



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

“DESARROLLO E IMPLEMENTACIÓN DEL DIRECTORIO DE
FUNCIONARIOS Y REPRESENTANTES DE LA FACULTAD DE
INGENIERÍA EN LÍNEA”

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

**ABAD MARTÍNEZ DAFNE CITLALLI
LÓPEZ HERNÁNDEZ ELY DANIEL**

DIRECTORA DE TESIS:

M. en I. ABIGAIL SERRALDE RUÍZ



Ciudad Universitaria 2007



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

A Dios por que me ha hecho constancia de su existencia en diversas ocasiones, siendo esta una de las más grandes.

A mis padres por darme el hermoso regalo de estar vivo, por confiar en mí y apoyarme en todo sin enjuiciar nunca ninguna de mis decisiones, por ser los más Grandes . . .

A mi hermana por que siempre ha estado allí dispuesta a lo que sea por mí y creciendo en su propio mundo.

A mi familia por ser tan grande y estar conformada por personas muy valiosas.

Al Ing. César por su apoyo incondicional y ser la persona clave en mi vida profesional, mi amigo, mi Maestro.

A mi Universidad que amo y amaré toda la vida por darme específicamente Todo.

A la banda Cuil porque todos somos profesionistas y conocemos el verdadero valor de la amistad.

A todos mis amigos, porque cada uno es muy especial para mí, y porque uno a uno han hecho mi corazón más grande aún.

*A el gran amor de mi vida, a quien tal vez aún no he tenido la fortuna de conocer
.....*

A mis hijos, aunque en este momento no existan físicamente, pero que siempre han estado en mi mente, y a quienes hoy tengo el privilegio de dedicar este trabajo.

Ely Daniello

ÍNDICE

CAPÍTULO 1 ANTECEDENTES

1.1 Marco Teórico de Referencia Institucional	1
La Secretaría General	2
El Departamento de Información y Estadística	3
El Directorio de Contactos de la Facultad de Ingeniería	5
Necesidad de un Sistema para Automatizar el Procedimiento de Realización y Actualización del Directorio	7

CAPÍTULO 2 MARCO TEÓRICO CONCEPTUAL

Introducción	9
2.1 La ingeniería de software	10
2.2 Redes de Computadoras	18
Seguridad en redes	28
2.3 Lenguajes de Programación	31
2.4 Bases de Datos	38
Normalización	42
Sistema Manejador de Bases de Datos (SMBD)	44

CAPÍTULO 3 ANÁLISIS Y DISEÑO

3.1 Diagrama de Flujo de Datos	56
3.2 Diagramas de Flujo de Datos Detallados	57
3.4 Diagrama Entidad Relación Modelo Lógico	59
3.5 Definición de Estándares	63
3.6 Diccionario de Datos	64
	65

CAPÍTULO 4 DESARROLLO

4.1 Estándares de desarrollo	70
4.2 Desarrollo y manejo del BackOffice o administración del sistema	74
Validación de usuario y contraseña	75
Acciones que se pueden realizar en la administración del sistema	76
Conclusiones y Recomendaciones	87
Mantenimiento	88
Glosario	90
Bibliografía	94

Prólogo

El objetivo de este trabajo es reportar la implementación de un sistema de consulta y administración electrónica que contiene los datos generales de la comunidad que labora en la Facultad de Ingeniería, haciendo un uso eficiente de la infraestructura con la que cuenta la institución dada su necesidad de obtener información de manera rápida y precisa. Este sistema pretende facilitar el acopio, actualización y homogeneización de estos datos contribuyendo así a la consistencia de la información.

De manera muy sencilla, en este trabajo se exponen temas como bases de datos, ingeniería de software, redes de computadoras y lenguajes de programación con la finalidad de tener elementos para determinar las herramientas adecuadas para la implementación del sistema.

Utilizando páginas dinámicas creadas en lenguaje PHP y utilizando MySQL como sistema manejador de bases de datos, esta aplicación permite tener una administración de la información descentralizada y remota, donde cada Secretaría y División Académica se hará responsable de los datos del personal adscrito a ella así como de su actualización.

Esta solución fue dirigida principalmente al Departamento de Información y Estadística, pues este departamento maneja de primera instancia toda esta información, sin embargo el beneficio alcanzará a diferentes áreas: divisiones, departamentos, jefaturas, etc. de la Facultad además de las personas interesadas en esta información desde cualquier parte del mundo ya que podrán contar con un servicio más dentro del portal de la Facultad.

Pretende contribuir a un manejo más sencillo de los procesos y ofrecerá plena confianza en los datos que se publiquen dentro de este sistema pues su actualización pretende llevarse a cabo desde los diferentes Departamentos evitando así confusiones y errores.

En el primer capítulo se explica el entorno institucional del directorio de la Facultad de Ingeniería así como la justificación para el desarrollo del sistema que es motivo del presente trabajo. El segundo capítulo exhibe la fundamentación teórica de los conceptos empleados como son Bases de Datos, Redes de computadoras, Ingeniería de programación, etc. El tercer capítulo muestra de manera descriptiva el análisis de la situación a resolver así como la etapa de diseño del sistema mediante los diagramas de flujo de datos, definición de estándares y diccionario de datos correspondientes. En el cuarto capítulo se explica el desarrollo del sistema mediante la descripción de sus aplicaciones desde la interfaz del usuario común hasta la del usuario que goza de privilegios para administrar la información contenida en el sistema. Se incluyen además conclusiones y recomendaciones así como un glosario y la bibliografía empleada.

Capítulo 1 Antecedentes

1.1 Marco Teórico de Referencia Institucional

Dentro de las Escuelas y Facultades de la Universidad Nacional Autónoma de México, se encuentra la Facultad de Ingeniería, una de las escuelas más antiguas del país que ha sido clave en la formación de recursos humanos para el desarrollo de México en todos los ámbitos.

La Facultad de Ingeniería ha evolucionando acorde a los nuevos tiempos, adaptándose a la modernidad e innovando con ejemplos claros como la revisión periódica de sus planes de estudio y la creación de nuevas carreras. Está ampliamente relacionada con la vida diaria del país, pues además de la formación de profesionistas de alto nivel, está involucrada con la investigación en el desarrollo de nuevos recursos tecnológicos por medio de sus Estudios de Posgrado de Educación Continua en estrecha relación con el Instituto de Ingeniería.

Esta entidad de la Universidad Nacional Autónoma de México en su parte académica actualmente se constituye por 935 profesores de asignatura, 245 profesores de carrera, 104 técnicos académicos, 4 investigadores y 274 ayudantes de profesor, de los cuales alrededor del 50% cuenta con estudios de posgrado o especialización.

La organización básica de la Facultad de Ingeniería se muestra en la Figura 1.1.



Figura 1.1 Organigrama principal de la Facultad de Ingeniería.

La Secretaría General

En el siguiente nivel jerárquico a la Dirección y el Consejo Técnico, se encuentra la Secretaría General. Las funciones de la Secretaría General son: planificar, implantar y coordinar la ejecución de los asuntos de carácter académico de la Facultad de manera incluyente respecto a la participación de los directivos en la planeación y administración de la Institución. Implícitamente los planes y programas de trabajo necesarios para el buen desempeño de la Facultad los cuales son elaborados, implementados y supervisados por esta entidad.

Entre las funciones del Secretario General de la Facultad de Ingeniería, se encuentra la de ejercer como Secretario del Consejo Técnico, proporcionar a las Divisiones de la Facultad el apoyo que le soliciten en lo relativo a servicios de información, de comunicación, de cómputo y otros de su competencia, solicitar a las unidades a su cargo la formalización de estudios, investigaciones e informes que requieran el desarrollo académico de la Institución, colaborar con el Director en las actividades de planeación y evaluación de apoyo académico de la facultad.

Además de impulsar y realizar coordinadamente el proceso de planeación, administración y difusión académica de la Facultad, el Secretario General también tiene a su cargo la supervisión y evaluación del cumplimiento de los acuerdos del Consejo Técnico.

El establecimiento de las políticas relativas al cómputo académico es también una actividad inherente del Secretario General, que tiene a su cargo la coordinación y actualización de distintos sistemas de información que se utilizan para el procesamiento de información requerida por la propia Facultad y otras dependencias además de las autoridades Universitarias.

Todas estas actividades son apoyadas mediante el trabajo de las siguientes Coordinaciones, Unidades y Departamentos que conforman la Secretaría General:

Unidad de Servicios de Cómputo Académico (UNICA).

Coordinación de Comunicación.

Coordinación de Programas de Atención Diferenciada para Alumnos (COPADI).
Coordinación de Programas de Superación del Personal Académico (CPSPA).
Coordinación de Procesos e Información del Consejo Técnico (CPICT).
Departamento de Personal Académico (DPA).
Unidad de Apoyo Editorial (UDAE).
Centro de Docencia (CDD).
Coordinación de la Revista de Ingeniería.
Coordinación de Evaluación Educativa.
Departamento de Información y Estadística (DIES).

El Departamento de Información y Estadística

El Departamento de Información y Estadística (DIES), tiene entre sus principales funciones la organización y administración del acceso a la información consolidada del personal académico de la Facultad de Ingeniería para lo cual hace uso de recursos tecnológicos tales como: bases de datos, redes de datos, páginas Web, redes de computadoras, sistemas de información, etc.

En este sentido la clasificación y proceso de la información estadística referente al personal académico de la Facultad demanda la realización de sistemas de cómputo y procedimientos correspondientes, quedando estas tareas a cargo del Departamento de Información y Estadística.

La representación de la Facultad de Ingeniería ante la Universidad en el ámbito de Información y Estadística esta sustentada por el trabajo del DIES. Entre sus múltiples responsabilidades están la coordinación a nivel Facultad de programas institucionales tales como el PAPIME (Programa de Apoyo a Proyectos Institucionales para el Mejoramiento de la Enseñanza) y el PAPIIT (Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica); la actualización de sistemas de información a través de bases de datos que contienen grados y correos electrónicos de profesores reguladas por la Dirección General Asuntos de Personal Académico, DGAPA; la Guía de carreras que publica anualmente la Dirección General de Orientación y Servicios Educativos, DGOSE, y la alimentación a

sistemas como el ARIES (Acervo de Recursos de Investigación en Educación Superior), el SAI (Sistema de Acopio de Información) y la Memoria UNAM, entre otros.

Debido a la estrecha relación entre DIES, la Dirección y la Secretaría General, las situaciones de contingencia son apoyadas por el Departamento para dar repuesta oportuna a las entidades solicitantes de información estadística relacionadas con sus actividades.

Otras actividades en donde participa el DIES son la coordinación y elaboración de la documentación relativa a las elecciones que se realizan en la Facultad, tales son los casos de Consejo Técnico, Consejo Universitario, Consejo Académico de Área, Comisiones Dictaminadoras, etc. También interviene en la organización de actos y ceremonias como la entrega de reconocimientos a docentes por su trayectoria en la UNAM, que es parte de la celebración anual del día del maestro, conferencias y homenajes a personajes distinguidos de la comunidad de ingenieros y apoyo en eventos como la Semana SEFI.

El conjunto de procesos y procedimientos del DIES, así como sus sistemas de información interna, permiten generar informes y estadísticas para elaborar proyectos tales como el Informe Anual de Actividades del Director de la Facultad, el anexo estadístico de la Memoria UNAM y la actualización de los principales indicadores de desempeño de la Facultad de Ingeniería para generar el anteproyecto de presupuesto institucional que coordina la Dirección General de Presupuesto, DGP. Existe además una importante participación del DIES en la respuesta a solicitudes de información general enmarcadas en el Acuerdo para la transparencia y acceso a la información en la UNAM, que conduce la Dirección General de Control e Informática – Patronato Universitario; atiende solicitudes de información estadística por entidades externas a la UNAM, tales como la Asociación Nacional de Escuelas y Facultades de Ingeniería, ANFEI y por el Consejo de Acreditación de la Enseñanza de la Ingeniería, CACEI, en apoyo a los procesos de reacreditación de las carreras que se imparten en la Facultad.

Si bien algunas de las tareas del DIES aún se realizan de manera tradicional, éste se esfuerza de manera cotidiana en la evolución de sus procesos como es el caso del sistema que da razón a este trabajo.

El Directorio de Contactos de la Facultad de Ingeniería

Como se ha mencionado, entre las diversas tareas que se realizan en el DIES se encuentra la realización y actualización del Directorio de Contactos de la FI el cual tiene como objetivo mantener los datos actualizados del personal docente, funcionarios, representantes académicos y estudiantiles así como proveer de información a otras entidades demandantes como por ejemplo el Directorio UNAM.

Los datos de contacto contenidos en el Directorio de la FI también son útiles para el Sistema de Auditoría Interna, la emisión de diversas convocatorias, la organización de la ceremonia de entrega de Reconocimientos al Personal Docente y para el envío de correspondencia dentro y fuera del campus universitario, entre otros.

La importancia de la existencia de esta herramienta es vital para el abastecimiento e integridad de la información que se procesa cotidianamente para la labor académica y administrativa de la Facultad de Ingeniería. Sin embargo su actual desempeño no es el más eficiente y por tal motivo el DIES se ha propuesto su mejoramiento por medio del uso recursos informáticos en cada tarea que se realiza para su obtención.

El futuro propuesto por el DIES para el Directorio de Contactos de la FI augura un sistema con información más oportuna, veraz, confiable y sobre todo rápido y eficiente.

Descripción actual del tratamiento de la información del Directorio del Personal, Funcionarios y Representantes Académicos

El mantenimiento del Directorio del Personal Académico de la Facultad de Ingeniería involucra actualmente varias actividades de carácter administrativo y se lleva a cabo por medio del trabajo que realizan aproximadamente doce personas: dos pertenecientes al DIES y una de cada División de la Facultad. Los responsables de proporcionar la información en cada área son las Divisiones Académicas, Coordinaciones y Secretarías.

Los datos que dicho directorio maneja actualmente son: nombre y apellidos, grado, nombramiento, teléfonos, fax y extensiones de la oficina, área de adscripción, teléfono y fax del área de adscripción.

El procedimiento para la elaboración y actualización del Directorio de la Facultad comienza con la elaboración del formato de captura por el DIES que él mismo distribuye en las diversas áreas, el llenado de los formatos, la recuperación de la información por el Departamento da como resultado la integración de la información y la homologación de formatos de donde se obtienen un aproximado de 600 registros para posteriormente publicar su edición de manera electrónica.

El tiempo estimado para cada una de las actividades mencionadas se distribuye de la siguiente manera:

Actividad	Tiempo en días*
Elaboración del formato	2
Distribución de los formatos	2
Distribución interna en cada División	2
Captura de la información	15
Captación de formatos al interior de las divisiones	2
Procesamiento de la información dentro de las divisiones	2
Recuperación de formatos por parte del DIES	1
Integración de la información	2
Publicación en diferentes medios	5
TOTAL	35

*Todos los días son hábiles de tiempo completo dedicados especialmente a esta actividad.

Tabla 1.1 Actual tratamiento de la información

La frecuencia de actualización de la información puede ser semestral o anual pero en realidad no existe un plazo, ésta puede llevarse a cabo en cuanto se tiene un dato reciente o se hace la actualización total de los datos en la manera descrita anteriormente.

Necesidad de un Sistema para Automatizar el Procedimiento de Realización y Actualización del Directorio

La importancia de ser incluido en un directorio de contactos radica entre estar informado y no estarlo, además ser una herramienta de gran ayuda para la administración de información del personal.

Sin duda, el tener información veraz, de calidad y a tiempo, es un ideal para mantener la comunicación entre dos o más entidades, sin embargo cuando existen múltiples papeleos de por medio, procedimientos administrativos en cada organismo y la información pasa por muchas manos, la consecuencia que se tiene es la falta de calidad en la información pues ésta se expone la falta de veracidad y el paso de los diversos procedimientos la hace menos confiable.

Como consecuencia de todas estas ineficiencias en donde se ponen en riesgo los objetivos de la existencia de un Directorio se vuelve absolutamente necesario un sistema automatizado que traiga consigo un menor número de procedimientos para la elaboración, gestión y actualización del mismo.

La inclusión de un sistema automatizado en esta labor significa un menor papeleo, fácil gestión para la obtención de datos, menor tiempo de procesamiento dedicado comparado con el que se utiliza en el método de elaboración tradicional.

Entre las características que debe cumplir este directorio están: descentralizar la administración de la información, delegando a cada División, mediante un ambiente Web; la responsabilidad de los datos de su personal, con la validación del producto final a cargo del DIES; así, las solicitudes que pudieran hacerse con respecto a la información que se desprende de nuestro sistema se atenderán en el momento por la persona solicitante y con ello se pueden agilizar de una manera sustancial las actividades del Departamento de la misma forma como las de las entidades solicitantes al tener una respuesta casi instantánea de manera que se incrementa la calidad de la información al hacerla confiable, eficiente, veraz y oportuna.

