso	Caso de uso, actor, asociación, extensión, inclusión, generalización de casos de uso.
	Vista de implementación	Diagrama de componentes	Componente, interfaz, dependencia, realización.
	Vista de despliegue	Diagrama de despliegue	Nodo, componente, dependencia, localización
dinámica	Vista de máquina de estados	Diagrama de estados	Estado, evento, transición, acción
	Vista de actividad	Diagrama de actividad	Estado, actividad, transición de terminación, división, unión
	Vista de interacción	Diagrama de secuencia	Interacción, objeto, mensaje, activación
		Diagrama de colaboración	Colaboración, interacción, rol de colaboración, mensaje
Gestión del modelado	Vista de gestión del modelo	Diagrama de clases	Paquete, subsistema, modelo
Extensión de UML	Todas	Todos	Restricción, estereotipo, valores etiquetados

Figura 1-1 Vistas y diagramas de UML

Vista Estática

La vista estática modela los conceptos del dominio de la aplicación, así como los conceptos internos inventados como parte de la implementación de la aplicación. Las clases son el centro alrededor del cual se organiza la vista de clases la cual se define como un conjunto de construcciones en UML que representa un aspecto determinado del sistema mediante uno o varios diagramas bajo la notación UML. Una clase es la descripción de un concepto del dominio de la aplicación o de la solución de la aplicación. La visión estática se exhibe en los diagramas de clases, llamados así porque su objetivo principal es la descripción de clases [10].

Vista de los casos de uso

Un caso de uso describe una funcionalidad particular que se supone que el sistema debe realizar o exhibir, modelando el diálogo que un usuario, un sistema externo u otra entidad tendrán con el sistema a ser desarrollado. Cada caso de uso describe un posible escenario de la interacción de la entidad externa con el sistema. Para cada escenario, el caso de uso identifica todos los eventos que pueden ocurrir, así como las respuestas del sistema. Los casos de uso en su totalidad constituyen una descripción completa de todas las formas posibles de utilización del sistema por todas las posibles entidades. Puede decirse que la colección de casos de uso pinta un cuadro de la funcionalidad completa del sistema [1].

El diagrama de casos de uso representa también el sistema como una caja rectangular con el nombre en su interior. Los casos de uso están en el interior de la caja del sistema, y los actores fuera, y cada actor está unido a los casos de uso en los que participa mediante una línea [4].

Los casos de uso se pueden también describir en varios niveles a detalle. Se pueden sacar partes como factor común y ser descritos en términos de otros casos de uso más simples. Un caso del uso se implementa como una colaboración en la vista de interacción [10].

Vista de interacción

Describe secuencias de intercambios de mensajes entre los roles que implementan el comportamiento de un sistema. Un rol de clasificado, o simplemente "rol", es la descripción de un objeto, que desempeña un determinado papel dentro de una interacción, distinto de los otros objetos de la misma clase. Esta visión proporciona una vista integral del comportamiento del sistema. La vista de interacción se exhibe en dos diagramas centrados en distintos aspectos: diagramas de secuencia y diagramas de colaboración.

Diagrama de secuencia

Un diagrama de secuencia muestra un conjunto de mensajes, dispuestos en una secuencia temporal. Cada rol en la secuencia se muestra como una línea de vida, es decir, una línea vertical que representa el rol durante cierto plazo de tiempo, con la interacción completa. Los mensajes se muestran como flechas entre las líneas de vida.

Diagrama de colaboración

Un uso de un diagrama de colaboración es mostrar la implementación de una operación. La colaboración muestra los parámetros y las variables locales de la operación, así como asociaciones más permanentes. Cuando se implementa el comportamiento, la secuencia de los mensajes corresponde a la estructura de llamadas anidadas y el paso de señales del programa.

Tanto los diagramas de secuencia como los diagramas de colaboración muestran interacciones, pero acentúan aspectos diferentes. Un diagrama de secuencia muestra secuencias en el tiempo como dimensión geométrica, pero las relaciones entre roles son implícitas. Un diagrama de colaboración muestra las relaciones entre roles geométricamente, y relaciona los mensajes con las relaciones, pero las secuencias temporales están menos claras, porque vienen dadas por los números de secuencia.

Vista de la máquina de estados

Una máquina de estados modela las posibles historias de vida de un objeto de una clase. Una máquina de estados contiene los estados conectados por transiciones. Cada estado modela un período de tiempo, durante la vida de un objeto. Cuando ocurre un evento, se puede desencadenar una transición que lleve el objeto a un nuevo estado. Cuando se dispara una transición, se puede ejecutar una acción unida a la transición. Las máquinas de estados se pueden utilizar para describir interfaces de usuario, controladores de dispositivo, y otros subsistemas reactivos.

Vista de actividades

Un grafo de actividades es una variante de una máquina de estados, que muestra las actividades de computación implicadas en la ejecución de un cálculo. Un estado de actividad representa una actividad: un paso en el flujo de trabajo o la ejecución de una operación. Un grafo de actividades describe grupos secuenciales y concurrentes de actividades. Los grafos de actividades se muestran en diagramas de actividades [10].

UML Basado en Ingeniería Web (UWE)

Es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación.

Consiste en una notación y en un método.

- La notación se basa en UML para aplicaciones Web en general y para aplicaciones adaptativas en particular.
- El método consta de seis modelos:
 1. Modelo de casos de uso para capturar los requisitos del sistema.
 2. Modelo conceptual para el contenido (modelo del dominio).
 3. Modelo de usuario: modelo de navegación que incluye modelos estáticos y dinámicos.
 4. Modelo de estructura de presentación, modelo de flujo de presentación.
 5. Modelo abstracto de interfaz de usuario y modelo de ciclo de vida del objeto.
 6. Modelo de adaptación [15].

Cada etapa es en sí misma un proceso (o una colección de procesos) que pueden ser descrito como un conjunto de actividades. Y cada actividad comprende restricciones, salidas y recursos [1].

BASES DE DATOS

Concepto de Bases de Datos

A continuación se presentan algunas definiciones de bases de datos, según diversos autores.

Colección de datos interrelacionados almacenados en conjunto sin redundancias perjudiciales o innecesarias; su finalidad es servir a una aplicación o más, de la mejor manera posible; los datos se almacenan de modo que resulten independientes de los programas que los usan; se emplean métodos bien determinados para incluir nuevos datos y para modificar o extraer los datos almacenados [2].

Conjunto estructurado de datos registrados sobre soportes accesibles por ordenador para satisfacer simultáneamente a varios usuarios de forma selectiva y en tiempo oportuno, menciona Delobel, 1982 [2].

Colección o depósito de datos integrados, almacenados en soporte secundario (no volátil) y con redundancia controlada. Los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de ellos, y su definición (estructura de la base de datos) única y almacenada junto con los datos, se a de apoyar en un modelo de datos, el cual ha de permitir captar las interrelaciones y restricciones existentes en el mundo real. Los procedimientos de actualización y recuperación, comunes y bien determinados, facilitarán la seguridad del conjunto de los datos [2].

Un sistema de base de datos es básicamente un sistema computarizado para guardar registros; es decir, es un sistema computarizado cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información con base en peticiones. [3]

Un sistema de base de datos se puede dividir en tres niveles, los cuáles se describen a continuación:

Niveles Externo, Conceptual e Interno

- El nivel interno (también conocido como el nivel físico) es el que está más cerca del almacenamiento físico; es decir, es el que tiene que ver con la forma en que los datos están almacenados físicamente.
- El nivel externo (también conocido como el nivel lógico de usuario) es el más próximo a los usuarios; es decir, el que tiene que ver con la forma en que los usuarios individuales ven los datos.
- El nivel conceptual (también conocido como el nivel lógico de la comunicación, o en ocasiones sólo como el nivel lógico, sin calificar) es un nivel de indirección entro los otros dos [3].

Proceso de Normalización

Una base de datos relacional es aquella que sigue el llamado modelo relacional formulado inicialmente por E.F. Codd. El modelo relacional describe los datos, tablas, registros, campos y operadores como select, Project y join de un modo formal, matemático.

En una tecnología relacional, el proceso de organizar los campos en tablas se denomina normalización, porque con cada paso el diseño se aproxima cada vez más al estándar relacional (es decir, la norma relacional). Los expertos han propuesto muchos grados de normalización, pero todos parecen estar de acuerdo con los cinco primeros.

- **Primera forma normal.** Un campo dado de un registro dado sólo puede contener un valor. Esto prohíbe todo tipo de grupos que se repiten dentro de un único registro. Para implementar grupos que se repiten, diseñe una tabla con registros que se repiten.

- **Segunda forma normal.** Una tabla en primera forma normal está en segunda forma normal si y sólo si ninguno de sus atributos no-principales son funcionalmente dependientes en una parte de una clave candidata. Esta forma también especifica que dos o más tablas no pueden tener la misma clave principal. Si su diseño tiene dos o más tablas con la misma clave principal, debe combinar dichas tablas en una.
- **Tercera forma normal.** Ningún campo no clave depende de otro campo no clave.
- **Cuarta forma normal.** Esta prohíbe varias relaciones independientes de uno a varios entre los campos de la clave principal y los campos no clave.
- **Quinta forma normal.** Esta forma es muy extensa y suele ignorarse. Requiere descomponer una tabla en las partes más pequeñas posibles para eliminar toda la redundancia.

Como se puede observar, las formas normales son más reglas de verificación que procedimientos. Conseguir una forma normal generalmente es cuestión de descomponer tablas grandes en varias más pequeñas hasta que todos los datos de cada tabla estén asociados de manera clara y exclusiva con los demás datos de la misma tabla. Esto hace que los datos sean más fáciles de mantener y la aplicación más fácil de codificar. Siempre se puede emplear la operación de combinación para volver a ensamblar las tablas que ha separado en búsqueda de la normalización [5].

Datos y modelos de los Datos

La palabra datos se deriva del vocablo latín para "dar"; por lo tanto, los datos en realidad son hechos dados, a partir de los cuales es posible inferir hechos adicionales. Los sistemas relacionados están basados en una teoría formal denominada el modelo de datos relacional, de acuerdo con el cual:

- En tablas, los datos son representados por medio de filas y estas filas pueden interpretarse directamente como proposiciones verdaderas.
- Se proporcionan operadores para operar sobre las columnas de las tablas, y estos operadores soportan directamente el proceso de inferir proposiciones verdaderas adicionales a partir de las ya dadas.

Un modelo de datos se puede definir como un conjunto de conceptos, reglas y convenciones que nos permiten describir y manipular (consultar y actualizar) los datos de un cierto mundo real que deseamos almacenar en la base de datos [7].

Un modelo de datos es una definición lógica, independiente y abstracta de los objetos, operadores y demás que en conjunto constituyen la máquina abstracta con la que interactúan los usuarios. Los objetos nos permiten modelar la estructura de datos. Los operadores nos permiten modelar su comportamiento.

Entonces podemos distinguir al modelo de su implementación:

La implementación de determinado modelo de datos es una realización física, en una máquina real, de los componentes de la máquina abstracta que en conjunto constituyen ese modelo [3].