El sistema que se busca tiene la intención de nutrirse de información auténtica y confiable funcionando en una manera ideal como lo muestra este diagrama (Figura 1.2):

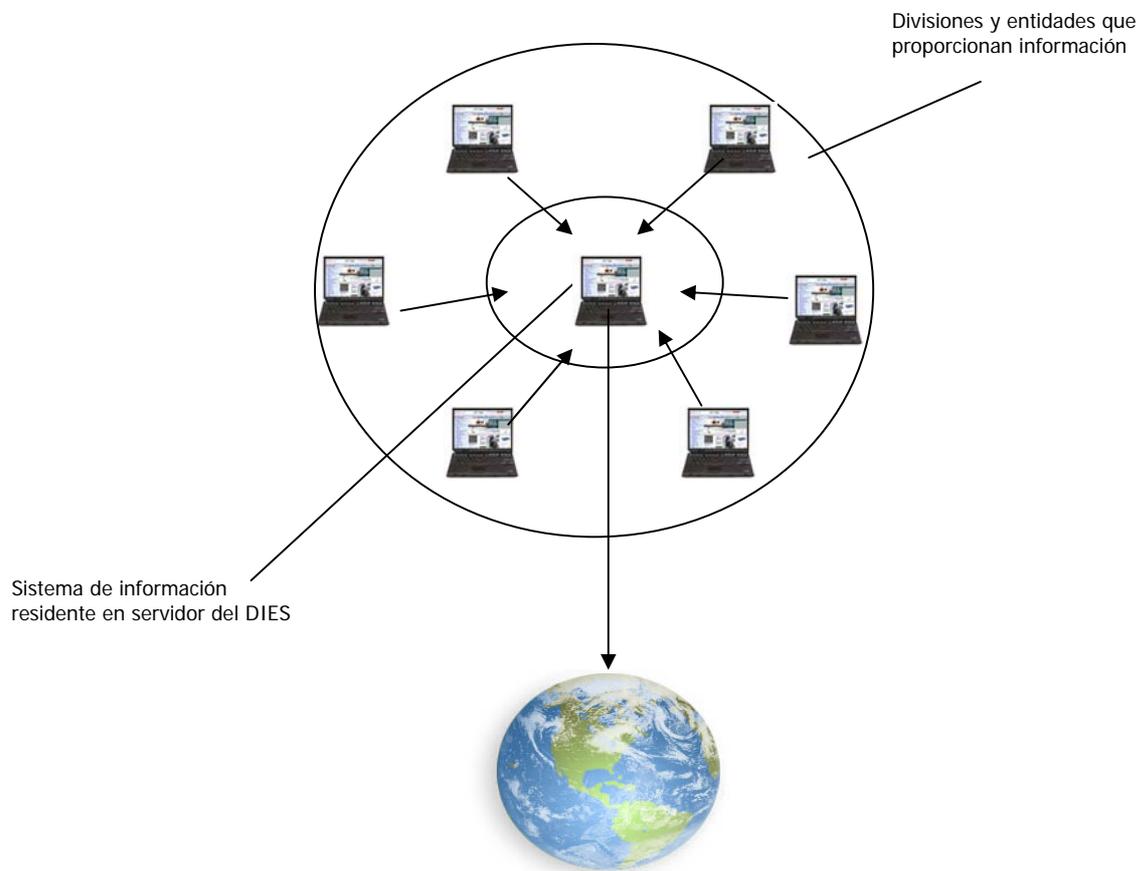


Figura 1.2 Funcionamiento del sistema

Capítulo 2 Marco Teórico Conceptual

Introducción

La modernización de los procedimientos en diferentes áreas sin importar la disciplina a la que se dedique una entidad es sin duda una constante que forma parte de los planes de desarrollo de las mismas. La competitividad y las mejores prácticas exigen sistemas automatizados que ayuden a desarrollar mejor las tareas y contribuyan en la evolución para poder tener una inserción en el mundo actual. Sin duda una institución que se negara a este tipo de cambios en su administración tendría una absoluta tendencia a quedar obsoleta e incluso a desaparecer.

Un sistema de cómputo trae consigo ventajas que serían difíciles de conseguir si se realizarán las actividades a las que esta enfocado de manera tradicional, es decir que la manera más veloz para procesar información, administrarla y compartirla sería sólo posible si se emplearán más personas o se dedicarán más horas hombre para llevar acabo ciertas tareas, la confiabilidad de los datos sería cuestionable pues estarían expuestos a errores humanos. Esto no quiere decir que un sistema automatizado sea 100% confiable pues la funcionalidad de éste depende de energía, recursos electrónicos, mantenimiento y actualización del mismo, elementos que también han de adaptarse a las nuevas tendencias.

2.1 La ingeniería de software

Equivocadamente se piensa que la actividad básica de la Ingeniería de Software es la programación. La evolución y complejidad de las exigencias de los sistemas de hoy en día han convertido al diseño de requisitos y la planeación en las bases fundamentales de esta rama de la Ingeniería. La importancia de la etapa de diseño radica en el costo y la dificultad de hacer cambios en etapas posteriores.

En general existen cuatro actividades fundamentales comunes para cualquier proceso de software: definir las especificaciones, desarrollar y validar la aplicación y monitorear su evolución. La manera en que se aborde un problema de ingeniería de software depende de cuanto tiempo se le dedique a cada etapa; a cada forma de crear estas aplicaciones se le conoce como Modelo de Proceso de Software.

El modelo de procesos de software es una guía para estimar costos, políticas y procedimientos que garantizarán la calidad de la herramienta desarrollada, especifica las actividades que son parte del proceso de software, las tareas de los integrantes del equipo de desarrollo, etc.

Hay dos modelos o paradigmas generales para desarrollar software, por métodos estructurados u orientados a objetos. Su principal diferencia es la elección de bloques con los que construyen las aplicaciones: los métodos estructurados eligen funciones y subprogramas y los orientados a objetos basan su construcción en objetos, por otra parte, el paradigma estructural enfoca su análisis en modular conforme a tareas y procesos orientados a objetos, examina el dominio del problema como un conjunto de objetos que interactúan entre si.

Los modelos de procesos orientados a objetos postulan la división de un programa en unidades autocontenidas llamadas objetos. En contraste con los métodos estructurados, los objetos contienen tanto los datos como sus definiciones y los algoritmos que los manipularán.

Los objetos se organizan en jerarquías, con lo que un objeto puede heredar datos y algoritmos de otro objeto lo que hace que la organización del proyecto sea más modular.

Entre los modelos incluidos dentro del enfoque orientado a objetos está el Método de Ensamblaje de Componentes (Figura 2.1), que incorpora características de métodos evolutivos como el Espiral ya que exige interactividad de usuarios y del equipo de desarrollo; enfatiza la reutilización de clases y la creación de ellas si es necesario.

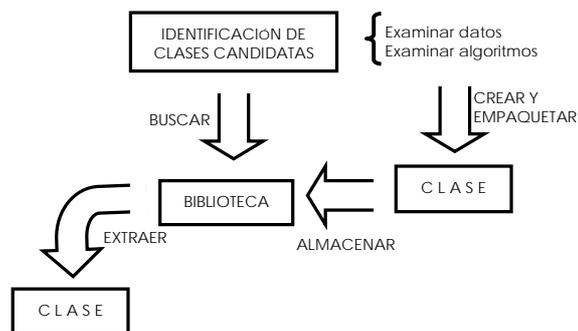


Figura 2.1 Método de Ensamblaje de Componentes-
Sucesión de pasos para cada etapa de la espiral

Cada ciclo del método tiene el objetivo de completar las clases necesarias para su cumplimiento, ya sea creándolas o buscándolas en bibliotecas ya existentes por causa de este mismo proyecto u otro anterior. Y así sucesivamente hasta la iteración de ensamblado del programa.

El Método de Desarrollo Concurrente se representa en forma de esquemas, las actividades dentro de él ocurren en el mismo tiempo pero en diferentes estados. Es usado en el desarrollo de aplicaciones para sistemas con arquitectura cliente/servidor dados los múltiples componentes que interactúan en este tipo de sistemas (Figura 2.2).

En esta técnica se define una red de actividades en lugar de una secuencia de las mismas, donde los componentes del sistema se pueden diseñar y desarrollar concurrentemente. Las actividades se definen en dos dimensiones: las actividades de sistema y las actividades que competen a la realización de los componentes. Las actividades del sistema se centran en el diseño, embalaje y uso del mismo, mientras que las de los componentes se enfocan en el diseño y la realización de éstos.

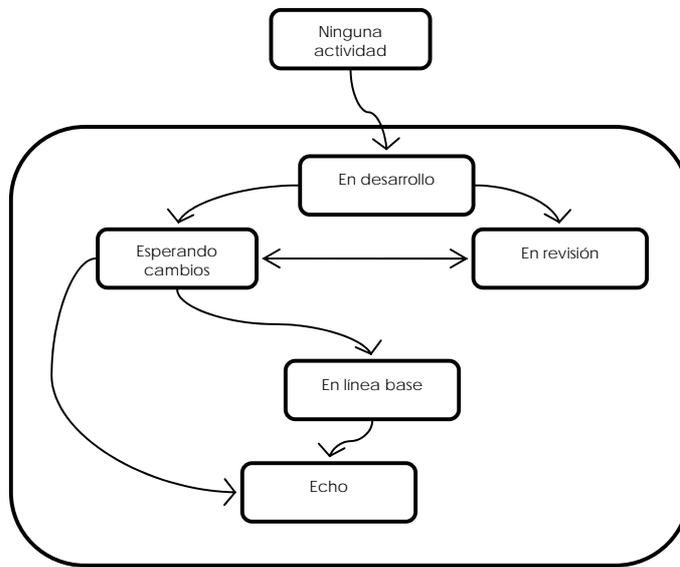


Figura 2.2 Modelo de Desarrollo Concurrente para la creación de sistemas

Cada representa un estado de una actividad en el proceso de desarrollo de la aplicación.

Por otra parte están los métodos procedimentales, que han sido estudiados y aprobados a lo largo del tiempo, puesto que han ayudado a implementar muchos sistemas anteriormente.

El proceso lineal por excelencia y tradición es el llamado Método de la Cascada o lineal secuencial, ejemplificado en la Figura 2.3. Con este método se desarrollan sistemas de gran complejidad dada su principal característica: se debe finalizar completamente la etapa anterior para pasar a la siguiente. Consiste en una sucesión de actividades en donde cada fase requiere que la información de entrada, los procesos y los resultados estén bien definidos.

Las etapas que constituyen el método son:

Planificación: se determinan la factibilidad y costos del proyecto, las necesidades que cubrirá el sistema y sus restricciones.

Análisis: se reconoce claramente el dominio de la aplicación que se desarrollará, sus entradas y salidas y cómo interactúan con el resto del entorno.

Diseño: enfocado en la estructura de datos, la arquitectura del software, detalle procedimental y la caracterización de la interfaz antes de codificar.

Codificación: traducir para la máquina la solución a las necesidades plantadas en el análisis.

Prueba: realizando ensayos se asegura que se produzcan los resultados esperados bajo entradas establecidas.

Mantenimiento: procura el óptimo funcionamiento y prepara el software para futuras necesidades.

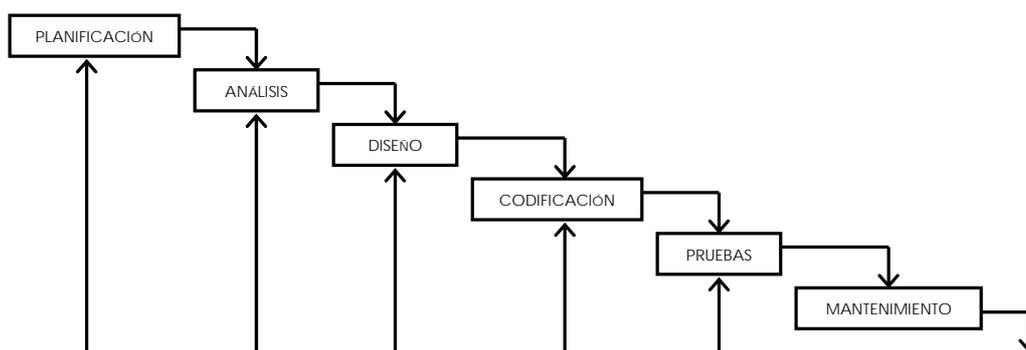


Figura 2.3 Método de cascada para construcción de sistemas

Dentro de los métodos evolutivos está el Modelo de Construcción de Prototipos (Figura 2.4), que en su etapa inicial se centra en conocer los objetivos globales del software que el equipo de desarrollo toma para crear una aproximación a la aplicación final, dando énfasis a la interfaz entre el usuario y el sistema. Luego de ser aceptado el prototipo, se robustecen sus otros componentes para crear una herramienta completamente funcional.

Las etapas con las que se desarrolla mediante la Construcción de Prototipos son:

Análisis y determinación de requerimientos, Diseño del prototipo con el usuario, Implementación del prototipo, Pruebas con el usuario, Refinación del prototipo, Pruebas del software terminado, Solución de fallas y optimizaciones y finalmente la Entrega e Implementación del software.

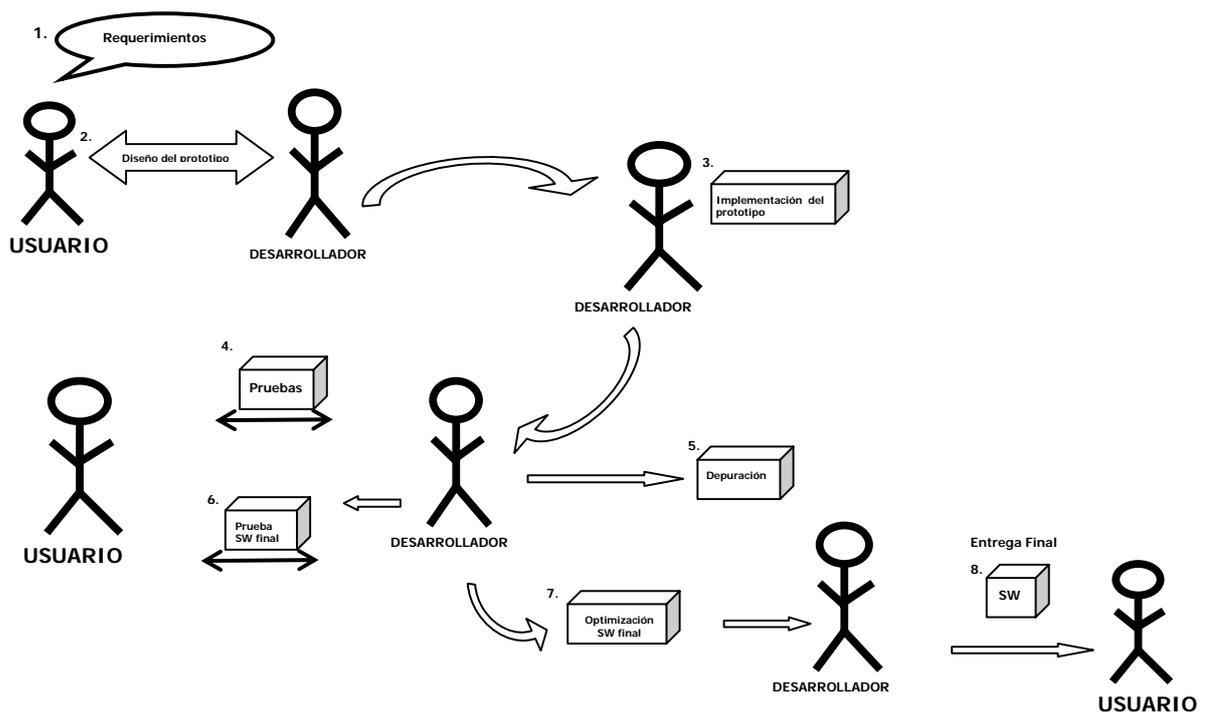


Figura 2.4 Construcción de prototipos

Otro modelo evolutivo es el Modelo Incremental, que es una adaptación entre el modelo de construcción de prototipos y la metodología en cascada (Figura 2.5).

Ingeniería: Desarrollo del producto que corresponde a la etapa en proceso.

Evaluación: El cliente evalúa los resultados entregados por la etapa de ingeniería.



Figura 2.6 Modelo Espiral

Cada uno de los métodos que existen para el desarrollo de sistemas se aplica según el estudio del equipo de desarrollo acorde con las características que debe tener el software, el tiempo de desarrollo, presupuestos, etc. En los siguientes párrafos se presentan algunas de las desventajas de los métodos expuestos anteriormente.

Orientados a Objetos: Romper con el paradigma de la programación estructurada suele ser complicado para programadores con gran experiencia en modelos procedimentales. Además se debe tener un conocimiento amplio y completo desde el arranque del proyecto, lo que se complica en la determinación de riesgos, y si no se tiene una planeación sólida se puede perder el objetivo del proyecto.

Cascada: En la vida real son pocos los proyectos que siguen con rigor el flujo secuencial que expone este método, además de que es complicado tomar en cuenta la incertidumbre inherente que hay al iniciar un proyecto nuevo.

Construcción de Prototipos: Al observar prontamente una aproximación visual del sistema final, el usuario puede exigir modificaciones mínimas para resolver el problema de manera rápida lo que impediría el desarrollo en los otros principios del sistema.

Incremental: Determinar con precisión en cada una de las iteraciones objetivos, metas y plazos suele ser rutinario y tedioso lo que puede provocar faltar de atención en el análisis.

2.2 Redes de Computadoras

¿Qué es una red de computadoras?

Se puede definir a una red de computadoras como un sistema de comunicación entre computadoras, es decir, una colección de concentradores, computadoras, impresoras, y otros dispositivos (electrónicos) conectados por un medio físico con el fin primordial de intercambiar información y compartir recursos como periféricos y unidades de almacenamiento.

Así mismo, las redes de computadoras involucran tanto hardware, anteriormente enumerado, como el software requerido para su administración y algunas aplicaciones que exploten sus recursos, mecanismos de control para la transmisión sobre el medio y protocolos que gobiernen las relaciones entre equipos.

El uso de las redes de computadoras no sólo por las máquinas mismas, es una herramienta para incrementar la producción en el desarrollo de actividades propias de empresas, por ello, se debe tomar en cuenta el factor humano dentro de los diseños, pues una red es más que dispositivos electrónicos interconectados con un medio de transmisión, también es constituida por los usuarios que administran y hacen uso de la información a través de este sistema de comunicación.

Objetivos y ventajas de las redes de computadoras

Entre sus objetivos están el mantener los recursos de la red disponibles para quien los solicite sin importar su ubicación física, disminuir costos al compartir periféricos y unidades de almacenamiento, mantener bases de datos actualizadas y accesibles desde diferentes ubicaciones y facilitar la transferencia de grandes cantidades de información y la comunicación en general como el correo electrónico o la mensajería en tiempo real.

Hoy en día sus alcances han llegado a las videoconferencias, redes de dispositivos portátiles, puntos de venta y tecnologías de información en general, ofreciendo ventajas como ahorro en los costos comparando la relación costo/rendimiento entre las computadoras "pequeñas" y los equipos más poderosos; acceso y ejecución remota de aplicaciones no residentes en todos los elementos de la red; disponibilidad de datos al tener varias fuentes

alternativas duplicando archivos en diferentes máquinas, asegurar el funcionamiento de la red -según la topología, ya que si una computadora falla, el resto de la red puede asumir sus labores sacrificando en parte su rendimiento global.

Elementos de la red

En las redes "el enlace entre las maquinas se realiza primero a través de un medio físico y posteriormente a través de un medio llamado protocolo", con anterioridad se enunciaron algunos elementos de las redes de computadoras, ahora se tratará de detallar más acerca de ellos.

Hardware

Servidor: Es el hardware central de una red de computadoras, es la máquina central donde ejecuta el Sistema Operativo de Red, NOS - por sus siglas en inglés - proporciona servicio a los nodos que lo requieran. Existen 2¹ tipos de servidores: dedicados y no dedicados con arquitectura cliente/servidor o con conexión punto a punto y se pueden configurar los que sean necesarios de acuerdo a los requerimientos de cada red, haciendo con ello más o menos eficiente la red.

Un servidor dedicado, como su nombre lo dice, es aquel que no puede ejecutar otro trabajo más que para el que es requerido, no puede utilizarse como estación de trabajo y generalmente utiliza una versión de sistema operativo que optimiza la velocidad en el intercambio de información. Por otro lado, un servidor no dedicado es el que funciona además como estación de trabajo, comparte sus recursos con el resto de la red como bases de datos o periféricos.

Estaciones de trabajo: Son las computadoras "clientes" en la red, ya sea que la configuración de ellas esté como cliente/servidor o *peer to peer*, reciben del servidor respuesta a sus peticiones de información o recursos como impresión y acceso a bases de datos. Son capaces de aprovechar los recursos de otras computadoras, pero no pueden compartir sus recursos, puesto que esto las convertiría en servidores.

¹ Todo acerca de las redes de computadoras, Stoltz, pág. 34

Tarjetas de Interfaz de Red (NIC): Son los dispositivos electrónicos pertenecientes al servidor o las estaciones de trabajo encargados de la transmisión física de los datos, interactúan con el medio de comunicación para efectuar la transferencia de la información dentro de la red codificando y decodificando paquetes de datos que se envían siguiendo un protocolo de transferencia. La topología a utilizar en la red de computadoras es determinada por el tipo de adaptador de red.

Medio de transmisión: Es por el cual se transmiten físicamente los datos en forma de señales electromagnéticas – impulsos electrónicos, microondas, señales de radio, satelitales, etc. Existen medios alámbricos, como cable UTP, cable telefónico, cable coaxial, fibra óptica; y medios inalámbricos como las microondas, infrarrojo u ondas de radio.

Concentradores: Son los elementos que concentran las señales de los nodos de la red, reparten su ancho de banda, identifican mediante algoritmos de encaminamiento la mejor ruta dentro de la red para enviar los paquetes de información. Existen diferentes tipos y se dividen de acuerdo a sus capacidades. Se utilizan para extender el área de cobertura de la red y los hay tanto para la transmisión cableada como inalámbrica.

El hub o repetidor es un dispositivo que amplifica y retransmite señales por lo que su función es sólo extender la longitud de la red. Los bridges o puentes revisan la dirección asociada a cada paquete de información y la envía a su correspondiente segmento de la red que enlaza el puente. Se utilizan para reducir el tráfico de paquetes en la red. El router o ruteador opera a un nivel diferente que los puentes y permite establecer funciones más complejas que él ya que trabaja en conjunto con el NOS, como el envío de paquetes de información determinando la ruta más eficiente y los gateways o compuertas permite la comunicación entre tipos diferentes de redes de computadoras.

Software

Sistema operativo de red: El sistema operativo de red o NOS es el programa a través del cuál el servidor puede administrar los recursos y el flujo de datos que corre por la red, ya sea que esté configurada con una arquitectura cliente/servidor o entre pares. Habilita la generación de conexiones lógicas a voluntad y necesidad del administrador de red sin que esto requiera cambiar la disposición física del hardware, determina las características de red

disponibles y sus capacidades, y permite configurar los nodos para que ejecuten funciones deseadas.

Las características esenciales de los NOS son: multiusuario, multiprocesos y multitarea, suelen tener herramientas de respaldo en unidades de almacenamiento masivo y opciones de restricciones de seguridad.

Clasificación

De acuerdo a su alcance geográfico, las redes de computadoras se dividen en Redes de Área Local, LAN, de 0.1 a 25 kilómetros y de Cobertura Amplia, WAN, decenas o cientos de kilómetros.

Las LAN por lo general son ubicadas dentro de un edificio u oficina, suelen tener un número de nodos conectados entre 2 y 100. Por otro lado, las WAN (Wide Area Network) podrían ser consideradas como dos o más redes LAN interconectadas, emplean líneas telefónicas u otro medio de transmisión a larga distancia, fibra óptica por ejemplo.

Otra manera de clasificar las redes de computadora es por la forma en la que están conectadas, es decir su Topología (definida por el hardware con el que se conecta).

La topología consiste en la organización que se toma al conectar el conjunto de elementos de los que se componen la red. Una red tiene dos diferentes topologías, una física y una lógica: la primera se refiere a la ubicación física de sus elementos, mientras que la segunda determina las rutas que toman los datos para llegar a los nodos, es decir el método que usa para comunicarse.

Bus: En esta topología cada nodo está conectado a un segmento común de cable que recorre todas las máquinas sin formar caminos cerrados o bifurcaciones. Para mantener la impedancia constante en el cableado de la red, se deben conectar dos terminadores en ambos extremos del cableado de la misma.

Anillo: En este caso, las líneas de comunicación forman un camino cerrado. Generalmente los paquetes de información recorren el anillo en forma unidireccional, cada máquina recibe la

información de la máquina previa, la analiza, y si no es para ella, la retransmite a la siguiente máquina.

Estrella: En ésta hay un concentrador al centro, en el que convergen todas las líneas de comunicación. Cada máquina tiene un enlace exclusivo con el concentrador. A esta topología generalmente se asocia el estándar Ethernet 802.x

La topología empleada en una red depende tanto de sus necesidades como de la distribución de los nodos, las aplicaciones que se ejecutarán, capacidad de expansión, el tráfico de información y la valoración de sus costos. La topología en bus es fácil de instalar y mantener, sin embargo, al igual que el anillo ofrece un problema importante: si el bus se abre toda la red quedará inoperable, en comparación con una red en estrella, si un nodo se “cae” esto no afectará en el desempeño de la red.

Por otro lado, al conjunto de capas y protocolos se le denomina arquitectura de red. Por la relación que hay entre sus miembros, las redes se subdividen en dos grandes grupos: las redes con servicios centralizados y descentralizados.

Las redes con servicios centralizados tienen como arquitectura base una de cliente/servidor, en ésta existe uno más servidores dedicados que comparten sus recursos con el resto de la red, dispuestos a satisfacer las peticiones de cada cliente o estación de trabajo que tenga acceso a él. Todos los recursos se centralizan en estas máquinas y las estaciones de trabajo sólo pueden usar sus propios recursos o los de los servidores. En este tipo de redes, las aplicaciones se parten entre el servidor y las estaciones de trabajo. En el Front End, la parte cliente de la aplicación acepta las peticiones del usuario, las prepara para el servidor y espera una respuesta del mismo. En el Back End, el servidor recibe la petición del cliente, la procesa y proporciona el servicio deseado por él, que presentará los datos u otro resultado al usuario a través de su propia interfaz.

Las redes con servicios descentralizados trabajan bajo la arquitectura *peer to peer* o punto a punto. En ella se permite a cada estación de trabajo funcionar como servidor no dedicado de los recursos con los que cuenta, cada computadora administra y da acceso a sus fuentes de información o periféricos, proporcionando mayor flexibilidad que una con arquitectura

cliente/servidor. Las computadoras que no ofrecen recursos son solamente clientes y las que lo hacen se llaman anfitrión/cliente o host/client.

Cada arquitectura tiene sus ventajas y desventajas, es por ello que no se puede decir cual de las dos es más eficiente, porque esto depende de las necesidades de la misma red.

Entre las ventajas de la arquitectura cliente/servidor están mayor rendimiento con servidores dedicados, gran seguridad contra pérdidas de información y el acceso a la información por ser centralizada, es de fácil administración, practicidad en actualización de software y respaldo de información; pueden existir terminales ejecutando sistemas operativos diferentes y si una el sistema de una terminal se “cae” la red no se verá afectada. Sus desventajas se encuentran en que se debe adquirir un equipo poderoso para el servidor, no se pueden compartir recursos en los nodos de red y la principal es que si el servidor falla se detendrá la actividad en la red.

En contraparte, las ventajas que tiene la arquitectura punto a punto son que el sistema operativo tiene un costo menor, es más fácil reconfigurar la red, es más económica porque no se deben utilizar equipos poderosos como servidores y los mismos servidores no dedicados pueden ser usados también como estaciones de trabajo, es adaptable para distribuir aplicaciones entre varios servidores y con ello generar mejor rendimiento sin aumentar los costos y, sobre todo es efectivamente flexible cuando se trata de acceder a diferentes nodos de la red. Sin embargo, su gran flexibilidad se compensa con la dificultad en su administración y carencia en la seguridad general, lo que presume una gran desventaja. Los servidores no dedicados son más lentos que los dedicados y requieren más memoria RAM que una estación de trabajo, y aunque el rendimiento del servidor se optimiza con el uso de uno dedicado, el rendimiento general de la red aumenta conforme la distribución de las tareas de la red entre varios servidores.

En general las redes importantes funcionan bajo una arquitectura de servidores dedicados.

Transferencia de datos

Estándares y protocolos

Ya que la arquitectura de una red de computadoras también comprende los protocolos con los que se manejan sus dispositivos, en este tema se describen los principales estándares desarrollados para las redes y su ocupación en Internet.

Los estándares se crean para que los equipos que componen una red, independientemente del fabricante puedan interactuar de manera correcta; definen la forma de conectar los dispositivos de hardware y el protocolo a usar para establecer la comunicación entre ellos y están respaldados por organizaciones dedicadas a ello como ANSI, IEEE, CCIT e ISO entre otras.

Ya que cada dispositivo trabaja individualmente y un software especial lo controla mediante procesos, al colocarlos en una red, estos dispositivos tienen que trabajar en conjunto intercambiando mensajes para coordinar sus acciones y lograr sincronía. Los protocolos describen el intercambio de información entre procesos, es decir, programas que se ejecutan en hardware o aplicaciones. Permiten entonces iniciar, mantener y terminar un diálogo entre los elementos de un sistema.

Sus funciones más importantes son: el control de errores, controlar el flujo y congestión, regir las estrategias de encaminamiento, también incluye el formato de envío de la información y las especificaciones físicas como su temporización.

OSI (Open System Interconnection) de ISO

OSI es un estándar internacional que establece los lineamientos que deben seguir hardware y software para establecer comunicación.

En el modelo OSI los protocolos son pilas de bloques donde, en cada capa se especifica una función de la comunicación. Cada bloque se comunica con su homólogo en la máquina con la que establece la transmisión de información. Este modelo está compuesto por siete capas o niveles.

Capa 7.- La capa de aplicación administra servicios como la transferencia de archivos (FTP), la transferencia de páginas de hipertexto (HTTP) y el sistema de archivos de red (NFS) entre otros. En ella se definen programas de aplicación que utilizan la red para la comunicación entre diferentes sistemas.

Capa 6.- La capa de presentación provee de una interfaz para realizar la transferencia, estandariza la presentación de los datos a las aplicaciones. Aquí se gestiona la encriptación y se realiza la compresión de los códigos incluidos en los datos enviados.

Capa 5.- Es llamada capa de sesión porque administra las sesiones entre aplicaciones. Algunas de sus funciones son: coordinar el intercambio de datos controlando la integridad y el flujo de información en ambos sentidos, establece y libera la conexión de sesión y sincroniza y administra la sesión.

Capa 4.- La capa de transporte permite el enlace entre el host y la red. Regula el control de flujo del tráfico de punta a punta y reconoce los paquetes duplicados. Entre los protocolos que trabajan sobre este nivel se encuentran el Protocolo de Transporte (TP), el protocolo de intercambio de paquetes en secuencia (SXP) y el Protocolo de Control de Transporte (TCP).

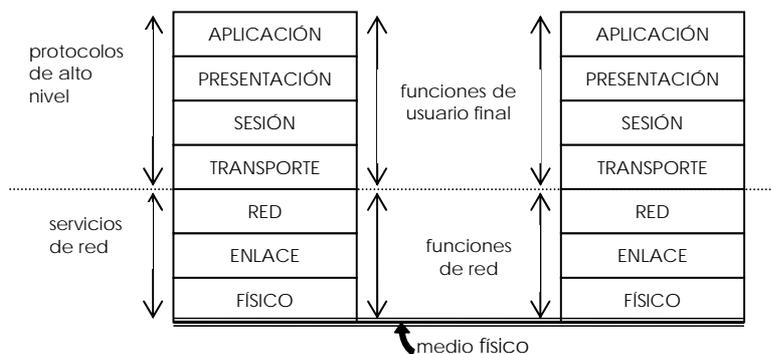


Figura 2.7 Capas del Modelo OSI

Capa 3.- Es la capa de red y se encarga del enrutamiento de la información mirando las direcciones de los paquetes y realizando el control de congestión. Asume la administración de las conexiones a través de la red para las capas superiores. Los estándares que efectúan alguna función incluyen el protocolo de intercambio de paquetes entre redes (IPX), el protocolo de Internet (IP), y el protocolo de entrega de datagramas (DDP).

Capa 2.- La capa de enlace de datos provee de una entrega confiable de información a través del medio físico. Detecta y controla los errores mediante CRC, tiene bajo control la secuencia, el flujo, el enlace lógico, el acceso al medio y la sincronización de la trama a nivel de datos.

Capa 1.- Esta capa define las características físicas (los tipos y especificaciones del cable), eléctricas (niveles de tensión) y funcionales (establecimiento, mantenimiento y liberación del enlace físico) del medio de transmisión; dependiendo de la topología adoptada es el estándar que trabaja bajo esta capa (IEEE 802.x). La capa física sólo reconoce bits individuales, no reconoce caracteres ni tramas.

Las capas de aplicación, presentación y sesión son llamadas capas de host pues dependen de la máquina anfitriona (host), mientras que las capas red, enlace y física están más orientadas hacia la comunicación.

Nivel	Algunos protocolos asociados	Elementos de conexión
Aplicación	NFS, Shell, Redirector	
Presentación	RFS, SMB, NCP, NFS	
Sesión	TCP, IPX, NetBIOS, FTP	
Transporte	TCP, UDP, SPX/IPX	Gateway
Red	IP, IPX, ICMP	Router
Enlace	CSMA/CD,	Switch
Físico	IEEE 802.3, IEEE 802.4, RS-449	Hub, NIC, Cableado

Tabla 2.1 Capas OSI y protocolos asociados

Relaciones con el Modelo OSI

TCP/IP, Transmission Control Protocol / Internet Protocol

TCP/IP es la base de Internet y sirve para enlazar computadoras que utilizan diferentes sistemas operativos, fue desarrollado por el Departamento de la Defensa de Estados Unidos en 1972.

Generalmente en el modelo OSI se le ubica en la capa de transporte (TCP) y en la capa de red (IP) sin embargo TCP/IP no encaja perfectamente dentro de OSI porque no están delimitados con precisión.

Aunque no de forma generalizada, TCP/IP también es concebido como un modelo en capas, pero cuando es así, se presentan de 3 a 5 niveles funcionales en la arquitectura del protocolo y de la misma manera que OSI, los datos van hacia abajo cuando son enviados y hacia arriba de la pila al recibirlos.

4	Aplicación	Proporcionan servicios que trabajan directamente con las aplicaciones de usuario, incluyen protocolos como HTTP, FTP, SSH y DNS.
3	Transporte (host to host)	Los protocolos que trabajan en esta capa resuelven problemas de fiabilidad y aseguran que los paquetes lleguen en orden y forma.
2	Interred	Se enrutan los paquetes a través de Internet. Aquí IP realiza las tareas básicas para lograr llevar datos de un origen hasta un destino.
1	Acceso a la red	En esta capa se dan las especificaciones del transporte de los paquetes sobre el nivel físico.

Tabla 2.2 Capas TCP/IP

Características de TCP/IP

Este protocolo es un conjunto de aproximadamente 100 estándares, entre sus principales están TCP e IP, está diseñado para enrutar además de tener alta fiabilidad, es adecuado para redes grandes y medianas. Se utiliza a nivel mundial para conectarse a Internet y a los servidores web pero es más difícil de configurar que NetBEUI o IPX/SPX.

TCP es un mecanismo de transporte fiable y orientado a conexión, asegura que los datos lleguen completos, sin daños y en orden. UDP (User Datagram Protocol) al igual que IP no verifica que los paquetes lleguen a su destino y no garantiza que lleguen en orden. Estos están orientados a transferencias de video o audio, en donde es más importante el tiempo de llegada del paquete que su fiabilidad. UDP Y TCP son utilizados para dar servicio a aplicaciones de alto nivel.

El conjunto TCP/IP se utiliza tanto en redes universitarias o empresariales, donde se utilizan enrutadores y conexiones a mainframes o a máquinas UNIX. La evolución de TCP/IP continua conforme lo demande la evolución misma de Internet, ya que Internet es disponible para cualquier usuario que disponga de una computadora y una conexión.

Seguridad en redes

El propósito fundamental de las redes de computadores es reducir costos compartiendo sus recursos: dispositivos, aplicaciones e información. Al facilitar los recursos la salvedad de ellos ya no sólo es responsabilidad de su dueño, ahora es de cada usuario, por ende, mientras más usuarios tiene una red, más compleja es su administración y su integridad es más vulnerable.

Errores humanos, robo de información, ataques de virus informáticos, intromisiones malintencionadas, fallas en los equipos y desastres naturales son los principales riesgos que corre cualquier sistema. Para determinar como fortalecer las debilidades de un sistema es necesario efectuar un análisis de riesgos y así tener una red o sistema lo menos inseguro posible.

Entre las medidas de seguridad que se toman para salvaguardar la integridad de los sistemas y las partes que lo componen se distinguen dos tipos:

Las físicas, como mantener los equipos en lugares acondicionados para tales efectos, contar con accesos controlados y restringidos, etc., y las lógicas integradas en los sistemas operativos o las aplicaciones que corren sobre ellos.

A nivel de redes de computadoras es su estructura administrativa quien provee estos mecanismos de seguridad como la autenticación, el cifrado de la información, la protección antivirus e intrusos, respaldos de seguridad, etc.

La literatura en torno a la seguridad en redes define los siguientes conceptos, según el texto "Administración de Redes" publicado por el Centro Universitario de Ciencias Económico-Administrativas de la Universidad de Guadalajara.

Identificación: Es la habilidad de saber quién es el usuario que solicita hacer uso del servicio.

Autenticación: Es la habilidad de probar que alguien es quien dice ser; prueba de identidad.

Control de Acceso: Una vez que se sabe y se puede probar que un usuario es quien asegura ser, el sistema decide lo que le permite hacer, dotando de permisos según identidad.

Confidencialidad: Es la protección de la información para que no pueda ser vista ni entendida por personal no autorizado.

Integridad: Es la cualidad que asegura que el mensaje es seguro, que no ha sido alterado. La integridad provee la detección del uso no autorizado de la información y de la red.