Algunos modelos con frecuencia utilizados en las bases de datos son:

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

- Base de datos de red: Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo: se permite que un mismo nodo tenga varios padres. Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos.
- Base de datos relacional: Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". 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. 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.
- Bases de datos multidimensionales: Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos multidimensional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.
- Bases de datos orientadas a objetos: Este modelo, trata de almacenar en la base de datos los objetos completos (estado y comportamiento). Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:
 - Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.
 - Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.
 - Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.
- Base de datos deductivas: Un sistema de base de datos deductivas, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. También las bases de datos deductivas son llamadas base de datos lógica, a raíz de que se basan en lógica matemática.

Lenguajes de Bases de Datos

Los modelos son la base para los lenguajes, aunque el nivel de abstracción de estos últimos es menor, ya que el lenguaje es el modelo más que una sintaxis. La existencia de distintos lenguajes puede proceder tanto del modelo como de la sintaxis; por ejemplo, el lenguaje SQL es el resultado de aplicar una determinada sintaxis al modelo relacional [7].

Todo modelo de datos tiene básicamente dos tipos de propiedades:

Estáticas: describen la estructura del Universo de Discurso y apenas varían en el tiempo. Se utiliza el Lenguaje de Definición de Datos (LDD) para describirlas.

Dinámicas: permiten el paso de un estado a otro, ejecutando operaciones sobre los datos almacenados en la estructura; se corresponden con la naturaleza evolutiva del mundo real. Para expresar las operaciones se utiliza el Lenguaje de Manipulación de Datos (LMD).

Los Sistema de Gestión de Bases de Datos (SGBD) se define como un conjunto coordinado de programas, procedimientos, lenguajes, etc. que suministra a los distintos tipos de usuarios los medios necesarios para describir y manipular los datos almacenados en la base, garantizando su seguridad.

Los SGBD no suelen ofrecer, los mecanismos suficientes para captar la semántica del mundo real, por lo que surgen modelos conceptuales, que facilitan la labor del diseñador ayudándole en su comunicación con el usuario [7].

Uno de los modelos de datos semánticos es el denominado Modelo Entidad/Interrelación (ME/R), propuesto por Peter P. Chen que como su nombre lo indica se basa en entidades (cualquier objeto de interés para el universo descrito) que se interrelacionan o asocian entre sí.

Bases de Datos Arquitectura Cliente – Servidor

Desde el punto de vista jerárquico, un sistema de base de datos puede ser visto como un sistema que tiene una estructura muy sencilla de dos partes, las cuales se conforman por un servidor (también denominado parte dorsal o servicios de fondo) y un conjunto de clientes (también llamados partes frontales, aplicaciones para el usuario o interfaces).

- El servidor es precisamente el propio SGBD. Soporta todas las funciones básicas del SGBD. En particular, proporciona todo el soporte de los niveles externo, conceptual e interno explicados. Por lo tanto, en este contexto "servidor" es sólo el nombre del sistema de gestión de datos.
- Los clientes son las diversas aplicaciones que se ejecutan sobre el SGBD, tanto aplicaciones escritas por el usuario como aplicaciones integradas [3].

El programa cliente pide lo que necesita al servidor, éste busca si cuenta con lo que le piden – el servidor puede contener programas computacionales diversos, información de bases de datos y otros- y luego transfiere al programa cliente la información pedida. El programa cliente organiza la información recibida y se la presenta al usuario final.

En el sistema cliente-servidor, los programas corren en ambas computadoras de modo interactivo y en tiempo real; por este motivo, si la computadora servidor no está funcionando, esta comunicación no está disponible.

Introducción a las bases de datos para web

La utilización de la World Wide Web para presentar y acumular datos se ha desarrollado mucho más allá de la sencilla presentación de páginas. Están muy lejanos los días en que los diseñadores web que necesitaban presentar los elementos de una colección creaban una página independiente para cada uno de ellos. Dichas páginas eran difíciles de organizar inicialmente y era casi imposible mantenerlas actualizadas con el paso del tiempo.

Cada vez más el volumen y la estructura de los datos que se presentan en la web permite su almacenamiento y organización en bases de datos, y después la generación de páginas web basadas en esas bases de datos. La gama de sitios que pueden beneficiarse de este enfoque es asombrosa. Cualquier sitio web (o parte del mismo) que presente información sobre un conjunto de elementos similares es candidato para la utilización de una base de datos web.

La solución general consiste en definir una base de datos, añadir un registro para cada elemento y después consultar dicha base de datos para generar páginas web sobre la marcha. Una página de menú en lenguaje de marcas de hipertexto (HTML) común se convierte en una consulta a una base de datos de varios registros. Una página subordinada se convierte en una presentación de detalle de un registro de la base de datos. Algunas ventajas son:

- La base de datos es mucho más fácil de mantener que todas las páginas web individuales.
- El empleo de una base de datos facilita la búsqueda de los elementos deseados.
- Una base de datos facilita la presentación de los mismos datos de distintas formas [5].

MYSQL

MySQL es un servidor de bases de datos distribuido bajo la licencia GPL. Las características más importantes de MySQL son las siguientes:

- Se trata de un sistema capaz de funcionar en diferentes plataformas, con resultados idénticos.
- Es multihilo, es capaz de trabajar con más de un procesador simultáneamente.
- Al ser código abierto, existen multitud de herramientas para trabajar con él, como clientes gráficos.
- Si se desea, es posible utilizar transacciones, lo que proporciona un nivel más de seguridad.
- Es posible utilizar diferentes lenguajes de programación [8].

MySQL es la base de datos elegida por la gran mayoría de programadores en PHP. Soporta el lenguaje SQL y la conexión de varios usuarios, pero en general se utiliza para aplicaciones de tamaño pequeño-medio [11].

Historia

MySQL comenzó con la intención de añadir a mSQL (otro sistema gestor de bases de datos) sus propias rutinas de bajo nivel para el acceso a tablas. Acabaron con la decepción de encontrar el resultado poco satisfactorio y lento para sus necesidades.

Fue entonces cuando tomaron la decisión de crear su propio conjunto de funciones de acceso a datos, manteniendo lo que fuese posible de la interfaz que utilizaba mSQL, lo que disminuía la dificultad del traspaso de tecnología y permitía utilizar software compatible con mSQL casi sin esfuerzo.

PHP

PHP es un lenguaje de alto nivel que necesita ejecutarse en el servidor, es decir, que se ejecute en el servidor donde están alojadas las páginas. La principal ventaja es que, al ejecutarse el código en el servidor, todas nuestras páginas van a poder ser vistas en cualquier computadora, independiente del navegador que tenga. Se trata de un lenguaje de programación gratuito [11].

PHP (acrónimo de PHP; Hypertext Pre-Processor) es un lenguaje de programación, relativamente nuevo, (su antecesor, PHP/FI), data de finales de 1994), concebido principalmente como herramienta para el desarrollo de aplicaciones Web.

Aunque existe una multitud de lenguajes y entornos de desarrollo concebidos para Internet, PHP se ha convertido en uno de los lenguajes, del lado servidor, más ampliamente utilizados para el desarrollo de páginas dinámicas junto con ASP, JSP, ColdFusion y Perl. En los últimos años, el número de servidores que utilizan PHP se ha disparado. De hecho, según datos de Netcraft (<http://www.netcraft.com/>) a fecha de Agosto de 2004 son casi 17 millones de dominios los que usan PHP [11].

Historia

Primeros comienzos de PHP

PHP comenzó siendo un conjunto de scripts escritos en Perl que permitían a su creador, Rasmus Lerdorf, el control de los accesos a sus páginas personales. A este conjunto de scripts les denominó como Personal Home Page Tools. Poco a poco, Ramus fue completando las funcionalidades básicas de su herramienta escribiendo programas en C. En 1995 decidió liberar el código fuente escrito en C para que cualquiera pudiera utilizarlo, e incluso, colaborar en su mejora. De este modo nació PHP/FI. A finales de 1997 se libera PHP/FI 2.0, pasando de ser el proyecto de una sola persona al desarrollo de un equipo. Tuvo un seguimiento estimado de varios miles de usuarios en todo el mundo, con aproximadamente 50.000 dominios informando que lo tenían instalado, lo que sumaba alrededor del 1% de los dominios de Internet.

En junio de 1998 se liberó oficialmente PHP 3.0, anunciando como sucesor oficial de PHP/FI 2.0, si bien había sido completamente reescrito por Andi Gutmans y Zeev Zuraski. Una de las mejores características de PHP 3.0 que atrajo a docenas de desarrolladores a unirse y enviar nuevos módulos de extensión era su gran extensibilidad, además de proveer a los usuarios finales de una sólida infraestructura para bases de datos, protocolos y APIs. En su apogeo, PHP 3.0 estaba instalado en aproximadamente un 10% de los servidores web en internet.

Evolución de PHP

El siguiente paso en la evolución de PHP consistió en la reescritura de su núcleo, dando lugar a un nuevo motor denominado Zend (acrónimo de los apellidos Zeev y Andi). PHP 4.0, basado en este motor, y acoplado con un gran rango de nuevas características adicionales, fue oficialmente liberado en mayo de 2000. Además de la mejora de ejecución de esta versión, PHP 4.0 incluía otras características clave, como el soporte para la mayoría de los servidores Web, sesiones

HTTP de forma nativa, ciertas facilidades de orientación a objetos, compatibilidad con las expresiones regulares de Perl, buffers de salida, encriptación, formas más seguras de controlar las entradas de usuario y muchas nuevas construcciones de lenguaje, etc.

La última y actual versión de PHP, liberada en Julio de 2004, es la 5.0. Está basada en el nuevo motor Zend 2, el cual ha vuelto a ser reescrito por completo. Entre sus características y novedades más resaltables está el completo soporte para la programación orientada a objetos. También incorpora la gestión de excepciones, una nueva biblioteca de XML, soporte nativo para el sistema gestor de base de datos SQLite, y mejoras en la gestión de las cadenas de caracteres. PHP 5.0 soporta también MySQLi, una nueva ampliación de MySQL, la cual, además de la interfaz habitual, encierra una interfaz basada en objetos [12].

APACHE

Apache es un servidor de web bajo el esquema de Open Source que tiene soporte para una gran variedad de plataformas.

El servidor opera como uno normal y contiene las siguientes extensiones:

- Autenticación basada en Cookies
- Soporte para definir de UID y GID al ejecutarse scripts CGI
- Autenticación de usuarios usando el sistema de BD de cuentas de UNIX
- Módulos mejorados en el lado del servidor: Server Side Incluye (SSI)
- Intérprete incrustado de Perl
- Autenticación basada en Kerberos
- Traductores de conjuntos de caracteres
- Módulos de autenticación de usuarios para BD Postgres95 y mSQL o MySQL
- Contadores de páginas web
- Formatos de registro extendidos (logs)
- Autenticación de Sistema de Información de Red (NIS)
- Módulos de reescritura de URI/URL
- Procesador de scripts para Tcl
- Diferentes alternativas para módulos CGI [13].

WAMP

WAMP (Windows-Apache-MySQL-PHP/Python/PERL) es el acrónimo usado para describir un sistema de infraestructura de Internet que usa las siguientes herramientas:

- Windows, el sistema operativo;
- Apache, el servidor web;
- MySQL, el gestor de bases de datos;
- PHP, Python, o Perl, lenguajes de programación.

El uso de WAMP permite servir páginas html o internet, además de poder gestionar datos en ella, utilizaremos WAMP Server para ejecutar nuestra aplicación para evitar instalar PHP, MySQL y el servidor de Apache por separado.