No repudiación: Es la prevención de la negación de que un mensaje ha sido enviado o recibido. Asegura que el remitente del mensaje no pueda negar que lo envió o que el receptor niegue haberlo recibido. La propiedad de no repudiación de un sistema de seguridad de redes de cómputo se basa en el uso de firmas digitales.

Firma Digital: Método para verificar el origen y el contenido de un documento electrónico. Se basa en la idea que si el texto de un documento se procesa con un algoritmo de encriptación, cualquier cambio en el documento original causará un cambio en la salida del proceso de encriptación.

Criptografía: Técnica que hace ininteligible la información cuando es transmitida, convirtiéndola en un texto cifrado. En el receptor se restaura el texto cifrado a la información

original o texto claro con el proceso de criptografía inverso. En un proceso de encriptación se usan una o más llaves en un algoritmo de cifrado. El texto codificado es inteligible para cualquier receptor sin el uso del algoritmo de encriptación y de la llave correcta para descryptar la información. Existen dos métodos de encriptación:

Simétrica o de llave privada: Conocida por el transmisor y por el receptor para encriptar y descryptar el mensaje. Asimétrica: Usa una llave pública para la encriptación y una llave privada para la descryptación del mensaje.

El hecho de disponer de redes de computadoras capaces de interconectarse no constituye el punto final de ésta. La seguridad informática va adquiriendo una importancia creciente con el aumento del volumen de información importante que se halla en las computadoras distribuidas.

2.3 Lenguajes de Programación

Las computadoras son herramientas que ayudan a resolver problemas con los que se enfrenta la humanidad. Sin embargo una computadora por si sola no consigue hacer nada, se debe de "acondicionar" para que brinde ayuda. Esto se logra desarrollando aplicaciones con fines específicos. Sin embargo, para crear estas aplicaciones es necesario un Lenguaje de Programación que sirva de enlace entre el mundo real y el de las computadoras.

El lenguaje de programación que se elija debe tener una documentación amplia y un buen soporte para el software ya que la decisión que se haga impactará en todo el proceso de creación de la aplicación.

Por la naturaleza del sistema del que es objeto este trabajo, es decir, una herramienta que ayude a automatizar en la medida de lo posible la gestión de información, que ofrezca una actualización oportuna además de manejar un número considerable datos y permita localizar a una persona dentro y fuera de la Facultad con diferentes criterios de búsqueda, etc., resulta obvio el utilizar la programación de páginas dinámicas.

Las páginas del servidor son reconocidas, interpretadas y ejecutadas en el mismo servidor, luego, es el servidor el que gestiona la información contenida en las bases de datos, conexiones en red u otros recursos, posteriormente envía al cliente una página codificada en HTML, compatible con todos los navegadores, con los resultados de todas las operaciones realizadas anteriormente.

Existen muchos y muy variados lenguajes concebidos para Internet que permiten crear páginas dinámicas; cada uno de ellos aprovechan características más a fondo y así se vuelven más o menos útiles para desarrollar diferentes aplicaciones. Además los alcances de cada lenguaje están relacionados directamente con su complejidad, por lo que mientras más tareas se puedan realizar, el aprendizaje del lenguaje será más complejo.

Entre los lenguajes que se ejecutan del lado del servidor más utilizados para desarrollar páginas dinámicas se encuentran ASP, PHP, JSP y Perl.

CGI (Common Gateway Interface) codificado en el lenguaje Perl, es el sistema más antiguo que existe para la codificación de páginas dinámicas. Por su parte, ASP (Active Server Pages) es un lenguaje derivado del Visual Basic desarrollado por Microsoft, se emplea sobre plataformas funcionando bajo sistema Windows NT. JSP (Java Server Pages) es una tecnología orientada a crear páginas web con programación en Java y PHP (Hipertext Preprocesor) se podría considerar como un lenguaje análogo a ASP utilizado en plataformas Unix y Linux que es distribución libre.

Estos lenguajes resultan bastante útiles para la explotación de bases de datos y su aprendizaje resulta realmente accesible. PERL por otro lado, es más rápido y potente, con gran facilidad para el manejo de grandes cantidades de datos y búsquedas de texto pero requiere un conocimiento más dedicado.

Ya que el servidor ejecuta y transforma los scripts en código HTML antes de enviarlos, estas tecnologías ofrecen una buena ventaja en el campo de la seguridad pues de esta manera se envía sólo información relacionada con la petición de la máquina cliente, además de ser independiente del navegador del cliente pues HTML, es fácilmente interpretado por muchos de ellos. Es obvio que los recursos del servidor deben ser por mucho más que los recursos del cliente. Se debe considerar, además la capacidad de soportar un número considerable de usuarios concurrentes ya que se requerirá más tiempo de procesamiento por cada uno.

A continuación se exponen las principales características de estos lenguajes de programación, para conocer sus alcances y con ello tener sustento en la elección del lenguaje a utilizar.

GCI

CGI fue uno de los primeros estándares para la programación de páginas dinámicas que se ejecutan en el servidor y normalmente se escriben en lenguaje Perl, aunque puede escribirse en C o Visual Basic. CGI especifica un estándar para transferir datos entre el cliente y la aplicación.

En general el funcionamiento de un programa CGI es el mismo que llevan a cabo otros lenguajes orientados a Internet con algunas particularidades: se realiza una petición, el servidor ejecuta los programas CGI y trabaja con los recursos necesarios para llevar a cabo las tareas, el programa CGI escribe en la salida estándar el resultado de la ejecución del CGI incluyendo en él etiquetas HTML, es decir, una página Web.

Perl es un acrónimo de Practical Extracting and Reporting Language y fue desarrollado por Larry Wall, en sus inicios fue creado para simplificar la administración de sistemas UNIX, sin embargo hoy es un popular lenguaje utilizado para la construcción de aplicaciones Web basadas en tecnología CGI y herramientas particularmente orientadas al tratamiento de textos y archivos.

No establece una filosofía de programación particular pues no es un lenguaje propiamente orientado a objetos, modular o estructurado aunque soporta todas estas formas de programación. Es un lenguaje interpretado: el código de los programas escritos en Perl no se compila sino que cada vez que se ejecuta el código se lee y ejecuta el intérprete.

Entre sus características principales están que es de distribución libre, bajo la filosofía GNU y se puede obtener para diferentes plataformas, es extensible a partir de otros lenguajes, ya que desde Perl se pueden hacer llamadas a subprogramas escritos en otros lenguajes y viceversa. Además es un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los archivos. Implementa las expresiones regulares que le da mayor potencia en el procesamiento de textos y, que a partir de su inclusión en Perl se adoptaron en otros lenguajes y plataformas como .NET o Java.

Una diferencia fundamental de Perl con relación a otros lenguajes similares es que no limita el tamaño de los datos, el límite lo pone la memoria disponible en el tiempo de ejecución. Esto puede suponer en cierto momento ineficiencia en el desempeño del lenguaje ya que cada una de las aplicaciones CGI que se corren lo hacen en un espacio de memoria propio; entonces, si tres usuarios corren un CGI a la vez se triplicarán los recursos utilizados.

ASP y Microsoft .NET

Desarrollado por Microsoft, .NET es un entorno de desarrollo para servicios Web de XML que permite a las aplicaciones comunicarse y compartir datos a través de Internet sin importar la plataforma o lenguaje de programación buscando que esta red se convierta en una fuente de servicios y no sólo de datos.

Internet ha sido un modelo de manejo de procesos e información centralizado, en donde el navegador accede a un sitio o completa un formulario, y el usuario sólo presiona un botón, permitiendo que los procesos se ejecuten en un servidor remoto para entregar la información en su PC.

Las aplicaciones distribuidas demandan costo y requerimientos de recursos para el despliegue y la administración de la aplicación. Por otro lado, en los servicios Web, para el usuario se vuelven mínimos el esfuerzo y el costo de manejo de la aplicación, ya que el uso del servicio es independiente del suministro del mismo.

En una arquitectura basada en los servicios, el usuario consume servicios de aplicación independientemente de su implementación técnica y estos servicios se pagan según el uso que se hace de ellos. Los servicios se implementan utilizando estándares abiertos y cualquiera puede utilizarlos si dispone de una tecnología que satisfaga los requisitos de dichos estándares.

.NET busca dejar de lado el modelo de software como producto y pasar a un modelo de software como servicio, es decir, hacer que el procesamiento de la información no se realice en el servidor, sino desde cualquier otro dispositivo como una PC, un teléfono o un PDA, mientras que el almacenamiento de la información y la autenticación de los usuarios sea realizada en el servidor remoto.

Características de .NET

FrameWork es el entorno de trabajo y clases que sirven de base para los lenguajes .NET, es completamente orientado a objetos y servicios Web. Es multilenguaje (soporta C#, C++, J#, Jscript, Delphi, Perl, entre varios más), siendo C# su lenguaje principal, el código escrito en un lenguaje puede heredar la implementación desde clases escritas en otro y proporciona un

entorno de ejecución virtual en el que los componentes .NET no son interpretados sino compilados de manera nativa, momento de la instalación o en tiempo de ejecución.

Con los ASP creados con .NET se realizan acciones como consultas a bases de datos y en general a todos los recursos propios del servidor. También se pueden comprar elementos ActiveX para realizar diversas tareas como el envío de correo electrónico, generar gráficas dinámicamente, etc. Lo que se puede ver como un inconveniente al no estar a la mano de cualquier programador, así como también es necesario comprar la licencia de uso por cada PC que utilizará el entorno de desarrollo.

Comprende CLR además de un conjunto de librerías de clases. El Common Language Runtime, CLR, es una infraestructura residente entre el sistema operativo y la aplicación, así se logra un nivel de encapsulación entre el desarrollador y el sistema operativo que simplifica la creación de aplicaciones permitiendo que los programadores elijan el lenguaje según sus conocimientos o características propias de la aplicación.

El entorno .NET está basado en estándares abiertos que apoyan un nivel mayor de flexibilidad. Su interfaz estándar es XML (Extensible Markup Language), aunque integra también SOAP (Simple Object Access Protocol), SCL (Soap Contract Language) y UDDI (Universal Description, Discovery and Integration).

PHP

Acrónimo de Personal Home Pages, PHP es un lenguaje de programación concebido para la creación de páginas dinámicas.

Posee independencia de plataforma por lo que las aplicaciones pueden ejecutarse en sistemas operativos tipo UNIX o Windows. Su sintaxis es muy parecida a la de lenguajes muy extendidos como C y Perl, lo que le da ventaja a la hora de la inicialización de los programadores al lenguaje. La estructura de sus programas es generalmente procedural y soporta con reservas la programación orientada a objetos en aplicaciones de grandes dimensiones. Además, ofrece un sinnúmero de funciones para que la explotación de las bases de datos sea de manera simple y práctica.

En el desarrollo de programas se integran instrucciones PHP dentro del código propio de la página Web codificado en HTML encapsuladas por medio de etiquetas, por lo que el diseñador gráfico de la Web puede trabajar de forma independiente al programador.

El servidor en que se ejecutan los Scripts de PHP codifica la página HTML y sustituye estas instrucciones por su salida con lo que al final solamente se obtiene HTML puro, lo que puede advertir una desventaja a medida que aumentan las solicitudes al servidor, pues éste realiza todo el trabajo sin delegar actividad alguna al cliente.

JSP

La tecnología orientada a la creación de páginas dinámicas desarrollada por Sun Microsystems se llama Java Server Pages o JSP.

Con los JSP se crean aplicaciones Web que se ejecutan en diferentes servidores por lo que la independencia de plataformas de software y hardware, característica más famosa de Java. Esta liberación se logra con una Máquina Virtual de Java específica para cada entorno que hace de vínculo entre el sistema operativo y el programa.

Las páginas JSP están compuestas de código HTML/XML y es el motor, basado en programas Java destinados a ejecutarse en el servidor, quien traduce el código entendible para la plataforma que se necesite. Tienen la misma función de los ASPs con la ventaja de que el programador no se debe preocupar por apuntadores, pues Java maneja la memoria de la PC. Otra característica es que es ligero pues solamente se compilan las clases que el programa necesita para que pueda correr. Cuenta con mecanismos de búsqueda y carga para cuando se modifican programas.

Al ser un lenguaje de programación bajo la filosofía Open Source, existe una gran comunidad dedicada al desarrollo de aplicaciones que al mismo tiempo se distribuyen gratuitamente como el compilador mismo.

Después de presentar las características básicas de nuestras opciones, se enumeran algunas comparaciones que ayudan a la elección del lenguaje de programación.

Java es un lenguaje orientado a objetos, ASP y CGI soportan objetos y programación estructurada y PHP es un lenguaje primordialmente estructurado con capacidad de manejar objetos.

Todas las tecnologías permiten conexión a bases de datos en mayor o menor medida, siendo el más restrictivo ASP y el menos PHP.

PHP y CGI son concebidos inicialmente para sistemas operativos tipo UNIX, ASP es orientado a sistemas Windows, especialmente NT, y JSP nace con independencia de plataforma.

Las tecnologías con filosofía open source PHP, CGI y JSP tienen una gran comunidad que las respalda y que desarrollan sus aplicaciones. ASP es específica de Microsoft el que desarrolla sus procesos internamente y vende sus APIS.

Con base las características de los LL PP anteriores, se puede observar la conveniencia de utilizar Java y sus JSPs, pues es un lenguaje robusto, de distribución libre y con muchas fuentes de soporte. Sin embargo, las aplicaciones que se han desarrollado anteriormente en el DIES han sido creadas bajo PHP por lo que se decidió utilizar este lenguaje que ofrece, al igual que Java, independencia de plataforma y gran soporte para los sistemas manejadores de bases de datos, sin olvidar que es el DIES quien se hará cargo del mantenimiento del sistema y se deben cumplir sus estándares.

2.4 Bases de Datos

Las bases de datos en los sistemas automatizados constituyen el núcleo de los mismos, ahí reside lo más valioso de todo el sistema, la información; ésta debe encontrarse organizada y es manipulada por un conjunto de programas que darán acceso en tiempo real a usuarios concurrentes con diferentes necesidades de información que solicitarán en tiempo real. La complejidad y la robustez de un sistema dependen en gran medida de cómo se diseñen la o las bases de datos que ocupará, ya que la interacción con el resto del programa informático se dará de una manera específica obedeciendo a la forma en que éstas hayan sido planteadas, además, es de ahí que se obtendrán cualidades elementales como son la Integridad de los Datos, la Seguridad, el Tiempo de Respuesta y la Concurrencia. Para un mejor control de todos los elementos y circunstancias que se deben tomar en cuenta en el Diseño de las Bases de Datos este se divide en diseño conceptual, diseño lógico y diseño físico.

Diseño Conceptual

Aquí se debe tener una visión de la Base de Datos desde el mundo real sin importar que Manejador de Base de Datos se ocupará o como será su diseño físico. El objetivo es que por medio de las especificaciones de requisitos del usuario se tenga una descripción del contenido de la información de la Base de Datos es decir, se debe descubrir el significado de los datos que se manejarán encontrando con esto las entidades, atributos y relaciones. Comprendiendo con ésto la perspectiva que cada usuario tiene de los datos, la naturaleza de los datos y el uso de los datos a través de las áreas de aplicación. El esquema conceptual resultante de este análisis puede ser utilizado para que el diseñador de la Base de Datos transmita a su cliente lo que ha entendido sobre la información que maneja.

El Diseño Lógico

Esta fase consiste en el análisis de los requerimientos del sistema para construir un esquema de la información que se utilizará basándose en un modelo de base de datos específico

como puede ser el modelo jerárquico, el modelo de red, el modelo orientado a objetos o el modelo relacional, los cuales se describen a continuación.

Modelo Jerárquico

La característica de este modelo es que los datos se organizan en forma de arborescencia de manera jerárquica, en donde el árbol formado tiene un nodo llamado raíz el cual no tiene nodos padres, un nodo padre de información puede tener varios hijos llamados hojas, los nodos hojas no pueden tener hijos y tampoco se permite que un nodo hoja tenga varios padres. El uso desmedido de apuntadores y la no independencia de los datos conllevan a la principal limitación de este modelo que es la incapacidad de limitar la redundancia de datos.

Modelo de Red

El modelo de red está formado por colecciones de registros, relacionados mediante apuntadores o ligas arbitrarias. Es muy similar al modelo jerárquico pero la modificación fundamental en éste es el concepto de nodo, pues aquí se permite que un mismo nodo tenga varios padres haciendo con esto un mayor número de arborescencias, trayendo consigo un mayor número de apuntadores reduciendo así la redundancia pero aún sin independencia de datos. Por lo tanto es difícil de administrar una base de datos justificada en este modelo.

Modelo Orientado a Objetos

Aquí las unidades básicas son los objetos que son entidades que combinan su estado (datos), comportamiento (procedimientos, métodos), e identidad (lo que los diferencia de los demás) para realizar tareas en colaboración con otros objetos.

Una Base de Datos orientada a objetos incorpora conceptos importantes del paradigma de objetos como son:

Herencia: Dentro de una jerarquía de clases se puede heredar comportamiento.

Polimorfismo: Distintos tipos de objetos pueden utilizar una operación.

Encapsulación: Esta permite gestionar los accesos incorrectos de otros objetos.

Una de las ventajas al definir una base de datos que utilice este modelo es que se pueden especificar operaciones sobre los datos.

Modelo Relacional

Este modelo considera a la base de datos como una colección de relaciones, una relación representa una tabla, una fila representa a una colección de valores que describen una entidad del mundo real, una fila se denomina tupla o registro y cada columna un campo.

Existe una correspondencia entre los términos utilizados al hablar de ficheros, tablas y relaciones:

FICHERO	TABLA	RELACIÓN
Registro	Fila	Tupla
Campo	Columna	Atributo
No. de registros	No. de Filas	Cardinalidad
No. de Campos	No. de Columnas	Grado

Tabla 2.3 Modelo Relacional

Es decir que según la tabla anterior un Registro de un Fichero en terminología relacional se representa por una Fila de la Tabla y se conoce por Tupla. Un Campo es una Columna y se conoce como Atributo.

La Cardinalidad de una Relación será el número de tuplas que la componen y el Grado será el número de atributos o columnas de la misma.

Las Relaciones son tablas con las siguientes propiedades:

No hay tuplas iguales

El orden de las tuplas no es significativo

El orden de los atributos no es significativo

En una tupla determinada cada atributo puede tomar sólo un valor (no se admiten grupos repetitivos)

Una relación tiene diferentes atributos dentro de los cuales se distinguen los siguientes:

LLAVE CANDIDATA: Es el atributo o conjunto de atributos que identifican unívoca y mínimamente cada tupla. Unívocamente ya que no puede haber dos tuplas con la misma clave y mínimamente porque no se puede pretender de ninguno de los atributos que forman la llave sin que ésta deje de ser un identificador único.

LLAVE PRIMARIA: Es aquella columna (pueden ser dos o más columnas) que identifica únicamente a una tupla. La llave primaria es un identificador que es único para cada fila. Se utiliza también como componente principal de los índices de una tabla.

LLAVE FORÁNEA: Cuando se tienen dos tablas o más, una llave ajena es aquella columna que hace referencia a una llave primaria de otra tabla.

LLAVE ALTERNATIVA: Son aquellas candidatas que no han sido elegidas.

LLAVE SIMPLE: Es una llave con atributo solamente.

LLAVE COMPUESTA: Está compuesta por más de un atributo.

El Dr. Edgar Frank Codd de los laboratorios IBM definió las bases del modelo relacional en 1970, como una serie de reglas que tiene como objetivo lograr la independencia de la representación lógica de los datos de su almacenamiento físico.

Los aspectos a que se refiere esta independencia Física-Lógica son los siguientes:

Independencia de la ordenación: Al tener acceso a los datos no se depende de cómo hayan sido ordenados físicamente.

Independencia de la indexación: Separando los índices de los datos, el sistema puede manejar la creación y mantenimiento de estos.

Independencia de los caminos de acceso: Se tiene formas de acceso más flexibles pues la navegación a través de los datos no tiene que estar previamente establecida.

Las reglas establecidas por Edgar Codd pretenden lograr la independencia física-lógica, la eliminación de redundancia, mayor flexibilidad, uniformidad, sencillez y un sólido fundamento teórico.

Con el modelo relacional a través del diseño conceptual se busca encontrar las relaciones entre los datos con los que se estará trabajando, este diseño debe pasar por un filtro que depura la integridad y consistencia de los datos llamado normalización.

Normalización

Consiste en examinar los datos que se encuentran en una tabla hasta remplazarlos por varias tablas que resulten ser simples y predecibles haciéndolas fáciles de manejar.

Los beneficios que dan razón a la realización de este proceso son los siguientes:

- Estructurar los datos de tal manera que se pueda establecer fácilmente la relación entre ellos

- Facilitar la recuperación de los datos para satisfacer las necesidades de información

- Facilitar el mantenimiento de los datos (altas, bajas, cambios)

- Reducir la posibilidad de reestructurar la base de datos ante nuevas necesidades de almacenamiento de información

- Reducir la posibilidad de redundancia e inconsistencia de datos

Formas Normales

Son técnicas para cubrir las necesidades de la mayoría de las bases de datos previniendo las anomalías en las tablas.

PRIMERA FORMA NORMAL (1FN)

Una relación R se encuentra en 1FN si y sólo si por cada renglón columna contiene valores atómicos o sea que cada uno de los atributos sólo puede incluir un dato, aunque sea compuesto, pero no se pueden incluir una lista de datos. En caso de existir grupos

repetitivos, se pone cada uno de ellos en una tabla aparte, esa nueva tabla hereda la clave de la relación que tenía y se crea una llave foránea para la nueva tabla.

Cada columna debe tener un nombre único aunque no sea importante el orden de las columnas.

Todos los ingresos en cualquier columna deben ser del mismo tipo.

Dos filas o renglones en una misma tabla no deben ser idénticas aunque el orden de las filas no es importante.

SEGUNDA FORMA NORMAL (2FN)

Una relación R esta en 2FN si y sólo si está en 1FN y los atributos no primos dependen funcionalmente de la llave primaria.

Una relación se encuentra en 2FN, cuando cumple con las reglas de la 1FN y todos sus atributos que son llaves dependen por completo de la clave. De acuerdo con esta definición cada tabla que tiene un atributo único como clave, esta en segunda forma normal.

TERCERA FORMA NORMAL (3FN)

Para definir la 3F N se requiere definir dependencia transitiva: En una afinidad (tabla bidimensional) que tiene por lo menos 3 atributos (A,B,C) en donde A determina a B, B determina a C pero no determina a A. Una relación R esta en 3FN si y sólo si esta en 2FN y todos sus atributos no primos dependen no transitivamente de la llave primaria. Consiste en eliminar la dependencia transitiva que queda en una segunda forma normal, en pocas palabras una relación esta en 3FN si esta en 2FN y no existen dependencias transitivas entre los atributos, se hace referencia a dependencias transitivas cuando existe más de una forma de llegar a referencias a un atributo de una relación.

Diseño Físico

El objetivo de esta etapa es producir una descripción de la implementación de la base de datos en memoria secundaria. Esta descripción incluye las estructuras de almacenamiento y los métodos de acceso que se utilizarán para conseguir un acceso eficiente a los datos. El diseño físico se divide de cuatro fases, cada una de ellas compuesta por una serie de pasos:

Traducir el esquema lógico global para el SMBD específico.

Diseñar las relaciones base para el SMBD específico.

Diseñar las reglas de negocio para el SMBD específico.

Diseñar la representación física.

Analizar las transacciones.

Escoger las organizaciones de ficheros.

Escoger los índices secundarios.

Considerar la introducción de redundancias controladas.

Estimar la necesidad de espacio en disco.

Diseñar los mecanismos de seguridad.

Diseñar las vistas de los usuarios.

Diseñar las reglas de acceso.

Monitorizar y afinar el sistema.

Sistema Manejador de Bases de Datos (SMBD)

Un SMBD consiste en una Base de Datos (BD) y un conjunto de programas (aplicaciones) para tener acceso y manipulación de los datos. El objetivo de un SMBD es crear el ambiente en donde sea posible almacenar y recuperar información en forma eficiente y conveniente.

Las funciones del SMBD son:

Interacción con el manejador de archivos (Sistema Operativo)

Implantación de la seguridad

Puesta en práctica de la integridad

Respaldo y recuperación

Control de la concurrencia (verificar los privilegios de los usuarios)

Lenguajes del SMDB

En la estructura básica de un Sistema Manejador de Base de Datos se enuncian los lenguajes que permiten trabajar sobre la base de datos. Estos lenguajes estándar son:

DDL (Data Definition language): Lenguaje de Definición de Datos. Por medio de éste el SMDB identifica las descripciones de los elementos de los esquemas y almacena la descripción del esquema en el catálogo del SMDB y el gestor especifica el esquema conceptual e interno (Base de datos Almacenada).

SDL (Store Definition language): Lenguaje de definición de almacenamiento. Es utilizado por el SMDB para especificar el esquema interno que corresponde a la base de datos almacenada.

VDL (View Definition language): Lenguaje de definición de vistas. Es utilizado por el SMDB para especificar las vistas del usuario y sus correspondencias con el esquema conceptual. En las bases de datos relacionales, el SQL, representa una combinación de los anteriores.

DML (Data Manipulation language): Lenguaje de manipulación de datos. Permite la manipulación de las operaciones de inserción, eliminación y modificación.

Tipos de DML:

De alto nivel o no por procedimientos: SQL.

De bajo nivel o por procedimientos.

Algunos sistemas gestores de bases de datos son:

MS SQL Server 2005

SQL Server 2005 permite mayor escalabilidad, disponibilidad y seguridad con optimización de los tiempos en implantación y gestión de las Bases de Datos (BBDD) que sus versiones anteriores.

Entre sus características principales cuenta con un entorno de desarrollo flexible, además tiene un CLR (Common Language Runtime) que permite a los desarrolladores elegir el lenguaje de sus aplicaciones para la base de datos con tecnologías .NET, MS Visual Basic y Visual C#, TrasactSQL entre otros, así como les permite utilizar sus propios tipos y funciones. En SQL Server 2005, se pueden crear servicios Web en la capa de Base de Datos convirtiendo a SQL Server en un servidor HTTP.

Tiene la capacidad de poder ejecutarse con 32 o 64 bits (según la versión) para un óptimo rendimiento en bases de datos que exigen 2 Gb en memoria RAM, también puede integrar y establecer conexión con aplicaciones y bases de datos descentralizadas mediante servicios estandarizados de Web, plataforma .NET y SQL Server.

En materia de disponibilidad cuenta con un sistema Snapshots que toma imágenes exactas de la base de datos para resolver problemas en caso de que falle el sistema, posee operaciones de replica de BBDD (mirroring) tiene la capacidad de modificar la estructura de la base de datos mientras otros usuarios acceden a la información además cuenta con niveles de seguridad predeterminados para una rápida administración con la clara opción de ajustar los perfiles según las necesidades del sistema.

Cuenta con una herramienta de administración desarrollada en Visual Basic que integra herramientas automatizadas de operaciones recurrentes en administración de base de datos para ahorrar tiempo de gestión y mantenimiento, capaz de cambiar atributos con un clic.

Una de las grandes ventajas que tiene este SMBD es la estrecha vinculación con otros productos de Microsoft como Visual Studio y Office, ya que esto ofrece flexibilidad de desarrollo con herramientas de esta familia.

Las versiones disponibles para SQL Server 2005 son: Express, Workgroup, Standard y Enterprise.

Para el análisis de este administrador de DDBB se seleccionaron dos versiones por tener entre sus características algunas de las esenciales para el proyecto. Las versiones restantes se descartaron además, por las características propias del ámbito de aplicación de las versiones.

El resumen de las características particulares de las versiones Standard y Enterprise se muestran en el siguiente cuadro.

Característica	Versión	
	Standard	Enterprise
Perfil de la versión	Administración completa de datos y plataforma de análisis para medianas y grandes empresas	Niveles de soporte para administración empresarial, y grandes sitios WEB con transacciones on-line y sistemas de warehousing.
Número de CPU s	1-4	Ilimitado
Memoria RAM	Ilimitada	Ilimitada
Características generales	Replicación de BD en espejo Servidor OLAP Servidor de Reportes Análisis con Minería de Datos Replicación completa incluyendo Servidores Standby ETL Básico Vistas dinámicas de administración y reportes Integración con Ms Baseline Security Analyzar	Replicación de BD en espejo avanzada Opresiones completas tanto en línea como paralelamente Snapshot de la BD Reportes avanzados: personalizados, a gran escala y ad hoc ETL avanzado con ruteo de datos
Precio aproximado	\$6 000 por procesador \$2 799 (servidor + 10 usuarios)	\$25 000 por procesador \$13 500 (servidor + 25 usuarios)

Tabla 2.4 Versiones SQL

Oracle 9i Database Standard ONE Edition

Oracle9i en su Standard ONE Edition permite una fácil instalación y administración de la BD en equipos de hardware cuya máxima capacidad sea de un procesador solamente.

La primer característica que tiene Oracle es que *corre* sobre las principales plataformas o sistemas operativos como Windows, Linux y todas las versiones de Unix ofrecidas por diversas empresas como IBM, Sun, Digital, HP, Sequent, etc.

Para que su administración sea sencilla, Oracle 9i entrega con su motor de BBDD, el Oracle Enterprise Manager (herramienta gráfica y centralizada para realizar toda la administración de todos los componentes Oracle) que permitirá realizar tareas de gestión desde la más sencilla hasta la más compleja por comandos dados desde el mouse, reduciendo así costos de soporte, administración y mantenimiento de los sistemas. Además tiene el Enterprise User Security que facilita la administración de usuarios, roles y privilegios.

Una ventaja importante que tiene este SDBD es su escalabilidad ilimitada sin importar su edición, es decir, se puede adquirir primero una versión de Oracle con características básicas y posteriormente, dadas las necesidades de la base de datos, puedan crecer los sistemas junto con el manejador, lo que repercute favorablemente en su mantenimiento, crecimiento y actualización.

En cuanto a seguridad es provisto de una característica llamada Readable Standby Database que permite tener copias de la base de datos productiva en lugares dispersos a la ubicación principal del sistema, protegiendo así los datos ante fallas, además de que estas copias pueden funcionar en modo de sólo lectura. Oracle 9i ofrece capacidades mediante las cuales protege los datos del sistema contra usuarios no autorizados. Características como la encriptación a 128 bits de datos permiten esta protección de los datos de una forma muy segura y confiable.

En materia de disponibilidad en Oracle 9i la lectura de datos no bloquea a la escritura de datos y viceversa, esto es posible sin comprometer la integridad de los datos porque Oracle recreará una imagen para que el usuario lector pueda ver la información que esta en la BD antes que alguien altere el dato. Asimismo, incluye La herramienta Flashback Query & Log

Miner que incluye capacidades para que el usuario final pueda auto servirse para la corrección de errores.

Oracle le permite al administrador controlar en tiempo real el comportamiento de la BD. De igual forma la BD de Oracle 9i, permite modificaciones dinámicas manteniendo así la continuidad de servicio hasta de 24x7. La memoria que se asigna a las instancias de BD puede ser aumentada dinámicamente si así fuera requerido con el propósito de aumentar su rendimiento en caso de ser necesario por sobrecargas de trabajo u hora pico.

En resumen, entre las ventajas que Oracle 9i puede ofrecer están:

Un entorno de protección de datos seguros, sin pérdida de datos

Proveerá una alta tasa de disponibilidad de los datos

Oracle 9i es multiplataforma

Proveer una reparación rápida y precisa de las bases de datos dañadas

Reducir los costos de mantenimiento de la BD través de características como auto-afinación y auto-administración

Oracle actualmente cuenta con 17 evaluaciones/certificados de seguridad

Adaptive Server Enterprise 15, Sybase

Adaptive Server Enterprise (ASE) 12.5.3 reduce el costo de migración entre plataformas y mejora el rendimiento para muchas aplicaciones. Adicionalmente, ASE 12.5.3 hace más fácil el manejo de documentos XML y el uso de encriptación avanzada para comunicaciones de red, es un SMBD de alto desempeño con características de misión-crítica, de alta demanda y grandes volúmenes de transacciones mientras esta disponible a bajo costo.

Entre las características que ofrece ASE 15 se incluyen una encriptación on-disk y un soporte mejorado para manejo de datos no estructurados. Incluye una nueva tecnología de procesamiento de consultas que ha demostrado una significativa mejoría en el desempeño del SMBD y reducción en el consumo de recursos de hardware además, proporciona mejor desempeño con el uso de particiones inteligentes que apuntan sólo a la información requerida. ASE 15 tiene un sistema fácil de operar de privacidad basado en encriptación que no requiere modificaciones en aplicaciones cuando se necesita escalabilidad. El sistema se asegura que los datos estén protegidos mientras se transfieren cuando se está accediendo a

ellos o cuando se están almacenando. El cifrado de ASE 15 proporciona el nivel de seguridad que se necesita para preservar el aislamiento de los datos.

Las nuevas capacidades hacen eficiente la disponibilidad de sistema: permite tener particiones de datos en dispositivos físicos independientes de modo que si un dispositivo falla, las particiones restantes todavía estén disponibles, por lo que no se necesitan de arreglos RAID para resguardar la DB.

Las nuevas características optimizan las ventajas operacionales que ofrecen las Tecnologías de Información, para mejorar su funcionamiento, reducir complejidad y riesgos, así como reducir costos. Entre otras nuevas características en esta versión se incluyen:

Columnas automatizadas para un uso más fácil de aplicaciones en el diseño

Funciones de indexado para incrementar el desempeño de las consultas

Un amplio soporte del servidor (Very Large Server Support, VLSS) para manejar grandes sets de datos

Estadísticas auto-actualizables que reducen la carga de administración

Vista Enterprise hace más fácil administrar varios servidores desde la misma estación de trabajo

Ayuda tipo Wizard, ideal para administradores principiantes

Sistema avanzado de medición que proporciona una sincronización más fácil, mejor visibilidad y supervisión del sistema y una vista de planificación que hace mas fácil para los DBAs y desarrolladores diagnosticar y sincronizar consultas programadas

La versión 12.5.3a es una versión completa de ASE que soporta actualizaciones a partir de versiones 11.9.x, 12.0.x,12.5,12.5.1, 12.5.2 y 12.5.3.

SQL Anywhere, Sybase

SQL Anywhere es un paquete que proporciona administración y sincronización de los datos para permitir un rápido desarrollo y un despliegue de aplicaciones dirigidas a ambientes móviles, embebidos y para las pequeñas y medianas empresas. SQL Anywhere aumenta el

alcance de los datos, la información y las aplicaciones en dondequiera que las transacciones ocurran.

Es ideal para la expansión ilimitada de grupos de trabajo y usuarios móviles. En un simple paquete integrado, se puede disponer de toda la tecnología para crear soluciones potentes para distribuir información hacia puntos distantes desde la empresa. Para computación móvil, SQL Anywhere Studio permite a los usuarios estar ocasionalmente conectados, así la información corporativa no solo está siempre disponible, sino también actualizada. Permite implementar la automatización de fuerzas de ventas, help desks y call centers, y sistemas con servicios de autoconsulta de clientes brindando la potencia de las aplicaciones empresariales a sitios remotos. Para soluciones embebidas, SQL Anywhere Studio ofrece el bajo sobre costo y rica funcionalidad de aplicaciones.

Entre sus principales características destacan:

Fácil uso y administración, ya que se auto-administra y se auto-sincroniza requiriendo una mínima implicación del administrador de la base

Nivel empresarial de desempeño y escalabilidad; soporta sistemas de multiprocesadores simétricos y tiene un nuevo algoritmo de procesamiento que acelera el performance de consultas de gran tamaño, más complejas y con un número mayor de usuarios concurrentes

Verdadera seguridad de extremo a extremo; con una fuerte opción de encriptación de 128 bits dando como resultado real seguridad de extremo a extremo que resuelve las necesidades de aislamiento-consistente para clientes en industrias tales como los servicios financieros.

Soporte multiplataforma; soporta un amplio rango de sistemas operativos y las plataformas más populares de handhelds lo que la provee de mayor flexibilidad en despliegue de aplicaciones.

Soporte dedicado para servicios WEB; simplifica el desarrollo de las aplicaciones de las DDBB conectadas a servidores WEB, la integración con sistemas existentes y el despliegue de las aplicaciones a través de diversos ambientes de TIs.

Optimizado para dispositivos handheld y aplicaciones inteligentes; proporciona una solución ideal para los ambientes de memoria-obligada tales como dispositivos handheld.