Capítulo 2. Análisis Del Sistema

En este capítulo se documenta todo el análisis del sistema. Los diagramas que se ilustran están basados en notación de UML, con los cuales se obtienen los requerimientos para la funcionalidad del sistema, incluyendo la base de datos utilizada.

Documento de requerimientos

Para reducir la ambigüedad en el análisis textual los requerimientos son expresados por medio de oraciones que empiezan con "El sistema debe", "El sistema debe requerir", "El sistema debe permitir". Los requerimientos se expresarán mediante R1, R2,..., RN como simbología utilizada para este documento.

Análisis textual del problema

R1. El sistema debe requerir el llenado de un formulario con información personal para obtener una nueva cuenta de usuario.

R2. El sistema debe requerir que los usuarios registrados se identifiquen mediante un usuario y una contraseña, que serán activadas con la información que se proporcionó en el momento de registrarse, de esta forma se habilitan características diferentes para un exalumno y un administrador.

R3. El sistema debe permitir que los usuarios registrados escojan su usuario y contraseña con la que ingresarán al sistema.

R4. El sistema debe identificar a un administrador por medio de su tipo de dato el cual es un número exacto que utiliza datos enteros.

R5. El sistema debe identificar a los exalumnos por medio de su matrícula, la cual es de tipo numérico entero y tiene una longitud de 9 caracteres.

R6. El exalumno debe dar de alta anuncios, un anuncio se define por su tipo correspondiente: congreso, conferencia, taller.

R7. El sistema debe permitir al exalumno, modificar o eliminar anuncios que pertenezcan al exalumno que lo dio de alta.

R8. El sistema debe permitir al exalumno, modificar sus datos personales de su cuenta con la que se encuentra registrado.

R9. El sistema debe permitir visualizar las opciones de ¿qué es exalumno?, los anuncios clasificados, y realizar la búsqueda de exalumno a todo tipo de usuario, es decir, usuario (no registrado), exalumno, administrador.

R10. El administrador puede enviar información a los correos de los usuarios registrados.

R11. El administrador puede publicar estadísticas.

R12. El sistema debe permitir el alta de un número indeterminado de usuarios.

R13. El sistema debe permitir todo tipo de control del sistema por parte del administrador, es decir, altas, modificaciones y eliminaciones tanto de exalumnos como de anuncios publicados por cualquiera de ellos.

Restricciones

Las restricciones para este proyecto son las siguientes:

- El administrador tendrá todos los permisos de acceso al sistema tanto para modificar la información de la base de datos, como la interfaz o estructura de la página web, hacer el mantenimiento pertinente (altas, eliminaciones y modificaciones), actualizaciones del sitio, enviar información a todos los usuarios que tengan una cuenta en el sistema y generar las estadísticas.
- El exalumno (deberá estar previamente registrado para poder acceder) solo podrá hacer modificaciones en su perfil de cuenta, pero en ningún otro sitio; agregará información y modificará la misma si es que lo requiere o eliminará lo que haya publicado y no podrá acceder a cuentas que no le pertenecen.
- Un usuario común solo podrá hacer consultas a la base de datos a través de la página, pero sin hacer modificación alguna.

Descripción de la información

Utilizaremos el término SIADEX como forma de abreviación del sistema de administración de exalumnos.

La figura 2-1 representa los casos de uso del sistema SIADEX, donde se especifican tres actores cada uno con sus respectivas funciones de casos de uso.

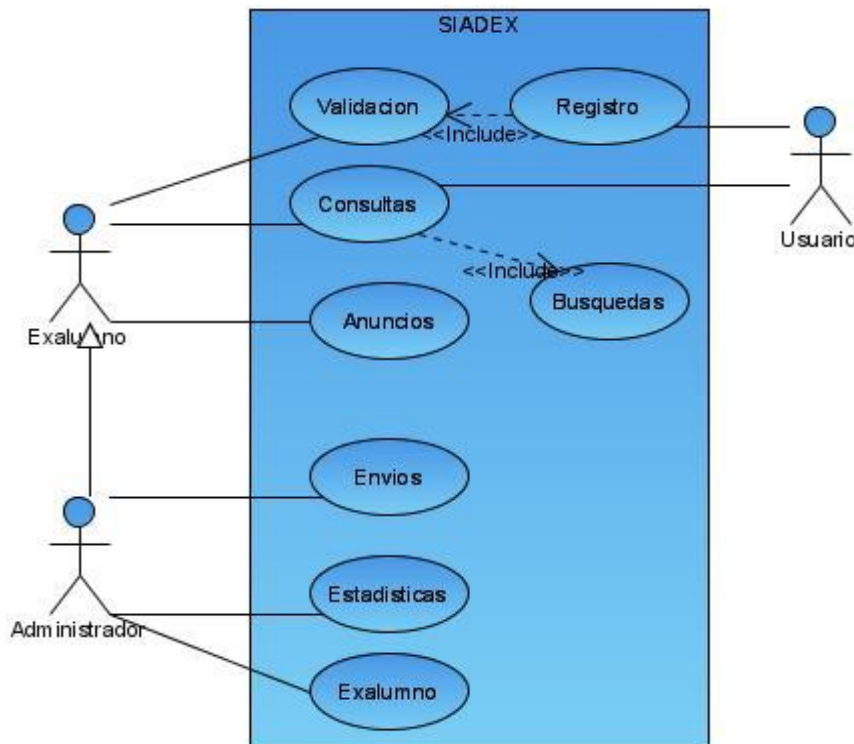


Figura 2-1 Diagrama de casos de uso general

El usuario no registrado por llamar a todo aquel que interactúe con el sistema únicamente tendrá los casos de uso que se mencionan en la figura 2-2.

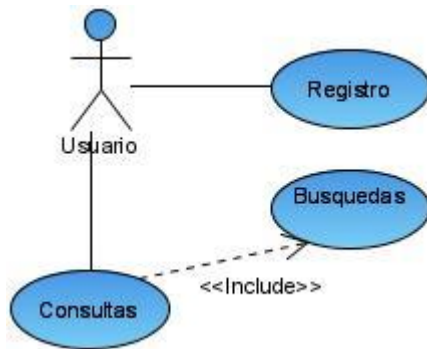


Figura 2-2 Diagrama de casos de uso usuario

Los casos de uso de un exalumno se muestran en la figura 2-3.

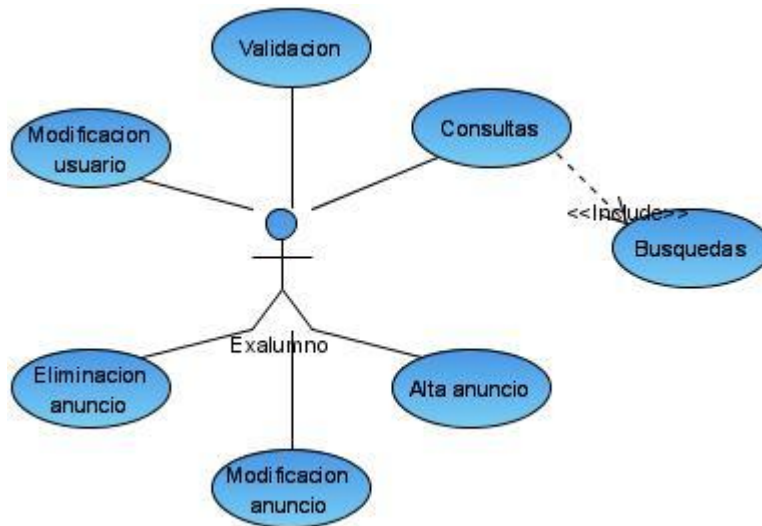


Figura 2-3 Diagrama de casos de uso exalumno

Se describirán los casos de uso del administrador que es el usuario con mayores privilegios para trabajar con la base de datos y el control del sistema, en el diagrama de casos de uso general se identifica que el administrador es una generalización del exalumno, es decir, hereda todas las funciones de un exalumno, la representación se muestra en la figura 2-4.

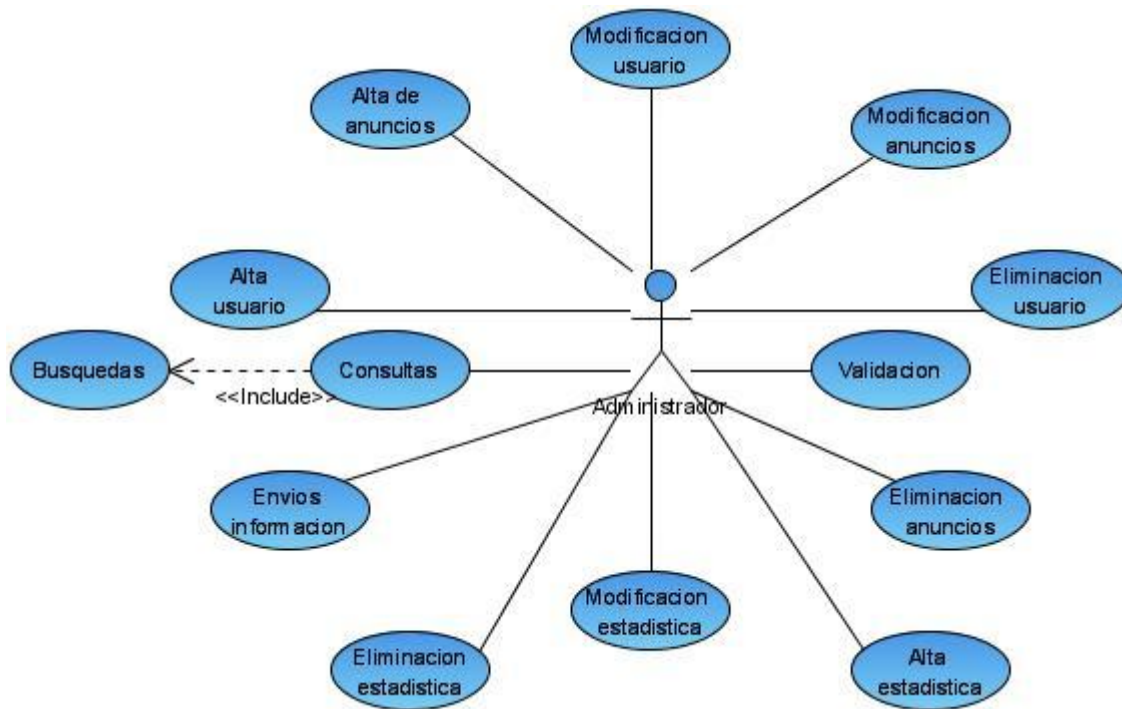


Figura 2-4 Diagrama de casos de uso administrador

Por comodidad y mayor entendimiento tomaremos la siguiente definición.

Usuario registrado.- Persona que obtuvo una cuenta en el sistema, es decir, tiene un usuario y una contraseña.

Usuario.- Persona que no posee un usuario y contraseña.

SIADEX
DESCRIPCIÓN DE CASOS DE USO "REGISTRO"

Nombre:	Registro
Actores:	Usuario
Función:	Permite obtener una cuenta de usuario en el sistema.
Descripción:	El usuario puede registrarse para obtener una cuenta, llenando un formulario con información de datos personales. El sistema debe validar que todos los campos obligatorios estén llenados correctamente. También es posible cancelar el registro si aun no opimió el botón Registrarse con los campos previamente validados.
Referencias:	De R1, R12.