MySQL

Creado por la empresa MySQL AB con licencia de software libre, MySQL es un SDBD relacional, su diseño le permite soportar de forma eficiente una gran carga. Aunque MySQL tiene licencia GPL, su empresa creadora vende también una versión comercial de MySQL siendo las mismas pero la segunda cuenta con soporte técnico y la posibilidad de integrar el SDBD dentro de un software propietario.

Este manejador de bases quizá sea el más usado en el mundo del software libre por su gran rapidez y facilidad de uso. Esta gran aceptación se debe a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Ya que en las aplicaciones web existe una baja concurrencia en la modificación de datos y gran afluencia en la lectura de los mismos, este SDBD es excelente para páginas dinámicas ligadas a bases de datos.

Se puede ejecutar bajo múltiples sistemas operativos como OS/2, SunOS, Solaris, Windows, Linux, entre varios más. Además existe un gran número de APIs que permiten acceder a las bases de datos de MySQL para aplicaciones escritas en varios lenguajes de programación como C, C++, Pascal, Delphi, PHP, Perl, etc. También existe una interfaz ODBC que permite a cualquier lenguaje que lo soporte comunicarse con este tipo de bases de datos.

Una de las particularidades de este manejador es que tiene múltiples motores de almacenamiento permitiendo al programador elegir la mas conveniente para cada tabla de la base.

La última versión disponible es la 5.0.22 y entre sus características podemos encontrar: soporte multiplataforma, procedimientos almacenados, Soporta gran cantidad de tipos de datos para las columnas, gran portabilidad entre sistemas, soporta hasta 32 índices por tabla, gestión de usuarios y contraseñas, manteniendo un buen nivel de seguridad en los datos.

PostgreSQL 7.1.x

PostgreSQL puede ser integrada al ambiente Windows permitiendo de esta manera a los desarrolladores, generar nuevas aplicaciones o mantener las ya existentes. Permite desarrollar o migrar aplicaciones desde Access, Visual Basic, Foxpro, Visual Foxpro, C/C++ Visual C/C++, Delphi, etc., para que se utilice PostgreSQL como servidor de BD.

Es un RSMDB considerado como el sistema gestor de BD de código abierto más avanzado del mundo por programadores especializados. Posee características que se pueden observar en productos comerciales de alta demanda, tales como DB2 y Oracle. Es confiable, rápido, escalable, de libre distribución, sin límite de usuarios, se puede montar en varias plataformas, entre ellas Windows y Linux. Con este sistema se pueden lograr bases de tamaño ilimitado, sólo restringido por la capacidad de almacenamiento del hardware, máximo tamaño de una tabla hasta 64 Tb y máximo tamaño de un campo de 1Gb, su máxima cantidad de registros es ilimitado y en columnas por tabla de hasta 1600.

Características principales:

SMBD Objeto-Relacional: aproxima los datos a un modelo objeto-relacional y es capaz de manejar complejas rutinas y reglas. Ejemplos de su funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transactions, optimización de consultas, herencia y arreglos.

Altamente Extensible: soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario e integridad referencial que es utilizada para garantizar la validez de los datos de la DB.

API Flexible: La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para este RSMDB. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike

Lenguajes Procedurales: soporta lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL que es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

MVCC: o Control de Concurrencia Multi-Versión (Multi-Version Concurrency Control), es la tecnología que usa para evitar bloqueos innecesarios. En otros SDBD como MySQL o Access, la lectura tiene que esperar para acceder a información de la BD, provocando un tiempo de espera, inducido por usuarios que están escribiendo en la BD, es decir, el lector está bloqueado por los escritores que están actualizando registros. Mediante MVCC se mantiene una ruta de todas las transacciones realizadas por los usuarios de la BD.

Write Ahead Logging (WAL): incrementa la dependencia de la base al registro de cambios antes de que estos sean escritos en la base de datos. Esto garantiza que en caso de que la DB el funcionamiento de la base se interrumpa, exista un registro de las transacciones a partir del cual se puede restaurar. Una vez que el sistema ha quedado restaurado, un usuario puede continuar trabajando desde el punto en que lo dejó cuando el sistema gestor o la base dejaron de funcionar.

Cuadro Comparativo

Característica / RSMDB	Oracle	Anywhere	ASE	PosgreSQL	SQL Server
Datos definidos por el usuario	*	*	*	*	*
API Flexible	?	*	*	*	
Lenguajes procedurales y nativos	*	*	*	*	*
Control de concurrencia	*	*	*	*	*
Disponibilidad de DB en caso de catástrofe	*	*	*	*	*
Soporte de 64 bytes	*		*	*	*
Encriptación de columnas	*		*		
Administración XML /XML Nativo	*	*	*	*	*
Opciones Java	*	*	*	*	
Servicios WEB (OLAP)	*	*	*	*	*
Herramientas de Administración / auto-administración	*	*	*		*
Multiplataforma	*	*	*	*	
Ayuda tipo Wizard			*		
Escalabilidad ilimitada con versiones anteriores(*) y diferentes(**)	* **	**	**	**	**
Índices inteligentes/eficientes	?	*	*	*	
Perfiles de seguridad	*	*	*		*

Tabla 2.5 Comparativo Manejadores de Bases de Datos

* ** Cuenta con la característica con sus propios métodos

? No se encontró suficiente información para determinar la inexistencia de la característica

Capítulo 3 Análisis del Diseño

3.1 Diagrama de Flujo de Datos

Los diagramas que se mostrarán en este capítulo tienen como objetivo expresar de manera visual los procesos contemplados para el funcionamiento del sistema. Con la finalidad de mostrar una representación gráfica del origen y destino de la información manejada en el sistema, se elabora el Diagrama de Flujo de Datos General (Figura 3.1) en donde se pueden observar los procesos que se llevan a cabo dentro del sistema, el flujo de información y los almacenes de datos.

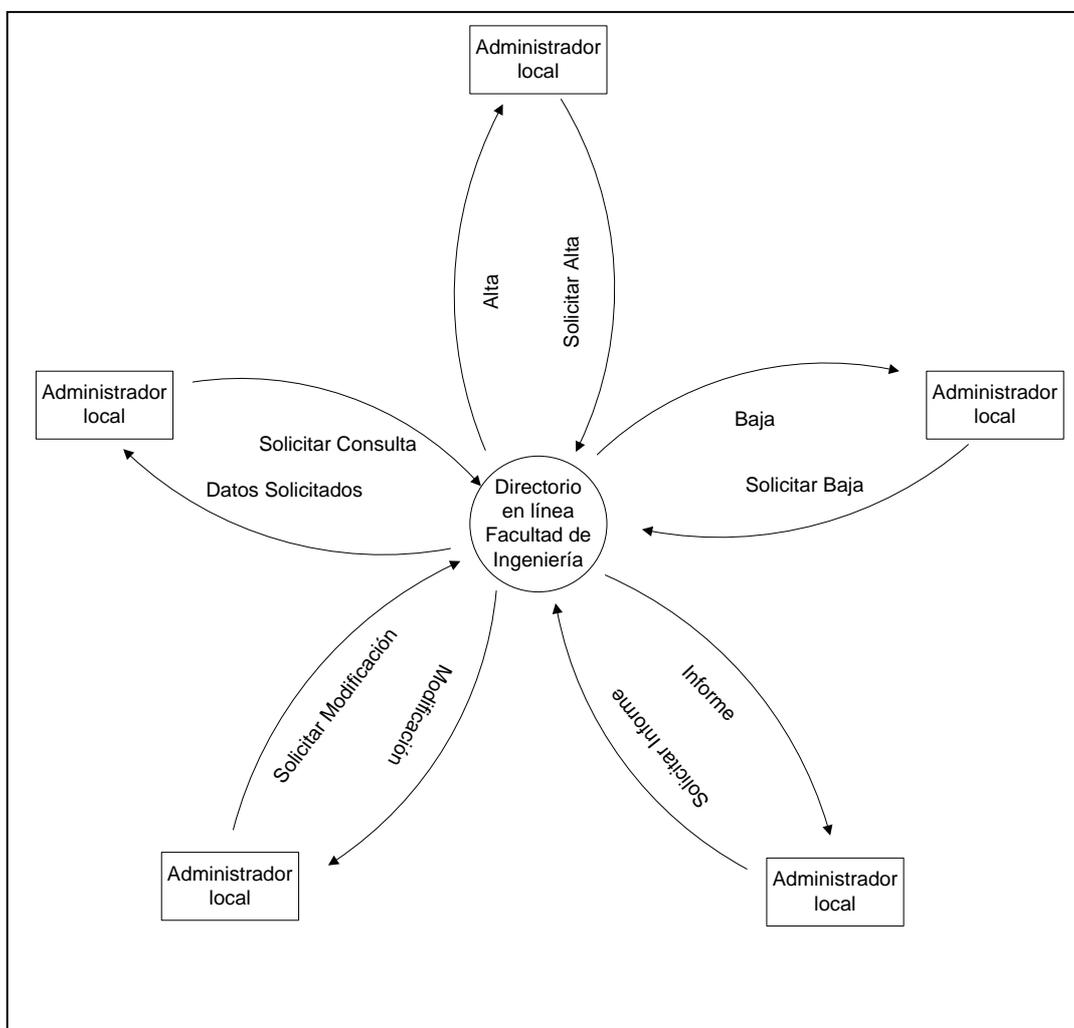


Figura 3.1 Diagrama de Contexto General

3.2 Diagramas de Flujo de Datos Detallados

Diagrama de Flujo de Datos (DFD); NIVEL 1

En la siguiente Figura 3.2 se observan los procesos y flujos de datos del Directorio en Línea de la Facultad de Ingeniería.

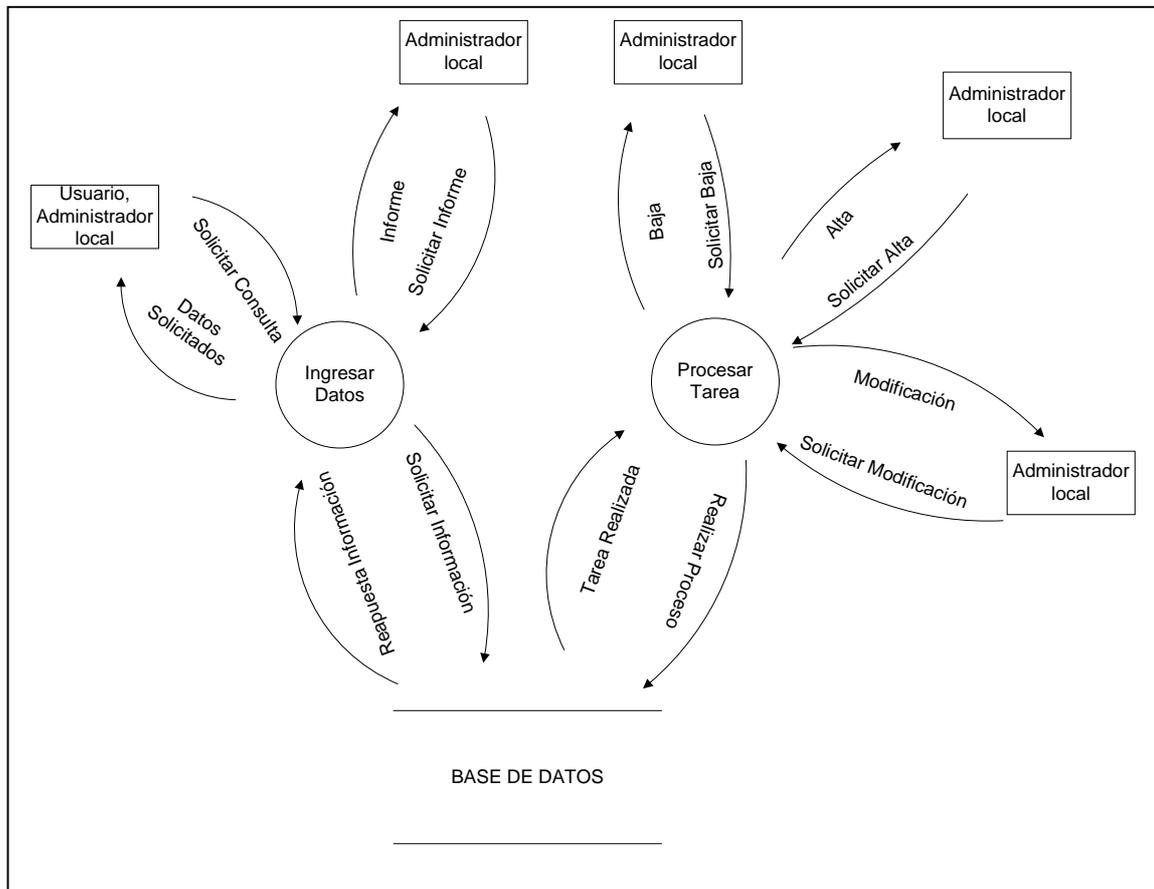


Figura 3.2 Diagrama de Flujo de Datos (Nivel 1)

Diagrama de Flujo de Datos para procesar Tarea; Nivel 2

Se muestra a continuación mediante la Figura 3.3 los procesos y los flujos de datos que se llevan a cabo al procesar tarea (ALTA).

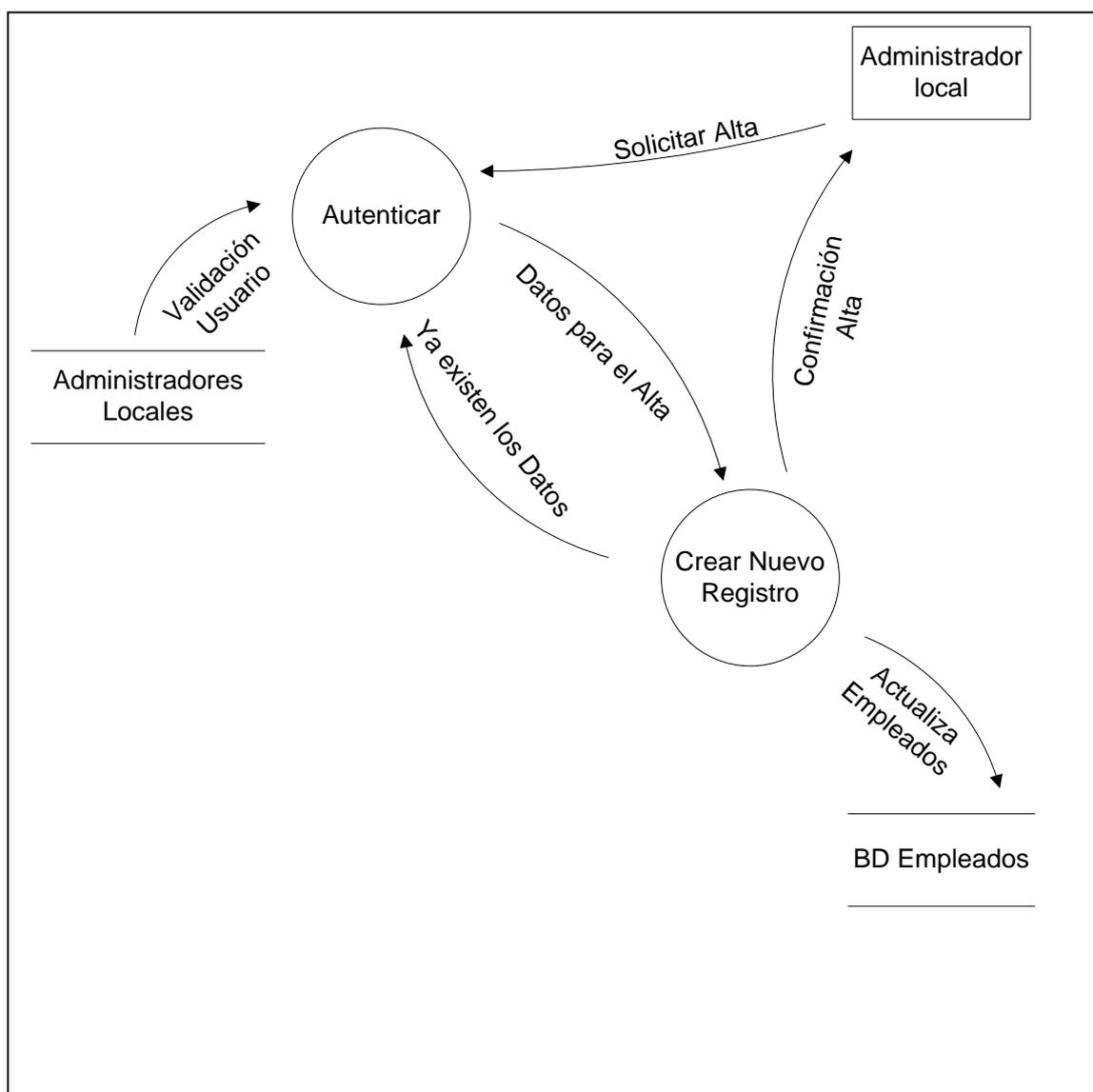


Figura 3.3 DFD para procesar un ALTA (Nivel 2)

3.3 Normalización

La normalización consiste en obtener varias tablas simples y predecibles que sean fáciles de manejar a partir de una reexaminación de los datos que se encuentran agrupados en una tabla obtenidos del diagrama Entidad-Relación Modelo Conceptual.

Utilizando la tabla que contiene la información de los Funcionarios, se ejemplificará como se llevó a cabo la normalización.

noEmpleado	CveCargoFun	Cargo	T Tel1_Email
1011	12	Director	56220101;director@ingenieria.unam.mx
1012	14	Secretario Académico	56223134;secacad@ingenieria.unam.mx
1025	18	Jefe de División	56220872;jdivision@ingenieria.unam.mx
1112	15	Secretario Técnico	56228327;sectec@ingenieria.unam.mx
1125	17	Secretario Administrativo	56221515;secadmin@ingenieria.una.mx

Tabla 3.1a Información de Funcionarios

Primera Forma Normal (1FN)

De acuerdo a la definición de esta Primera Forma, una relación está en 1FN si y sólo si cada uno de los campos contiene un solo valor para un registro determinado. Si se observa el contenido de Tel_Email en donde se tiene dos valores uno para Teléfono y otro para Delegación, y aplicamos la 1FN, se tiene la siguiente Tabla 3.1b:

noEmpleado	CveCargoFun	Cargo	Tel1	Email
1011	12	Director	56220101	director@ingenieria.unam.mx
1012	14	Secretario Académico	56223134	secacad@ingenieria.unam.mx
1025	18	Jefe de División	56220872	jdivision@ingenieria.unam.mx
1112	15	Secretario Técnico	56228327	sectec@ingenieria.unam.mx
1125	17	Secretario Administrativo	56221515	secadmin@ingenieria.una.mx

Tabla 3.1b Primera Forma Normal (1FN)

Segunda Forma Normal (2FN)

Para considerar que la tabla está en 2FN primero debe cumplir que está en 1FN y que todos los atributos no llave dependen por completo de la llave primaria. Como se muestra en la Figura 3.4.