DESCRIPCIÓN DE CASOS DE USO "CONSULTAS"

Nombre:	Consultas
Actores:	Usuario, Exalumno, Administrador
Función:	Permite hacer una consulta en general de toda la página.
Descripción:	El usuario selecciona la liga de interés y visualiza texto, si requiere hacer una búsqueda de Exalumnos la realiza, por año de egreso, carrera, nombre. Seleccionar opción correspondiente. Ingresar datos para realización de búsqueda.
Referencias:	De R9.

DESCRIPCIÓN DE CASOS DE USO "VALIDACION"

Nombre:	Validación
Actores:	Exalumno, Administrador.
Función:	Permite el acceso al sistema como usuario registrado y habilitar características diferentes para altas, modificaciones o eliminaciones.
Descripción:	El usuario registrado ingresa un nombre de usuario y una contraseña. El sistema valida la información que sea correcta de acuerdo a la información almacenada en la base de datos. En caso de olvidar la contraseña permite recuperarla enviando información personal.
Referencias:	De R1, R12.

DESCRIPCIÓN DE CASOS DE USO "ALTA ANUNCIO"

Nombre:	Alta anuncio
Actores:	Exalumno, Administrador
Función:	Permite dar de alta un anuncio clasificado (Conferencia, Taller o Congreso).
Descripción:	El usuario registrado selecciona el tipo de anuncio, llena un formulario ingresando un título, describe el anuncio y una fecha de expiración. El sistema debe validar la información y guardar en la base de datos.
Referencias:	De R2, R6, R13.

DESCRIPCIÓN DE CASOS DE USO "MODIFICACION ANUNCIO"

Nombre:	Modificación anuncio
Actores:	Exalumno, Administrador
Función:	Permite realizar modificaciones en un anuncio clasificado.
Descripción:	El sistema debe mostrar los anuncios en existencia es decir dados de alta, con la restricción de mostrar a un Exalumno únicamente sus anuncios publicados, si es un administrador podrá modificar cualquier anuncio aún sin pertenecerle o haberlo dado de alta. Modifica el anuncio clasificado ingresando el número de anuncio y nuevamente ingresa al formulario de alta cambiando el título y/o descripción de anuncio y/o fecha de expiración. El sistema debe validar la información y guardar en la base de datos.
Referencias:	De R2, R4, R5, R7, R13.

DESCRIPCIÓN DE CASOS DE USO "ELIMINACION ANUNCIO"

Nombre:	Eliminación anuncio
Actores:	Exalumno, Administrador
Función:	Permite realizar la eliminación de un anuncio clasificado.
Descripción:	El sistema debe mostrar los anuncios en existencia es decir dados de alta, con la restricción de mostrar a un Exalumno únicamente sus anuncios publicados, si es un administrador podrá eliminar cualquier anuncio aún sin pertenecerle o haberlo dado de alta. Elimina el anuncio clasificado ingresando el número de anuncio.
Referencias:	De R2, R4, R5, R7, R13.

Los casos de uso donde se dan de alta anuncios, se modifican anuncios y se eliminan anuncios tienen el mismo patrón que cuando se realizan las actividades para los usuarios y también las estadísticas, la restricción se dará con la validación del tipo de usuario y sus posibilidades de acceso al sistema.

En seguida se establecen los eventos que pueden generar los usuarios y van a ser atendidos por cada caso de uso. Todos los eventos van numerados en orden secuencial de acuerdo a la secuencia lógica como ocurrirían en la aplicación, cabe mencionar que al registrarse (obtener una nueva cuenta) el usuario se convierte en un exalumno, el cual tendrá mayores funcionalidades en accesos a la base de datos.

SIADEX			FORMATOS DE EVENTOS		
Nombre caso de uso Actores			Registro Usuario		
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor	Evento alternativo sistema
1	Oprimir opción Obtener una nueva cuenta				
2		Mostrar pantalla de captura de datos			
3	Escribir datos de usuario		a	Oprimir cancelar	
					a No graba datos termina caso de uso
4	Oprimir registrarse		a	Oprimir cancelar	
					a No graba datos termina caso de uso
5		Validar datos de nuevo usuario			a Validar datos: 1. Mensaje: Datos incompletos 2. Muestra formulario
					b Validar datos: 1. Mensaje: Usuario inexistente 2. Muestra formulario
6		Grabar datos en sistema			

SIADEX			FORMATOS DE EVENTOS		
Nombre caso de uso Actores			Consultas Usuario, Exalumnos, Administrador		
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor	Evento alternativo sistema
1	Oprimir liga de interés		a	Búsqueda por: 1. Carrera 2. Nombre 3. Generación	
					a Muestra pantalla para selección
			b	Selecciona Información	
					b Muestra resultados de búsqueda
2		Muestra información de la liga (objetivo, beneficios, anuncios clasificados, búsquedas) del sistema			
3	Clic en regresar				
4		Muestra página principal del sistema			

SIADEX			FORMATOS DE EVENTOS		
Nombre caso de uso Actores			Validación Exalumno, Administrador		
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor	Evento alternativo sistema
1	Escribir nombre de usuario y contraseña		a	No escribe datos	
			b	Oprime ingresar	
					a Mensaje: Usuario no registrado
2	Oprime ingresar				
3		Validar datos			a Validar: usuario o contraseña incorrecto
					b Mensaje: Usuario no registrado
4		Mostrar pantalla de usuario registrado			

SIADEX			FORMATOS DE EVENTOS		
Nombre caso de uso Actores			Alta anuncio Exalumno, Administrador		
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor	Evento alternativo sistema
1	Oprimir opción Agregar anuncio				
2		Mostrar pantalla de captura de datos			
3	Escribir datos de anuncio		a	No escribe datos	

			b	Oprimir Guardar		
					a	Validar datos: 1. Mensaje: datos incompletos 2. Muestra formulario
4	Oprimir Guardar					
5		Validar datos de anuncio				
6		Grabar datos en sistema				

SIADEX			FORMATOS DE EVENTOS			
Nombre caso de uso			Modifica anuncio			
Actores			Exalumno, Administrador			
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor		Evento alternativo sistema
1	Oprimir opción Modificar anuncio					
2		Muestra anuncios				
3	Elegir anuncio a modificar					
4	Oprimir enviar		a	Oprime cancelar		
					a	No actualiza datos termina caso de uso
5		Mostrar pantalla de modificacion de datos				
6	Oprimir actualizar					
7		Actualiza datos en sistema				
8		Mensaje: Anuncio actualizado				

SIADEX			FORMATOS DE EVENTOS			
Nombre caso de uso			Eliminar anuncio			
Actores			Exalumno, Administrador			
n	Flujo principal de evento actor	Flujo principal de eventos sistema		Evento alternativo actor		Evento alternativo sistema
1	Oprimir Eliminar anuncio					
2		Muestra anuncios				
3	Oprimir Eliminar en el anuncio elegido					
4		Elimina anuncio del sistema				
5		Mensaje: Anuncio Eliminado				

Los casos de uso de altas, modificaciones y eliminaciones, de usuarios, anuncios, y estadísticas siguen la misma secuencia de los descritos con anterioridad, la diferencia es la información que en cada uno de ellos se almacena, es decir, los campos que se llenan de información varían de acuerdo a lo solicitado, por lo tanto si se realiza una modificación o eliminación se ejecuta sobre los datos existentes para cada caso de uso.

Capítulo 3. Diseño

A través de este capítulo, se mostrará el desarrollo del diseño del sistema, en el cual abarcaremos, el diccionario de datos, el diagrama entidad-relación, y el modelo relacional. Además se justificará las formas normales en las que está basado el sistema.

Los diagramas nos ayudan a entender de forma fácil el comportamiento del sistema; a partir de las descripciones que se hicieron en el capítulo uno, los diferentes diagramas que existen en notación UML expresan información similar, desde diferentes vistas, en este documento solo se mostrará el siguiente ejemplo:

El diagrama de actividad y estados que se muestra en la figura 3-1 representa los estados que lleva una actividad, en los cuales podemos ver tres acciones llevadas a cabo: agregar, modificar y eliminar, éste diagrama hace referencia a las actividades sobre los anuncios pero como ya se menciono anteriormente es el mismo procedimiento que se tiene para agregar, modificar o eliminar usuarios.

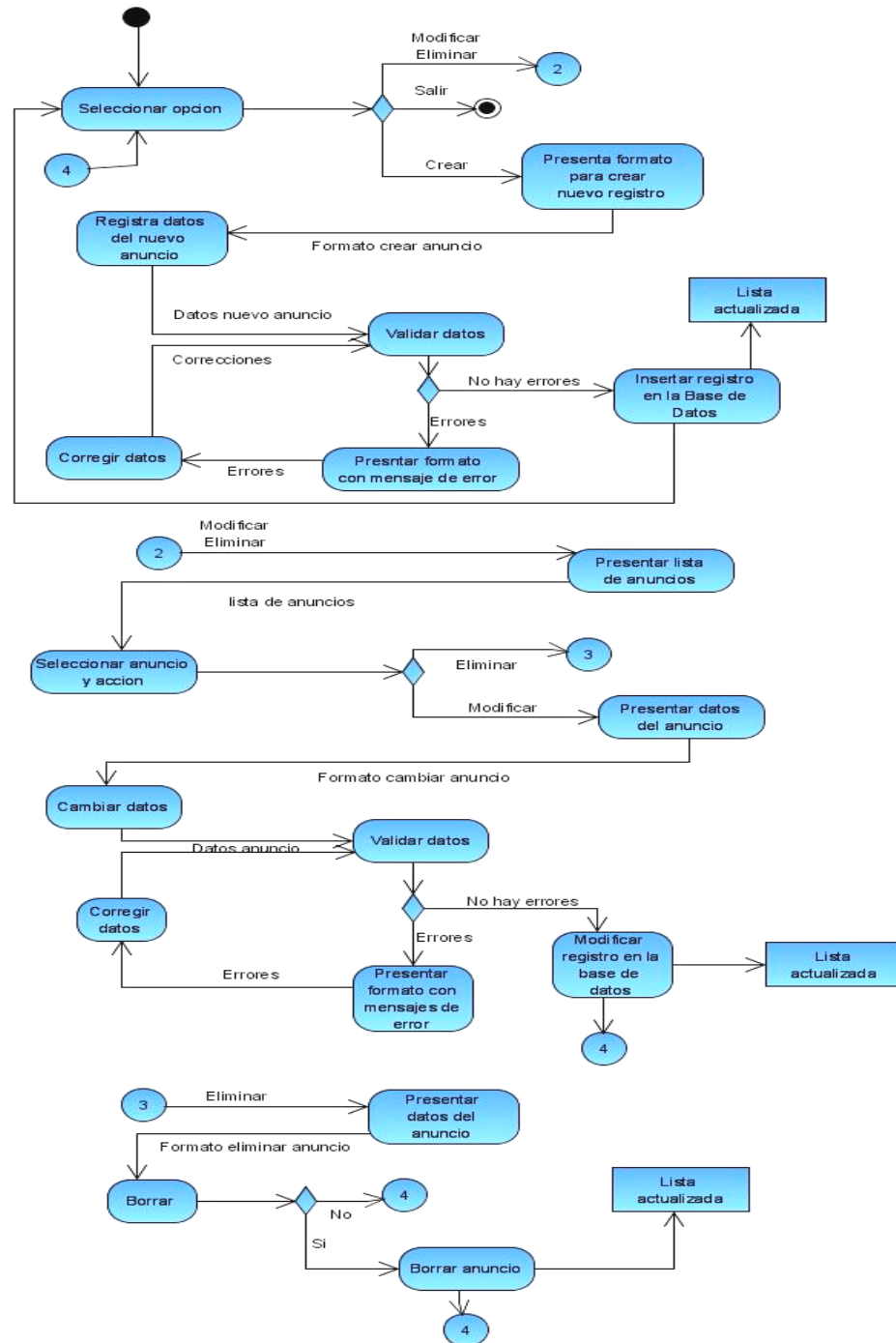


Figura 3-1 Diagrama de actividad y estados

En la figura 3-2 se muestra el diagrama de navegación del Sistema de Administración de Exalumnos, el cuál servirá para ver como se encuentran entrelazadas las páginas del sistema, es posible representarlo de forma horizontal o vertical:

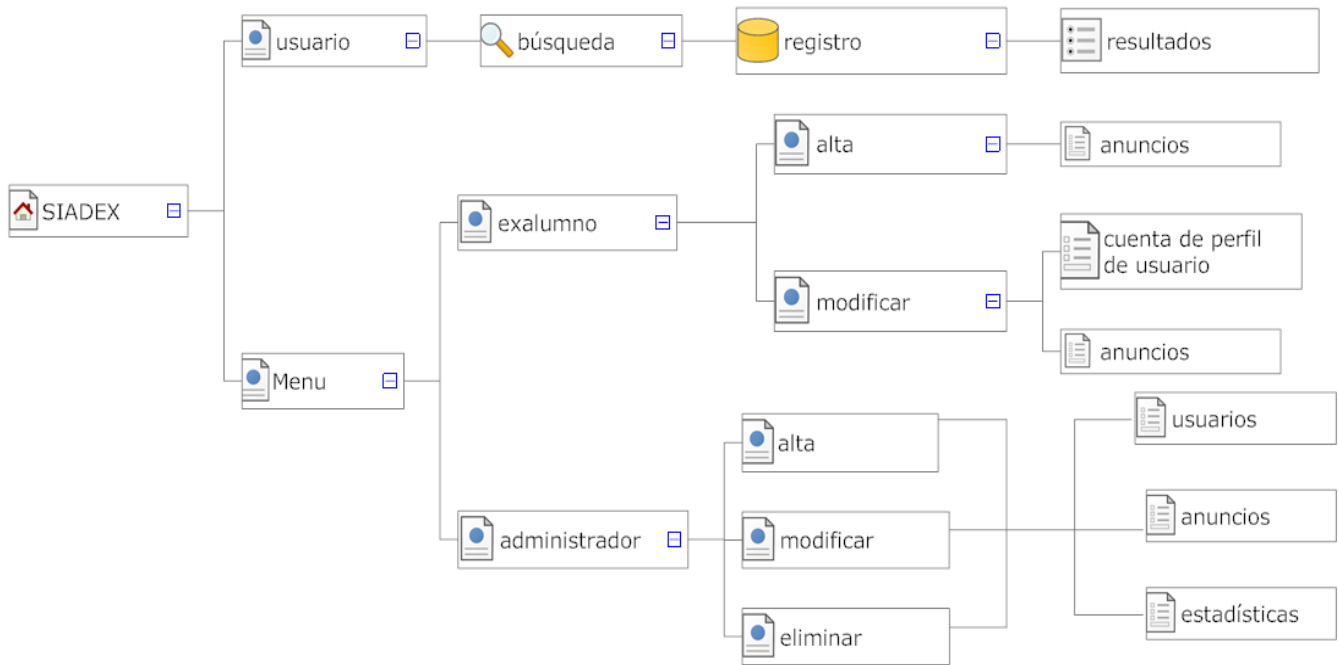


Figura 3-2 Diagrama de navegación

Diagrama entidad – relación

Para desarrollar el diagrama entidad – relación se analizan las entidades y sus relaciones, para el sistema SIADEX, se tienen tres entidades denominadas de la siguiente forma:

Usuario representa a toda persona que se ha inscrito en el sistema y puede tener acceso a él, la entidad usuario se describe por: matrícula, nombre, apellidos, sexo, fecha de nacimiento, fecha de ingreso, fecha de egreso, nombre de usuario, contraseña, pregunta, respuesta, ¿trabaja?, nombre compañía, puesto, correo electrónico, estado civil, ciudad, carrera, teléfono, celular, fecha de alta en el sistema, dirección.

Anuncio representa los anuncios que son guardados por los usuarios, los datos guardados que corresponden a un anuncio son: matrícula, fecha de alta, fecha de caducidad, título, texto y tipo.

Tipo representa un número del cual forman parte los anuncios, para clasificarlos en congresos, conferencias, talleres, se describe por: número de tipo y nombre.

Después del análisis del sistema procedemos a hacer nuestro diagrama entidad – relación, en el cual mostraremos las relaciones que existen entre los datos.

La primera relación que se toma, es la entidad del usuario con los anuncios, la cual tiene la cardinalidad de muchos a uno (m:1), debido a que el usuario puede publicar uno o más anuncios, y un anuncio puede ser publicado por un usuario. La relación entre ambas tablas se llamará *pública*.

La segunda relación es la de anuncios con tipo; la cardinalidad es de uno a muchos (1:m), puesto que un anuncio pueden tener un tipo pero un tipo puede tener muchos anuncios. La relación será nombrada *tiene*.

Una vez identificadas las relaciones y sus cardinalidades, se muestra el diagrama entidad – relación en la figura 3-2.

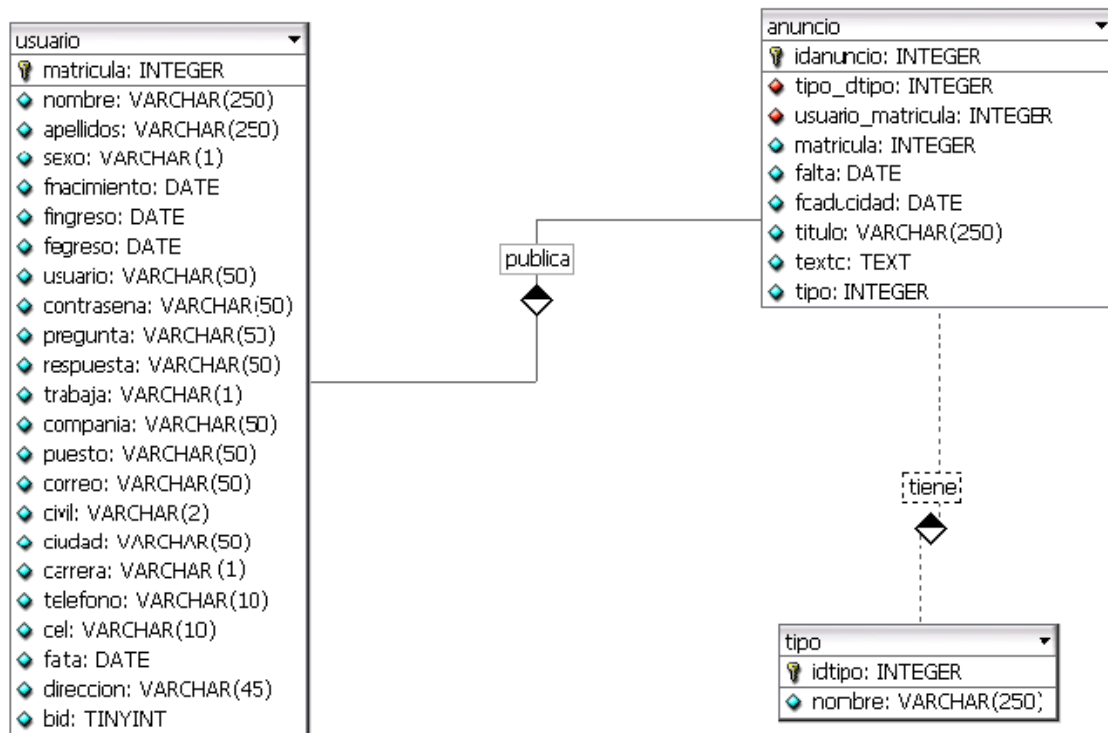


Figura 3-3 Diagrama entidad - relación

Diccionario de datos

Ahora se realiza el diccionario de datos, que tendrá los valores para cada atributo de las tablas obtenidas:

usuario

Atributo	Descripción	Tipo_dato	Longitud	Nulidad
Nombre	nombre del usuario	char	50	No
Apellidos	apellido del usuario	char	50	No
Matricula	Matricula correspondiente	numérico	11	No
Sexo	Sexo	char	1	No
fnacimiento	fecha de nacimiento	fecha		No
Fingreso	fecha de ingreso a la carrera	fecha		No
Fegreso	fecha de egreso de la carrera	fecha		No
Usuario	alias para ingresar como usuario registrado	alfanumérico	50	No
Contrasena	contraseña del usuario	alfanumérico	50	No
Pregunta	pregunta para recordar contraseña	char	50	No
Respuesta	respuesta a la pregunta secreta	char	50	No
Trabaja	respuesta a si trabaja o no	char	1	Si
Compañía	nombre de la compañía donde trabaja	char	50	Si
Puesto	ocupación en la compañía	char	50	Si
Correo	correo del usuario	char	50	No
Civil	estado civil	char	2	Si
Ciudad	ciudad	char	50	No
Carrera	lic. o ing.	char	1	No
Teléfono	Telefono	numérico	10	Si
Cel	Celular	numérico	10	Si
Falta	fecha de alta al sistema	fecha		No
Dirección	dirección actual	char	80	No
Bid	Tipo de usuario 0 administrador 1 usuario normal	numérico	1	no

La entidad usuario se relaciona directamente con anuncios, ya que el usuario publica anuncios.

anuncios

Atributo	Descripción	Tipo_dato	Longitud	Nulidad
Idanuncio	identificador del anuncio	numérico	10	No
Matricula	matricula del usuario que lo publica	numérico	11	No
Falta	fecha de alta del anuncio	fecha		No
Fcaducidad	fecha de caducidad del anuncio	fecha		No
Titulo	titulo del anuncio	char	250	No
Tipo	tipo del anuncio	numérico	11	No
Texto	texto o descripción del anuncio	char	250	No

Anuncio va relacionado con Tipo, ya que cada anuncio debe tener un tipo correspondiente.

tipo

Atributo	Descripción	Tipo_dato	Longitud	Nulidad
Idtipo	identificador del tipo	numérico	11	No
Nombre	nombre del tipo	char	250	No

Modelo relacional

Las tablas resultantes del modelo E –R son las siguientes:

usuario				
nombre	apellidos	<u>matricula</u>	sexo	Fnacimiento
fingreso	Fegreso	usuario	contraseña	Pregunta
respuesta	Trabaja	compania	puesto	Correo
Civil	Ciudad	carrera	telefono	Cel
Falta	Dirección	bid		

anuncios						
idanuncio	matricula	Falta	fcaducidad	titulo	Tipo	Texto

Tipo	
<u>Idtipo</u>	nombre

Normalización de la Base de Datos

Tabla Usuario:

Esta tabla esta en primera forma normal debido a que no se repite algún atributo dentro de ella, es decir, cada uno es atómico. En sus atributos solo contienen un valor con esto se justifica que esta en 1FN.