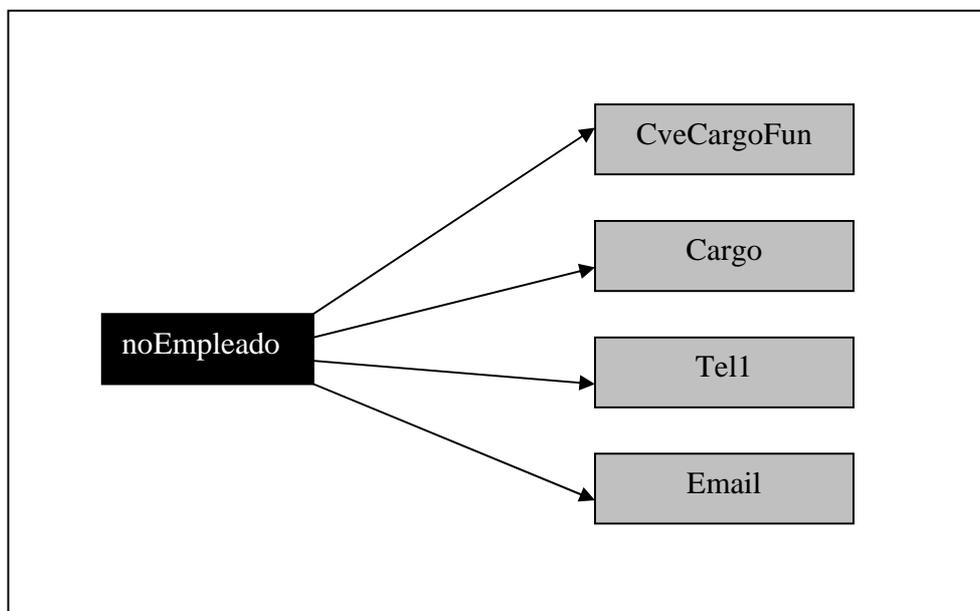


Figura 3.4 Segunda Forma Normal (2FN)

En este caso la llave primaria es noEmpleado, dado que todos los atributos que aparecen en la tabla dependen totalmente de la llave primaria, entonces esta relación se encuentra en 2FN.

Tercera Forma Normal (3FN)

Sólo cuando se haya cumplido con la Segunda Forma Normal y cuando los campos no dependan unos de otros, se considera que la relación se encuentra en Tercera Forma Normal.

Para el caso analizado, la Figura 3.5 nos muestra los siguientes elementos:

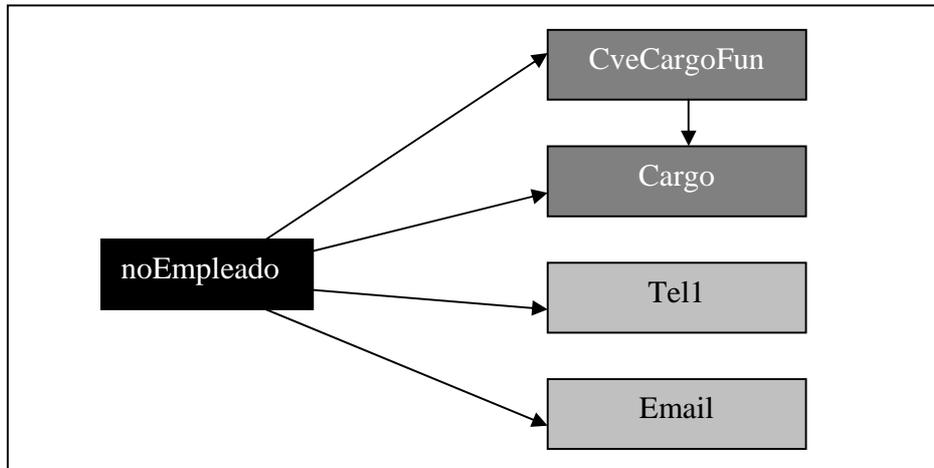


Figura 3.5 Los campos dependen unos de otros.

Como se observa en la figura anterior, Cargo depende de CveCargoFun por lo que se procede a descomponer la relación de la manera siguiente (Figura 3.6):

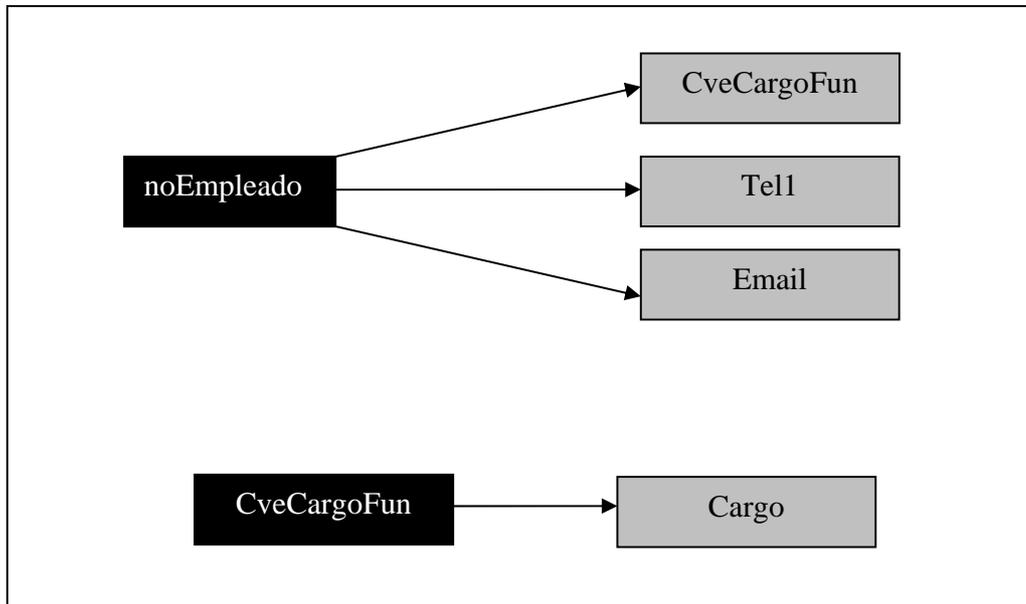


Figura 3.6 Tercera Forma Normal (3FN)

Finalmente se observa las relaciones del caso:

noEmpleado	CveCargoFun	Tel1	Email
1011	12	56220101	director@ingenieria.unam.mx
1012	14	56223134	secacad@ingenieria.unam.mx
1025	18	56220872	jdivision@ingenieria.unam.mx
1112	15	56228327	sectec@ingenieria.unam.mx
1125	17	56221515	secadmin@ingenieria.una.mx

Tabla 3.2 Datos de contacto de los Funcionarios

CveCargoFun	Cargo
12	Director
14	Secretario Académico
18	Jefe de División
15	Secretario Técnico
17	Secretario Administrativo

Tabla 3.3 Cargo de los Funcionarios

3.5 Definición de Estándares

Los estándares utilizados durante la etapa de Desarrollo del proyecto respecto a la nomenclatura de la base de datos se indican a continuación.

Nomenclatura de la Base de Datos

Tablas

El nombre de las tablas estará dado por un sustantivo o sustantivos en español o sus abreviaciones, estos sustantivos deberán estar en singular, sólo serán permitidos caracteres alfanuméricos. El total de caracteres que lo conformen no debe exceder de cien caracteres. Si el nombre constara de dos o más palabras, éstas serán de hasta cinco caracteres por palabra comenzando con mayúscula. Para las tablas tipo catálogos, su nombre se construirá con las reglas anteriormente descritas y además se antepondrá una C al nombre de la tabla que indique que se trata de un catálogo y para las tablas que refieran a las identidades del empleado, su nombre empezará con T.

Atributos

El nombre de cada atributo describirá su función y no excederá de treinta caracteres, comenzando empezarán con minúscula excepto las llaves primarias y foráneas.

Si el nombre constara de dos o más palabras, éstas serán de hasta 5 letras por palabra comenzando con mayúscula desde la segunda palabra.

El nombre de los atributos que hagan referencia a llaves foráneas en los catálogos comenzará con el sufijo Cve.

Consultas

Las palabras para consultas serán de nombres representativos y no sobrepasarán juntas 40 caracteres.

3.6 Diccionario de Datos

Un Diccionario de Datos contiene metadatos, es decir, datos acerca de los datos. Un sistema de base de datos consulta el Diccionario de Datos antes de leer o modificar los datos reales.

La gran utilidad del Diccionario de Datos es aprovechada tanto por el Diseñador de bases de datos, ya que es para él una forma de documentación donde se guardan los detalles y descripciones de todas las partes del sistema, como para el Analista porque por medio de esta información puede conocer cuántos caracteres abarca un determinado dato, que otros nombres recibe en distintas partes del sistema, o donde se utiliza.

De acuerdo con la descripción teórica y la definición de estándares para su desarrollo definidos anteriormente, a continuación se presenta el diccionario de datos del sistema que es motivo de este trabajo:

CAreaDepto

Catálogo de los diferentes departamentos que comprenden las divisiones o subdependencias.

Campo	Tipo	Nulo	Único	Comentarios
<u>CveAreaDepto</u>	varchar(4)	No	Si	Identifica los diferentes departamentos de las divisiones
abrevAreaDepto	varchar(7)	No	No	Abreviatura de los departamentos o áreas
descAreaDepto	varchar(50)	No	No	Nombre completo de las entidades

CCargoFun

Catálogo que almacena los cargos de funcionarios comprendidos en el organigrama oficial

Campo	Tipo	Nulo	Único	Comentarios
<u>CveCargoFun</u>	tinyint(3)	No	Si	Identificador del cargo de funcionario
abrevCargoFun	varchar(15)	Si	No	Abreviatura del cargo de funcionario
descCargoFun	varchar(60)	No	Si	Nombre completo del cargo de funcionario

CCargoNom

Catálogo para los cargos del personal administrativo

Campo	Tipo	Nulo	Único	Comentarios
<u>CveCargoNom</u>	smallint(3)	No	Si	Identificador del cargo administrativo
abrevCargoNom	varchar(15)	Si	No	Abreviatura del cargo administrativo
descCargoNom	varchar(60)	No	Si	Nombre completo del cargo administrativo

CCargoRep

Catálogo que contiene los cargos de representación

Campo	Tipo	Nulo	Único	Comentarios
<u>CveCargoRep</u>	tinyint(2)	No	Si	Identificador del cargo de representación
abrevCargoRep	varchar(10)	No	No	Abreviatura del cargo de representación
descCargoRep	varchar(60)	No	Si	Nombre completo del cargo de representación

CCategoria

Catálogo de categorías para los empleados con carácter académico

Campo	Tipo	Nulo	Único	Comentarios
<u>CveCategoria</u>	tinyint(3)	No	Si	Identificador de las categorías para académicos
descCategoria	varchar(60)	No	Si	Nombre completo de la categoría de académico
abrevCategoria	varchar(10)	No	No	Abreviatura de las categorías para académicos

CCuerpoRep

Catálogo de grupos colegiados a los que puede pertenecer una persona

Campo	Tipo	Nulo	Unico	Comentarios
<u>CveCuerpoRep</u>	smallint(2)	No	Si	Identificador de los grupos o cuerpos colegiados de los representantes
abrevCuerpoRep	varchar(6)	Sí	no	Abreviatura de los cuerpos colegiados
descCuerpoRep	varchar(25)	Sí	no	Nombre completo de los cuerpos colegiados

CDependencia

Catálogo que contiene las dependencias de las que se conforma la Facultad de Ingeniería

Campo	Tipo	Nulo	Único	Comentarios
<u>CveDependencia</u>	varchar(1)	No	Si	Identificador de la dependencia
abrevDep	varchar(5)	No	Si	Abreviatura del nombre de la dependencia
descDep	varchar(60)	No	Si	Nombre completo de la dependencia

CGrado

Catálogo de grados académicos

Campo	Tipo	Nulo	Único	Comentarios
<u>CveGrado</u>	tinyint(2)	No	Si	Identificador del grado académico
abrevGrado	varchar(10)	No	Si	Abreviatura del grado académico
descGrado	varchar(30)	No	Si	Nombre completo del grado académico

CSubDep

Catálogo que contiene las divisiones académicas y secretarías que comprenden a la Facultad de Ingeniería

Campo	Tipo	Nulo	Único	Comentarios
<u>CveSubDep</u>	varchar(3)	No	Si	Identificador de la División o Secretaría
abrevSubDep	varchar(10)	No	No	Abreviatura del nombre de las diferentes Divisiones o Secretarías
descSubDep	varchar(50)	No	No	Nombre de la División o Secretaría

TAcademico

Tabla que almacena los datos de contacto de una persona como Académico

Campo	Tipo	Nulo	Único	Comentarios
<u>CveAcademico</u>	int(6)	No	Si	Identificador único del registro por académico
<i>noEmpleado</i>	int(5)	No	No	Número que asocia el empleado con su carácter de académico
<i>CveDependencia</i>	varchar(1)	No	No	Atributo que refiere a la entidad donde el empleado es académico
<i>CveSubDep</i>	varchar(3)	No	No	Identificador de la División o Secretaría a la que esta adscrito el académico
<i>CveAreaDepto</i>	varchar(4)	No	No	Área o departamento a la que pertenece el académico
<i>CveCategoria</i>	tinyint(3)	No	No	Categoría con la que está contratado el académico
materia	varchar(250)	Si	No	Materia en la que tiene el nombramiento de la categoría
antiguedad	date	Si	No	Antigüedad del nombramiento como académico
<i>CveCat2</i>	tinyint(3)	No	No	Segunda categoría que puede tener el académico
materia2	varchar(250)	Si	No	Materia en la que tiene el segundo nombramiento
antig2	date	Si	No	Antigüedad del segundo nombramiento académico
<i>CveCat3</i>	tinyint(3)	No	No	Tercera categoría puede tener el académico
materia3	varchar(250)	Si	No	Materia en la que tiene el tercer nombramiento
antig3	date	Si	No	Antigüedad del tercer nombramiento académico
<i>CveCat4</i>	tinyint(3)	No	No	Cuarta categoría que puede tener el académico
materia4	varchar(250)	Si	No	Materia en la que tiene tercer nombramiento
antig4	date	Si	No	Antigüedad del segundo nombramiento académico
Horarios	varchar(250)	Si	No	Horario en el que se encuentra en su cubículo o la hora de clases y el salón
fechaRegistro	date	No	0000-00-00	Fecha de actualización

TEmpleado

Tabla que almacena los datos de contacto personales de un empleado

Campo	Tipo	Nulo	Único	Comentarios
<u>noEmpleado</u>	int(5)	No	Si	Identificador único para un trabajador en la dependencia
<i>CveGrado</i>	tinyint(2)	No	No	Máximo grado de escolaridad que tiene la persona
nombre	varchar(30)	No	No	Nombres de la persona
ap1	varchar(40)	No	No	Primer apellido de la persona
ap2	varchar(40)	Si	No	Segundo apellido de la persona
<i>CveSubDep</i>	varchar(20)	No	No	Divisiones a las que pertenece el trabajador
fechaRegistro	date	No	No	Fecha de última actualización

TAdministrativo

Tabla que almacena los datos de contacto de unan persona como Administrativo

Campo	Tipo	Nulo	Único	Comentarios
<u>CveAdministrativo</u>	int(6)	No	Si	Identificador único del registro por administrativo
<i>NoEmpleado</i>	int(5)	No	Si	Número que asocia el empleado con su carácter de administrativo
<i>CveDependencia</i>	varchar(1)	No	No	Clave de la dependencia donde el empleado tiene un cargo como administrativo
<i>CveSubDep</i>	varchar(3)	No	No	Clave de la División o Secretaría en la que se desempeña el administrativo
<i>CveAreaDepto</i>	varchar(4)	Si	No	Área o departamento a la que pertenece el administrativo
<i>CveCargoNom</i>	smallint(3)	No	No	Clave del cargo que desempeña en administrativo
antiguedad	date	No	No	Fecha inicial del cargo
fechaRegistro	date	No	No	Fecha de ultima actualización

TDirección

Tabla que almacena los datos de contacto y dirección de oficinas internas y externas a la unam

Campo	Tipo	Nulo	Único	Comentarios
<u>CveDireccion</u>	int (7)	No	Si	Identificador único para datos de contacto de una persona
<i>NoEmpleado</i>	int(5)	No	Si	Atributo que hace referencia a un empleado
tipo	varchar(7)	No	No	Identificador del tipo de empleado al que se liga la dirección
calle	varchar(100)	No	No	Dirección de la oficina externa
numero	varchar(10)	No	No	Número exterior de la oficina
inter	varchar(7)	Si	No	Número interior de la oficina
col	varchar(60)	No	No	Colonia
delMun	varchar(60)	No	No	Clave de la delegación en la que se encuentra la oficina
edo	varchar(17)	No	No	Clave del estado en el que se encuentra la oficina
cp	varchar(5)	No	No	Código postal
tel	varchar(13)	Si	No	Teléfono de la oficina
tel2	varchar(13)	Si	No	Teléfono de la oficina
ext	varchar(7)	Si	No	Extensión de la oficina
ext2	varchar(7)	Si	No	Extensión de la oficina
Fax	varchar(13)	Si	No	Fax de la oficina
horario	varchar(250)	Si	No	Horario en el que se encuentra la persona en la oficina
email	varchar(60)	Si	No	Correo electrónico de oficina
imagen	varchar(126)	Si	No	Imagen relacionada con el edificio u oficina en la UNAM
fechaRegistro	date	No	No	Fecha de última actualización

TFuncionario

Tabla que almacena los datos de contacto de una persona como funcionario

Campo	Tipo	Nulo	Único	Comentarios
<u>CveFuncionario</u>	int(4)	No		Identificador único del registro por funcionario
<i>NoEmpleado</i>	int(5)	No		Número que asocia el empleado con su carácter de funcionario
<i>CveDependencia</i>	varchar(1)	No		Clave de la dependencia donde el empleado tiene un cargo de funcionario
<i>CveSubDep</i>	varchar(3)	No		Clave de la División o Secretaría en la que se desempeña el funcionario
<i>CveAreaDepto</i>	varchar(4)	Sí		Área o departamento a la que pertenece el funcionario
CveCargoFun	tinyint(3)	No		Clave del cargo que desempeña el funcionario
antiguedad	date	No		Fecha inicial del cargo
fechaRegistro	date	No		Fecha de última actualización

TRepresentante

Tabla que almacena los datos de contacto de una persona como representante

Campo	Tipo	Nulo	Comentarios
<u>CveRepresentante</u>	int(5)	No	Identificador unido del registro por representante
<i>NoEmpleado</i>	int(5)	No	Número que asocia el empleado con su carácter de representante
<i>CveDependencia</i>	varchar(1)	No	Clave de la dependencia donde el empleado esta adscrito
<i>CveSubDep</i>	varchar(3)	No	Clave de la división o secretaria en la que esta adscrito
<i>CveAreaDepto</i>	varchar(4)	Sí	Área o departamento a la que pertenece el representante
<i>CveCargoRep</i>	tinyint(2)	No	Clave del cargo que desempeña el representante
desde	date	Si	Fecha inicial del cargo
hasta	date	Si	Fecha final del cargo
CveCuerpoRep	smallint(2)	no	Clave del cuerpo colegiado o grupo de representación a la que pertenece
representandoA	varchar(60)	No	Grupo de personas a las que representa
comision	varchar(100)	No	Comisión a la que pertenece
fechaRegistro	date	No	Fecha de última actualización

Capítulo 4 Desarrollo

4.1 Estándares de desarrollo

Aquí se mostrarán los lineamientos seguidos para el desarrollo del sistema, la descripción de las pantallas, la descripción de las consultas para un usuario promedio y las funciones de un usuario con calidad de administrador local.