La segunda forma normal nos dice que todos los atributos dependerán de la llave primaria, por lo consiguiente esta tabla también esta en en 2FN, ya que sólo depende de su llave primaria. Ya que esta tabla esta en 1FN y 2FN, se puede deducir que esta en 3FN, ya que no tiene dependencias transitivas.

Tabla Tipo:

En esta tabla sólo se muestran 2 atributos, que son totalmente independientes, debido a esto la tabla cumple con la Primera Forma Normal.

Ninguno de ellos es una llave foránea, sólo hay una llave primaria y el otro atributo depende de ella. Por esta razón la tabla también está en segunda forma normal.

Como no existe otra llave dentro de esta tabla, no hay dependencias transitivas, y si esta en 1FN y 2FN entonces se concluye que la tabla esta en 3FN.

Tabla Anuncios:

Analizando esta tabla ningún atributo esta repetido, esto concuerda con la 1FN, además de que ninguno de sus atributos tiene múltiples valores, por esta razón la tabla se encuentra en 1FN.

Además, todos los atributos dependen de la llave primaria, así que también se encuentra en 2FN.

Como todos los atributos dependen de la llave primaria, y ninguno de ellos depende de la llave foránea, no puede haber ninguna dependencia transitiva, por esta razón también esta en 3FN. Y de esta forma la Base de Datos queda normalizada, hasta la 3FN.

Capítulo 4. Implementación

En este capítulo se documenta acerca de la implementación del sistema de administración de exalumnos, utilizando las herramientas de PHP y MySQL, y la interfaz mediante la cual se manipulará la base de datos.

Implementación de la interfaz (PHP)

Página de Inicio

En consecuencia del avance de la tecnología y del software que facilitan el trabajo del programador hoy en día, se utilizó como herramienta de trabajo "RapidPHP", por su sencilla forma de utilización con 4 vistas, editor de código, vista preliminar, división de pantalla horizontal, división de pantalla vertical.

La gran importancia de que un sistema web sea vistoso por sus usuarios, tiene mucho que ver con la apariencia del mismo, se crea la página de inicio, la cual consta de encabezado y pie de página, costado izquierdo de consultas comunes, costado derecho acceso de usuario registrado y centro refleja las acciones que se realizan de ambos costados, cuenta con animaciones Flash.

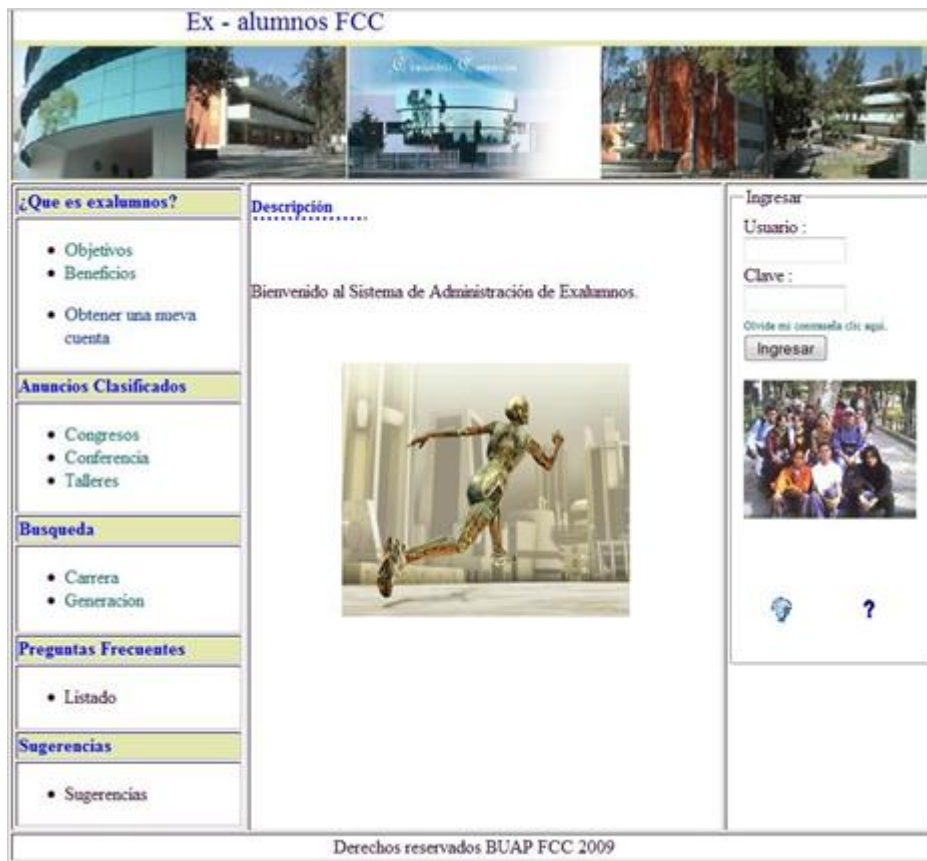


Figura 4-1 Página principal SIADEx

Las opciones que se muestran del costado izquierdo, son todas las acciones que un usuario no registrado puede realizar, si se da clic sobre objetivos se verá en pantalla la siguiente información:



Figura 4-2 Objetivo

La pantalla mostrada en la figura 4-2 contiene iconos en la parte inferior del texto del objetivo del sistema, los cuales sirven para regresar a la pantalla vista anteriormente, ir a la página principal de SIADEX. De igual forma al oprimir sobre "Beneficios" se muestra la figura 4-3 la cual contiene información con la propiedad de ser sólo lectura:

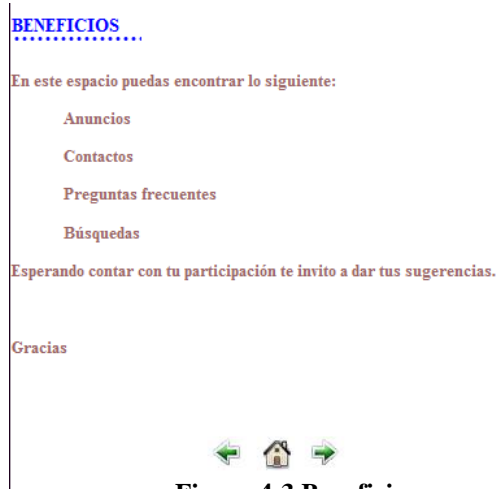


Figura 4-3 Beneficios

La información en pantalla es para todo tipo de usuario, las siguientes pantallas que se muestran en las figuras 4-4 y 4-5 son de búsquedas, requieren de información para poder llevar a cabo la acción.

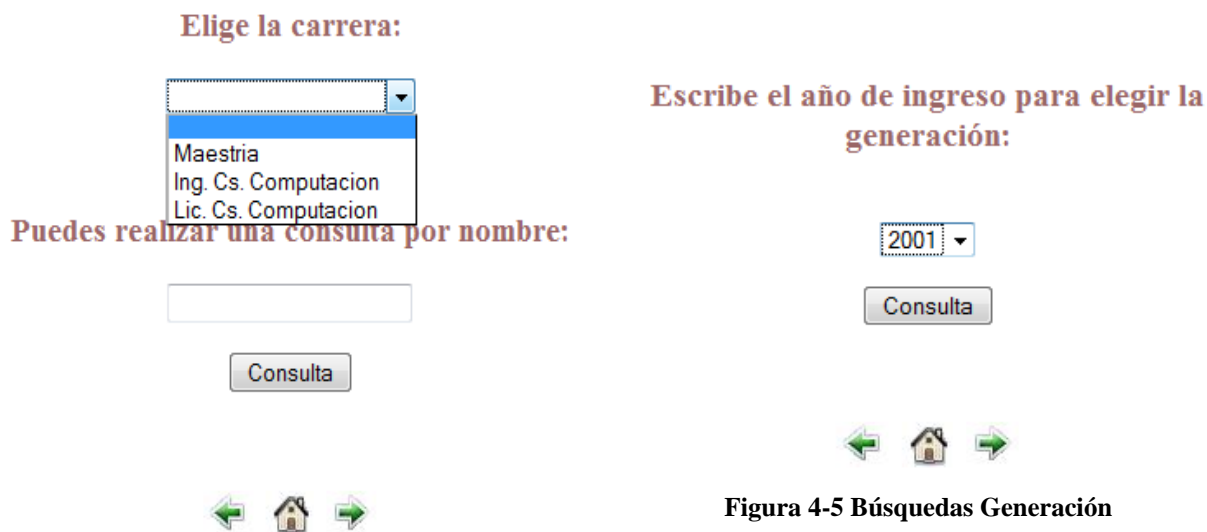


Figura 4-4 Búsquedas Carrera o Nombre

Figura 4-5 Búsquedas Generación

Al ingresar o seleccionar respectivamente la información, de acuerdo al tipo de búsqueda que se realiza, se obtienen los siguientes resultados en pantalla:

La figura 4-6 muestra los resultados de una búsqueda por carrera en Licenciatura en Ciencias de la Computación, la cual muestra todos los exalumnos registrados en la licenciatura, si se realiza una búsqueda por generación, en este caso por 2001 solo muestra a dos usuarios registrados, es decir las coincidencias que se localicen, la información mostrada es la ingresada por el usuario en el momento de realizar el registro de usuario (obtener una cuenta).

Datos de la búsqueda
.....

Nombre: Juan Carlos
Apellidos: Ramirez
Correo: juancho@gmail.com
Generación: 1990

Nombre: Casiviani
Apellidos: Fuentes Castillejos
Correo: cachifc@hotmail.com
Generación: 2001

Nombre: Jessica
Apellidos: Marcial Romero
Correo: marjess@gmail.com
Generación: 2001



Figura 4-6 Resultados de búsqueda

La figura 4-7 muestra el formato en general de los anuncios clasificados (congresos, conferencias, talleres) que sirven de información para todo tipo de usuario, la propiedad de ellos es solo lectura, para aquellos usuarios no registrados.

<p>Título: IMSS</p> <p>Tratamientos quirúrgicos</p> <p>Autor: Juan Carlos Ramirez</p> <p>Fecha de expiración: 2009-03-30</p>
<p>Título: Vacaciones</p> <p>Hola estos son los congresos que estaran disponibles en las vacaciones</p> <p>Autor: Jessica Marcial Romero</p> <p>Fecha de expiración: 2009-04-13</p>

Figura 4-7 Anuncios

Si se desea "Obtener una nueva cuenta", se tiene que proporcionar información en el formulario que se muestra en la figura 4-8, con ellos se podrá acceder al sistema y realizar otro tipo de actividades sobre esté:

Usuario:	Contraseña:
Pregunta frecuente:	Respuesta:
Correo electrónico:	Matricula:
Nombre:	Apellidos:
Dirección:	Ciudad:
Teléfono:	Celular:
Sexo: <input type="radio"/> M <input type="radio"/> F	Estado Civil: Soltero (a) <input type="text"/>
Fecha de nacimiento: Día 1 <input type="text"/> Mes Enero <input type="text"/>	Año 1975 <input type="text"/>
Carrera: Lic. Cs. Computacion <input type="text"/>	
Fecha de ingreso: 1990 <input type="text"/>	Fecha de egreso: 1990 <input type="text"/>
Trabajas: <input checked="" type="radio"/> Si <input type="radio"/> No	
Compañía o Institución:	Puesto:
<input type="button" value="Registrarse"/>	<input type="button" value="Cancelar"/>

Figura 4-8 Obtener una nueva cuenta

Interfaz de usuario “Exalumno”

La figura 4-9 muestra la forma de ingresar al sistema, el cual mediante un nombre de usuario y una contraseña se identifica y tiene acceso a realizar opciones como agregar, modificar y eliminar.

The screenshot shows a login form titled "Ingresar" (Login). It contains two input fields: "Usuario :" (User) and "Clave :" (Password). Below the password field is a link that says "Olvide mi contraseña clic aqui." (Forgot my password click here). At the bottom of the form is a button labeled "Ingresar". Two callout boxes with arrows point to the input fields: one points to the "Usuario :" field with the text "Nombre de usuario registrado" (Registered user name), and the other points to the "Clave :" field with the text "Contraseña" (Password).

Figura 4-9 Usuario registrado

Existe la opción de recuperación de contraseña en donde se oprime en la liga de “Olvide mi contraseña clic aquí”, se pide ingresar la matrícula y la respuesta a la pregunta frecuente, como se muestra en la figura 4-10:

La pregunta es: relacion

The screenshot shows a password recovery form. It has two input fields: "Respuesta:" (Answer) and "Matricula:" (Matriculation number). Below the "Matricula:" field is a button labeled "Enviar" (Send). Two callout boxes with arrows point to the input fields: one points to the "Respuesta:" field with the text "Ingresar la respuesta de la pregunta frecuente registradas al momento de obtener cuenta" (Enter the answer to the frequent question registered at the time of getting the account), and the other points to the "Matricula:" field with the text "Confirmar ingresando matrícula" (Confirm by entering matriculation number).

Figura 4-10 Olvide mi contraseña

Cuando el exalumno se ha identificado, es decir, se ha validado la información que proporcionó, ingresa con su nombre de usuario reflejado en el costado derecho de la pantalla del sistema, y una opción de Cerrar Sesión, que sirve para que el usuario se desconecte del sistema, se muestra en figura 4-11.

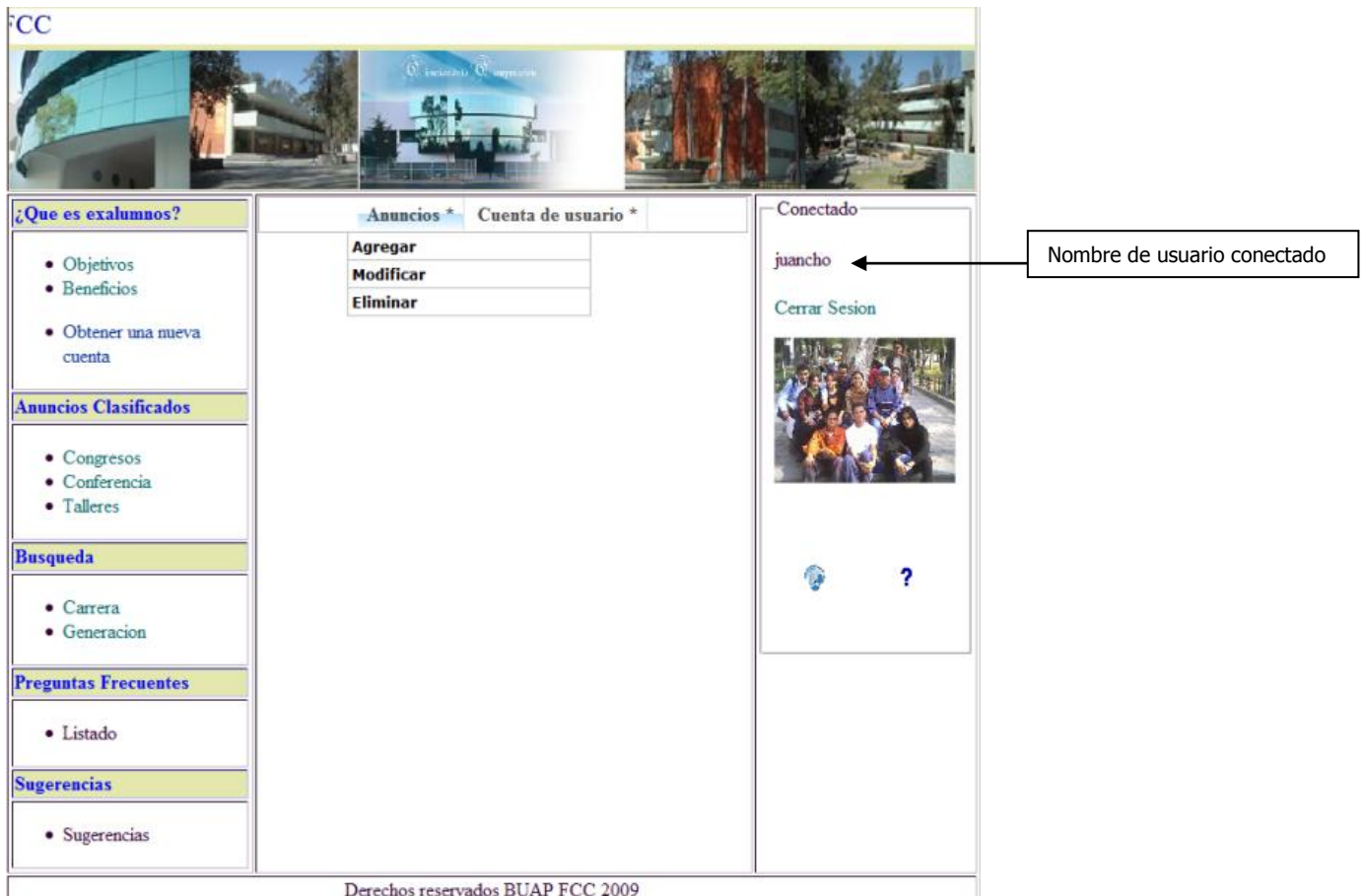


Figura 4-11 Sesión iniciada

El menú que le permite agregar, modificar o eliminar anuncios, también permite modificar su perfil de usuario, figura 4-12.

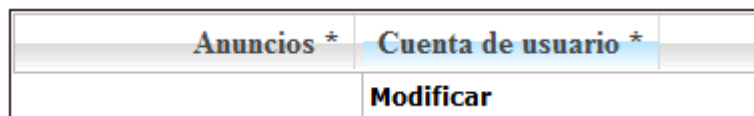


Figura 4-12 Menú de exalumno

Si se ha elegido modificar cuenta de usuario, se especifica que no es necesario modificar o escribir nuevamente todos los campos con información, figura 4-13:

No necesita escribir todos sus datos nuevamente solo los que requiera modificar

Usuario:	Contraseña:
Pregunta frecuente:	Respuesta:
Correo electrónico:	
Nombre:	Apellidos:
Dirección:	Ciudad:
Teléfono:	Celular:
Sexo: <input type="radio"/> M <input type="radio"/> F	Estado Civil: Soltero (a) ▾
Fecha de nacimiento: Día 1 ▾ Mes Enero ▾	Año 1975 ▾
Carrera: Lic. Cs. Computacion ▾	
Fecha de ingreso: 1990 ▾	Fecha de egreso: 1990 ▾
Trabajas: <input checked="" type="radio"/> Si <input type="radio"/> No	
Compañía o Institución:	Puesto:
Actualizar	Cancelar


Figura 4-13 Modificar exalumno



Los anuncios publicados son de gran importancia, debido que a través de estos se mantendrá un contacto con los usuarios que utilicen el sistema. Una de las acciones a realizar sobre los anuncios es agregar, del cual se muestran las figuras 4-14 y 4-15.

Anuncios *	Cuenta de usuario *
Agregar	
Modificar	
Eliminar	

Figura 4-14 Agregar anuncio

Ex - alumnos FCC



<p>¿Que es exalumnos?</p> <ul style="list-style-type: none"> • Objetivos • Beneficios • Obtener una nueva cuenta <p>Anuncios Clasificados</p> <ul style="list-style-type: none"> • Congresos • Conferencia • Talleres <p>Busqueda</p> <ul style="list-style-type: none"> • Carrera • Generacion <p>Preguntas Frecuentes</p> <ul style="list-style-type: none"> • Listado <p>Sugerencias</p> <ul style="list-style-type: none"> • Sugerencias 	<p>Anuncios * Cuenta de usuario *</p> <hr/> <p>Anuncios</p> <p>Tipo de anuncio: Conferencia ▾</p> <p>Título: <input type="text"/></p> <p>Anuncio: <input style="width: 100%; height: 60px;" type="text"/></p> <p>Fecha de expiracion: <input type="text"/> </p> <p style="text-align: center;"><input type="button" value="Guardar"/></p>	<p>Conectado</p> <p>juancho</p> <p>Cerrar Sesion</p>  <p style="text-align: center;"> ?</p>
---	---	---

Derechos reservados BUAP FCC 2009

Figura 4-15 Agregar anuncio

Una vez que un usuario ha agregado anuncios, este podrá modificar sus anuncios publicados de los cuales las figuras 4-16 y 4-17 muestran la forma de realizar las modificaciones:

Anuncios *	Cuenta de usuario *
(1) Título: .NET	
Tema: La nueva forma de programación en un entorno de .NET	
Autor: Casiviani Fuentes Castillejos Fecha de expiración: 2009-04-27	
(2) Título: Internet 2	
Tema: Velocidad y mejoras en Internet	
Autor: Casiviani Fuentes Castillejos Fecha de expiración: 2009-04-30	

Elige el anuncio que quieras modificar

Figura 4-17 Modificación anuncio

Elige el número de anuncio que se va a modificar

Anuncios *	Cuenta de usuario *
Anuncios	
Tipo de anuncio:	Conferencia ▼
Título:	Internet2
Anuncio:	Velocidad y mejoras en Internet
Fecha de expiración:	2009-04-30 <input type="button" value="Calendar"/>
<input type="button" value="Actualizar"/>	

Figura 4-16 Modificar anuncio

La primera acción a realizar es elegir modificar del menú de anuncios, la cual muestra la ventana que contiene los anuncios publicados por el propietario de la cuenta que se encuentra activa en ese momento, al elegir el número de anuncio que se va a modificar y confirmar el envío, la nueva pantalla muestra la información que contiene ese anuncio para que se modifique y se confirma pulsando en el botón de actualizar.

La forma de eliminar un anuncio se muestra en la figura 4-18, en el ejemplo se pueden observar los anuncios existentes por la cuenta activa, se tiene que dar clic sobre la palabra Eliminar en la fila que se encuentra el anuncio deseado a ser borrado.

Anuncios *		Cuenta de usuario *	
Eliminar Anuncio			
Título	Texto	Fecha de caducidad	Eliminar
IMSS	Tratamientos quirurgicos	2009-03-30	Eliminar
Sistemas web	Creacion y mantenimiento	2009-04-21	Eliminar
Obtener titulo	Tramite para titulacion	2009-03-31	Eliminar
T-systems	Alumnos que han cumplido 95% de creditos de la ingenieria o licenciatura de Ciencias de la Computacion que esten interesados	2009-04-30	Eliminar
GEDAS	Practicas profesionales	2009-04-30	Eliminar

Clic para eliminar anuncio

Figura 4-18 Eliminar anuncios

Interfaz de usuario “Administrador”

La interfaz de usuario de un administrador es similar a la interfaz de un exalumno, la diferencia se presenta en el menú que se despliega al conectarse con una cuenta de administrador, validado en la base de datos del sistema, figura 4-19.

Anuncios *	Usuario *	Estadísticas *	Inicio *
Agregar			
Modificar			
Eliminar			

Figura 4-19 Administrador

Todas las opciones que despliega el menú anterior nos muestran un submenú de: agregar, modificar y eliminar. Las pantallas mostradas con anterioridad de la forma de agregar, modificar o eliminar, incluyen la forma de realizar las acciones de un administrador, las actividades incrementadas son que el administrador ve, modifica, agrega y elimina no solo lo publicado por él sino puede hacerlo en todo tipo de información sin limitarse, tal el caso de eliminar a un usuario el cual se muestra en la figura 4-20:

Eliminar
Usuario
.....

Usuario	Nombre(s)	Apellido(s)	Eliminar
Juancho	Juan Carlos	Ramirez	Eliminar
200112505	Casiviani	Fuentes Castillejos	Eliminar
Jessica	Jessica	Marcial Romero	Eliminar

Figura 4-20 Eliminar usuarios

La figura 4-21 muestra la modificación de anuncios, la cual se realiza similar a la descripción de las figuras 4-16 y 4-17, sin embargo se puede observar que los anuncios que se pueden modificar no son solo los pertenecientes a la cuenta activa, por poseer privilegios de administrador.

Anuncios *	Usuario *	Estadísticas *	Inicio *
(1) Título: .NET			
Tema: La nueva forma de programación en un entorno de .NET			
Autor: Casiviani Fuentes Castillejos Fecha de expiración: 2009-04-27			
(2) Título: IMSS			
Tema: Tratamientos quirurgicos			
Autor: Juan Carlos Ramirez Fecha de expiración: 2009-03-30			
(3) Título: Programacion en C			
Tema: Teorico, practico			
Autor: Jessica Marcial Romero Fecha de expiración: 2009-05-05			

Anuncios publicados por diversos autores

Figura 4-21 Todos los anuncios

Las estadísticas que se presentan gráficamente son realizadas de acuerdo a información proporcionada por exalumnos.

Existen algunos iconos que sirven para facilitar el traslado a otras páginas o de ayuda para mejoramiento del sistema, figura 4-22.

Buzón de sugerencias para mejoramiento del sistema web

Derechos reservados BUAP FCC 2009
Figura 4-22 SIADEX

Conclusiones

En el presente trabajo de tesis se cumplió el objetivo general planteado. Se llevó a cabo el análisis, diseño e implementación de un sistema web, que permite la administración de exalumnos, describiendo los servicios que ofrece de manera fácil y rápida.

El sistema funciona bajo cualquier navegador debido a su portabilidad.

Aunque el trabajo realizado no es nuevo, puede ser considerado como una aportación a la Facultad de Ciencias de la Computación.

Perspectivas

En estos momentos se está, realizando pruebas desde diferentes terminales, creando las modificaciones que se observan o incrementando algunos servicios disponibles.

Integrar mecanismos de confiabilidad y seguridad para el envío y recepción de información.

Se espera, que la comunidad estudiantil y los exalumnos de la Facultad de Ciencias de la Computación, participen en este proyecto una vez puesto en marcha, debido a la importancia que tiene para el éxito de éste.

Apéndice A: Instalación del servidor

Se muestra los pasos para crear una base de datos desde un entorno gráfico utilizando wamp (mamp) desde el phpMyAdmin

Instalación de WAMP

Para instalar WAMP Server descargarlo de la red para tener el archivo ejecutable en nuestra computadora.

Cuando haya finalizado la instalación, ejecutamos WAMP Server y aparecerá un icono como el que se muestra en la figura 4-23:



Figura 4-23 Icono de wamp

Implementación de la Base de Datos

Para implementar la base de datos desde una interfaz gráfica para evitar las consolas, podemos ocupar WAMP, en la cual se crea de manera automática la base de datos, y se introducen cada una de las tablas de manera sencilla.

En la figura 5-2 se muestra la forma de crear una nueva base de datos desde phpmyadmin.

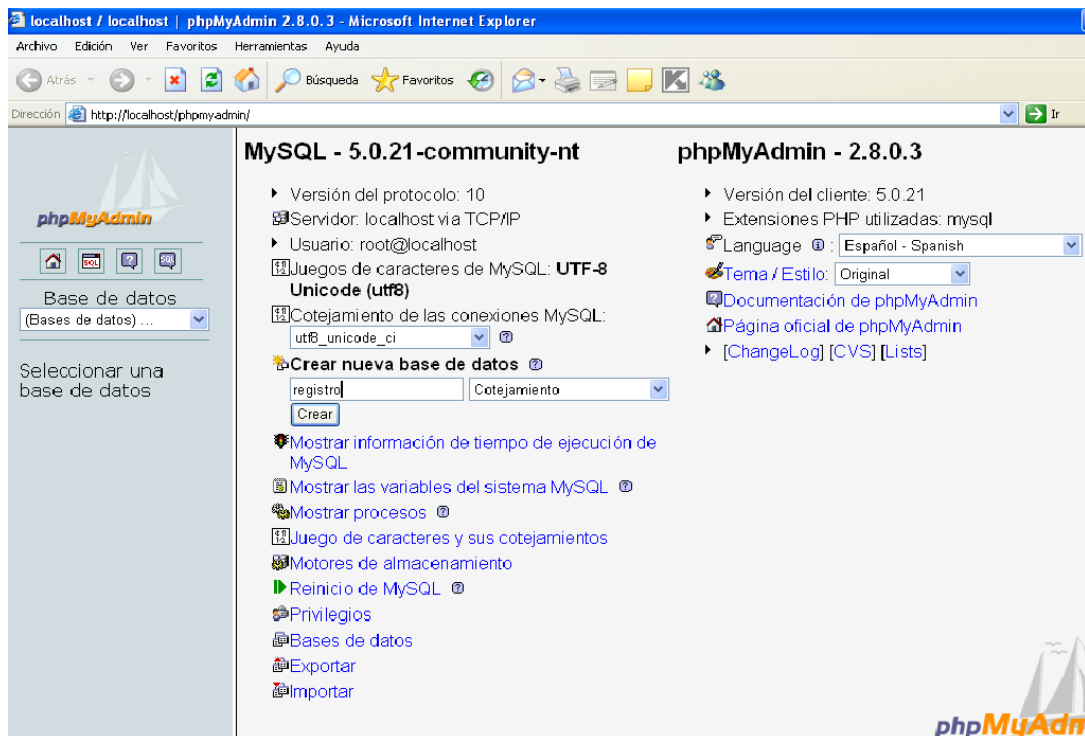


Figura 4-24 Pantalla de phpMyAdmin

Una vez realizada la base de datos sobre la cual trabajaremos, es necesario crear las tablas correspondientes para el almacenamiento de la información, es decir las entidades requeridas.

Debido a que es una interfaz gráfica sobre la cual estamos trabajando únicamente escribiremos el nombre de la tabla y el número de campos que necesitamos. En la figura 5-3 se muestra la forma de crear una tabla.

Crear nueva tabla en la base de datos registro

Nombre: Número de campos:

Figura 4-25 Creación de tablas

Después de haber creado la tabla requiere de otro sencillo paso, seleccionar los datos que lleva la misma así como los valores que lleva cada campo.

localhost / localhost / registro / tipos | phpMyAdmin 2.8.0.3 - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Dirección http://localhost/phpmyadmin/

Servidor: localhost ▶ Base de datos: registro ▶ Tabla: tipos

Examinar Estructura SQL Buscar Insertar Exportar Importar Operaciones

Vaciar Eliminar

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> idtipo	int(11)			No		auto_increment
<input type="checkbox"/> nombre	varchar(250)	utf8_general_ci		No		

↑ Marcar todos/as / Desmarcar todos Para los elementos que están marcados:

Vista de impresión Vista de relaciones Planteamiento de la estructura de tabla

Añadir 1 campo(s) Al final de la tabla Al comienzo de la tabla Después de idtipo Continuar

Índices:					Espacio utilizado		
Nombre de la clave	Tipo	Cardinalidad	Acción	Campo	Tipo	Uso	
PRIMARY	PRIMARY	3		idtipo	Datos	60	Bytes
					Índice	2,048	Bytes
					Total	2,108	Bytes

Crear un índice en 1 columna(s) Continuar

Estadísticas de la fila	
Enunciado	Valor
Formato	dinámico/a
Cotejamiento	utf8_general_ci
Filas	3
Longitud de la fila ø	20
Tamaño de la fila ø	703 Bytes

Figura 4-26 Insertando campos en tablas

Bibliografía

[1]. -Ingeniería de software teoría y práctica
Shari Lawrence Pfleeger
Prentice Hall, 2002
ISBN: 9879460715

[2].-Fundamentos y modelos de bases de datos,
Adoración de Miguel
Mario Piattini, Martín, 1975.
Alfaomega, Ra-ma
ISBN: 9586824993

[3].- Introducción a los sistemas de bases de datos
C.J. Date
Pearson Educación, 2001
ISBN: 968-444-419-2

[4].-<http://www.clikear.com/manuals/uml/diagramasouso.asp>

[5].- Aprenda desarrollo de bases de datos web ya
Jim Buyens
Mc Graw Hill, 2001
ISBN: 84-481-2903-2

[6] Ingeniería del software. Un enfoque práctico
Roger S. Pressman
Mc Graw Hill, 2005
ISBN: 9701054733

[7] Fundamentos y modelos de bases de datos
Adoración de Miguel Castaño Mario G. Piattini Velthuis
Ra-ma, 1999
ISBN: 8478973613

[8] MySQL5
Juan Diego Gutiérrez Gallardo
Anaya Multimedia, 2005
ISBN: 9788441519367

[9] Análisis y diseño orientado a objetos con aplicaciones
Grady Booch, Addison Wesley Iberoamericana, S.A.
USA 1996
ISBN: 0-201-60122-2

[10] El Lenguaje Unificado de Modelado. Manual de referencia.
James RUMBAUGH, Ivar JACOBSON, Grady BOOCH Rational Software Corporation
Addison Wesley Longman Inc.
ISBN: 84-7829-037-0

[11] Navegar en Internet Creación de un portal con PHP y MySQL 2ª Edición
Jacobo Pavón Puertas
Ra-Ma 2006
ISBN: 84-7897-690-6

[12] PHP5 a través de ejemplos
Abraham Gutiérrez Rodríguez, Ginés Bravo García
Alfaomega Ra-Ma
Madrid 2005
ISBN: 84-7897-656-6

[13] <http://www.apache.org>

[14] http://www.mundoprogramacion.com/colabora/puntoNET/canchala_UML.htm

[15] http://es.wikipedia.org/wiki/Ingenier%C3%ADa_web