Descripción de los procesos

Antes de desarrollar el sistema se deben identificar los procesos que éste debe llevar a cabo, tal y como se planteó en el capítulo de Análisis y Diseño, sin embargo aquí se hace referencia debido a que se mostrará justamente el desarrollo y funcionamiento de estos.

A continuación se describe un proceso que es muy importante en el desarrollo del sistema debido a que es la pauta para poder utilizarlo, porque será la parte que los usuarios manipularán.

Diseño de la interfaz de usuario

El sitio está diseñado para mostrar varias secciones de manera que se presente el sistema de forma funcional, de nombre “Directorio en línea de Funcionarios y Académicos de la Facultad de Ingeniería”, ofrece principalmente información de datos de contacto al visitante, mediante dos diferentes tipos de consulta, una dirigida a visitantes de cualquier parte del mundo y otra sólo para administradores del sistema y personas autorizadas para ello debido a la confidencialidad de algunos de los datos contenidos en él.

Para explicar la estructura general, tomamos como referencia al índice maestro, que en el sitio es denominado técnicamente *Home* (Figura 4.1).

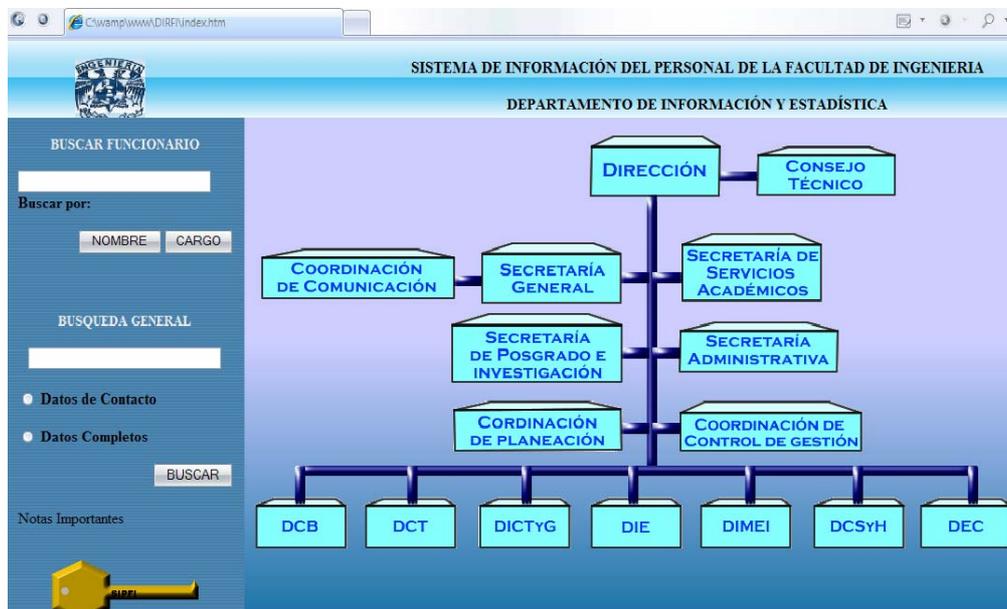


Figura 4.1 Página Principal

Home es la página que funciona como bienvenida y como índice maestro, en ella se encuentran referencias a otras secciones informativas, así como la existencia de un buscador general y un buscador específicamente para encontrar personas con carácter de funcionarios. La estructura principal de las referencias que se encuentran en esta página, se repite casi de manera idéntica en todas las secciones, de manera que se facilite la navegación para el visitante, ya que no necesitará regresar al índice para cambiar entre secciones.

Debido al propósito de facilidad en navegación, se dividen estas estructuras en diferentes secciones, cada página en tres áreas, y se explicará la estructura general de sistema de consulta con base en las áreas en cada página (Figura 4.2).

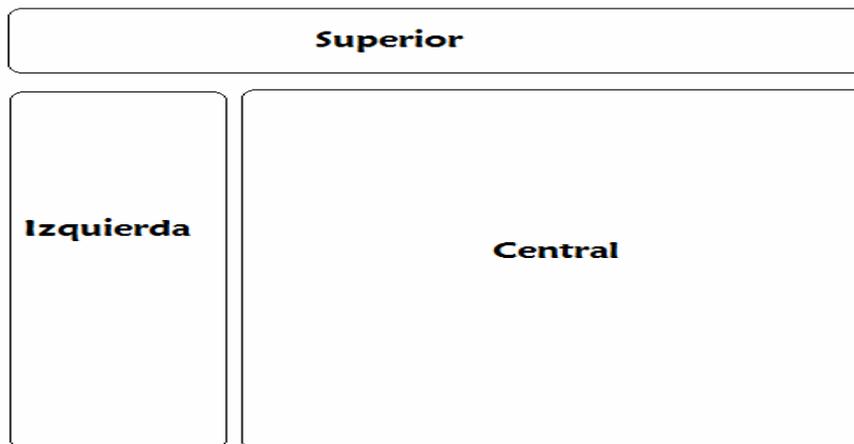
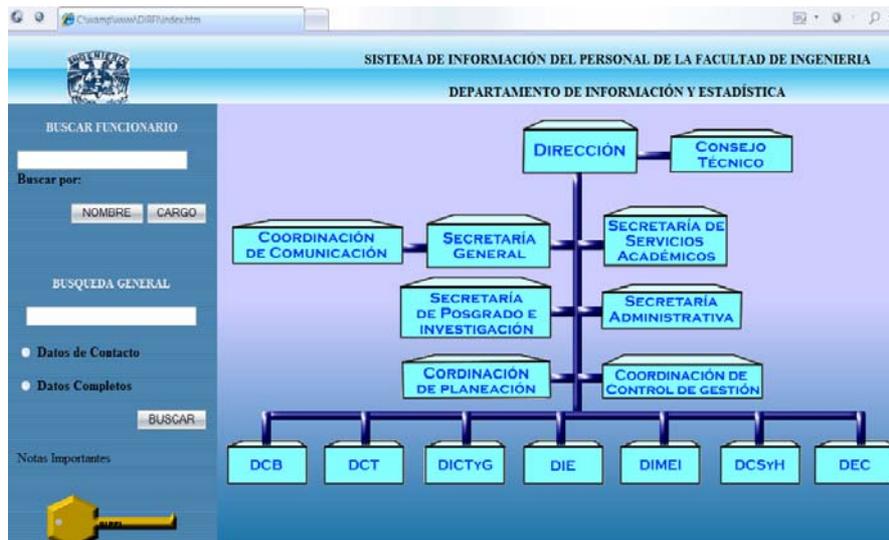


Figura 4.2 Secciones en página principal

Sección Izquierda

Esta área es una franja que cubre la porción izquierda de la página en un 20% a partir de la parte baja de la superior de la página. Se compone de los siguientes elementos:

Buscador: “Buscar Funcionario”. Aquí el usuario introduce el nombre o cargo de la persona con carácter de funcionario en quién está interesado y luego de elegir algún radio de opción: “Nombre” o “Cargo” y presionar “Buscar” obtendrá el resultado de su búsqueda en el área central de la página.

Buscador: “Búsqueda General”. Esta sección es útil al usuario para encontrar a cualquier persona adscrita a este directorio en línea, el tipo de búsqueda a elegir ya sea sólo datos de contacto o datos completos estará definida por el tipo de usuario que desee realizar la

consulta, siendo así que sólo los usuarios registrados como administradores del sistema y debidamente dotados de un Nombre de Usuario y Contraseña tendrán acceso a los datos completos de la persona que buscan.

Imagen “Llave”. Esta imagen está relacionada con la administración del sistema, de manera que al dar clic sobre ella se tendrá acceso a la primera pantalla de la administración del sistema, siendo esta la de identificación del usuario ante el sistema como Administrador.

Sección Superior

Esta parte contiene el encabezado del portal, la fecha y la hora del sistema además de los logotipos de la Facultad de Ingeniería y de la Universidad Nacional Autónoma de México que funcionan como ligas directas a las páginas principales de las mismas:

www.ingeniería.unam.mx y www.unam.mx

Sección Central

Esta parte contiene de manera original el Organigrama Principal de la Facultad de Ingeniería mediante el cual se podrá navegar a través de todos los suborganigramas para consultar los datos de contacto de las personas que lo conforman.

Además esta sección será la que contenga los resultados de las búsquedas que se realicen en la sección izquierda, de manera que ésta será la más dinámica de todas.

Tipos de Servicios

Los servicios que se ofrecen son básicamente relacionados con la búsqueda de información acerca del personal de la Facultad de Ingeniería.

Como se ha mencionado el tipo de información que se puede consultar dependerá del tipo de usuario que pretenda hacer una búsqueda. Esta información puede ser solamente datos de contacto o en una consulta para alguien con más privilegios de acceso, datos tales como los de tipo personal: dirección de su domicilio o datos relacionados con su situación actual

laboral, es decir el cargo o nombramientos que ocupa, materias que imparte, antigüedad laboral, etc.

La actualización, modificación y eliminación de información estará a cargo de las personas asignadas por la Facultad de Ingeniería, quienes tendrán acceso al sistema en calidad de administradores por lo que, la veracidad, confiabilidad y actualización de la información dependerán de estas personas, quienes sólo tendrán acceso a la gestión de información de la Secretaría o División Académica a la que sean asignados.

4.2 Desarrollo y manejo del BackOffice o administración del sistema

Utilización del Modelo Cliente/Servidor

El funcionamiento de las páginas web es básicamente el siguiente. Al introducir una dirección web lo que se hace es solicitar un archivo que se encuentra en una computadora (el servidor). El servidor enviará el archivo solicitado y la computadora que funge como cliente interpretará este archivo como una página web, desplegándolo mediante un navegador.

Si las páginas que se están solicitando son dinámicas, entonces el servidor es más complejo, se trata de dos bloques, uno que envía la página web y otro bloque que genera la información dinámicamente, en este caso se trata de una base de datos.

De manera general, el modelo cliente servidor trata de una relación entre procesos corriendo en máquinas separadas. Donde el servidor provee servicios y el cliente es un consumidor de servicios.

Cuando se divide y especializan los programas utilizando este modelo a fin de que cada tarea se realice y efectúe con la mayor eficiencia, también se permite simplificar las actualizaciones y el mantenimiento del sistema.

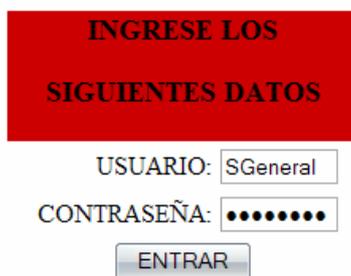
En los sistemas que trabajan utilizando el modelo cliente/servidor, el acceso es frecuentemente limitado para proteger ciertos datos en un servidor dado, ya que no hay un control físico para el uso de los datos por parte de múltiples clientes, excepto para controlar

el medio físico de almacenaje, sin embargo los usuarios comunes no tienen acceso a esta opción, de tal manera que la validación de usuario es uno de los procesos de protección del sistema.

Validación de usuario y contraseña

Es muy importante el papel que juega la validación de usuario, ya que este proceso permite el acceso a los datos que aloja el sistema. Aquí, los datos que se requieren son un Nombre de Usuario y una Contraseña que pueden ser proporcionados por el administrador del sistema. Cabe mencionar que para el sistema que nos atañe, este acceso estará previsto para que el usuario acceda a la Secretaría o División Académica que le corresponda.

Para comenzar este proceso el usuario debe de dar un clic en la imagen de la “llave” que se encuentra en la parte inferior de la sección izquierda de la página principal, con lo cual ingresará a una pantalla que le pedirá su Nombre de Usuario y Contraseña (figura 4.3).



INGRESE LOS
SIGUIENTES DATOS

USUARIO:

CONTRASEÑA:

Figura 4.3 Pantalla de validación de usuario

Cuando el usuario da clic en ENTRAR, se verifica que ambos datos hayan sido proporcionados. Se valida que sean del tipo correcto que en este caso será alfanumérico. Además se verifica que los datos proporcionados tengan acceso al sistema.

En caso de no cumplir con uno o más de estos puntos, el sistema retornará a la misma página pidiendo nuevamente los datos, esto hasta que sean correctos simultáneamente. En caso contrario, se identifica a que División o Secretaría corresponden, ya que el usuario sólo

puede ver la información referente a ésta, y se muestra la pantalla para elegir la acción a realizar.

Acciones que se pueden realizar en la administración del sistema

Después de haber pasado a través de la validación correctamente, el usuario deberá introducir el Número de Empleado para el cual desea realizar alguna acción. En esta misma pantalla finalmente escogerá que acción realizará mostrándose las opciones ALTA, BAJA Y MODIFICACIÓN (Figura 4.4). Cabe destacar que en esta pantalla el cuadro de texto donde se indica la Secretaría o División está solamente con fin informativo, pues éste no se puede modificar.

PROPORCIONE LOS DATOS REQUERIDOS Y LA ACCIÓN QUE DESEA REALIZAR

Número de Empleado:

División o Secretaría:

ALTA BAJA MODIFICAR

Figura 4.4 Pantalla elegir acción a realizar

ALTA

Este proceso tiene como finalidad capturar datos, referentes al personal que labora o representa en la Facultad de Ingeniería. Es necesario que el usuario proporcione los datos de entrada (formulario) para que estos puedan ser almacenados exitosamente en la base de datos.

El proceso de Alta comienza cuando el usuario ha seleccionado esta opción en la pantalla principal después de haberse identificado como administrador y da clic al botón COMENZAR, es entonces cuando se lleva a cabo el siguiente proceso:

Se visualiza una pantalla (Figura 4.5) en donde se informa que los datos que se ingresen deben ser sólo los correspondientes al personal que labora en la División para la cual el usuario esta autorizado y la información ahí vertida será responsabilidad de esta persona.

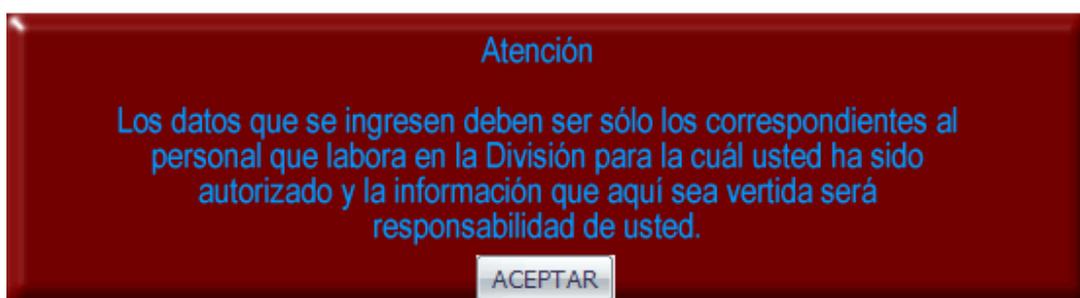


Figura 4.5 Pantalla mensaje a usuario permitido

Continuando con el proceso se debe dar clic en el botón ACEPTAR, para entonces poder visualizar la pantalla de la Figura 4.6, en donde aparece el formulario que contiene los campos vacíos para agregar la información correspondiente a datos personales como son: nombre, dirección particular, teléfono y correo electrónico personal; datos de contacto en oficina como: dirección de oficina, teléfonos y extensiones y horario; además de elegir de que tipo de miembro de la comunidad de la Facultad de Ingeniería se trata: Académico, Funcionario, Administrativo o en su caso Representante. Esto con el fin de continuar el registro de la persona en el formulario que le corresponde de acuerdo al tipo de miembro de la comunidad del que se trate.

INGRESE DATOS PERSONALES

Número de Trabajador:

Nombre:

▾

Prefijo

Nombre (s)

Apellido Paterno

Apellido Materno

Dirección Particular:

Calle

Num. Exterior

Num. Interior

Colonia

C.P.

Delegación o Municipio

Estado

Teléfono:

Teléfono

Fax

Móvil

Correo Electrónico:

INGRESE DATOS DE OFICINA

Agregar dirección de oficina fuera de la UNAM

Dirección Oficina:

Calle

Num. Exterior

Num. Interior

Colonia

C.P.

Delegación o Municipio

Estado

Teléfono:

Teléfono 1

Teléfono 2

Extensión

Fax

Correo Electrónico:

Horario:

AGREGAR DATOS COMO MIEMBRO DE LA COMUNIDAD

Agregar Datos Como:

Académico Funcionario

Administrativo Representante

BORRAR **SIGUIENTE**

Figura 4.6 Formulario Datos de Contacto

Una vez que haya comenzado a llenar el formulario se puede optar en cualquier momento por el botón BORRAR, con el cual el formulario se restaurará al estado inicial es decir un formulario limpio.

Si el usuario ha agregado los datos correctos y decide continuar con el alta de la persona, entonces deberá dar clic en siguiente para que de esta manera continúe el proceso, mientras tanto la información es enviada para almacenarse en la base de datos.

Dependiendo de la elección del tipo de miembro que haya hecho se accederá a uno de los siguientes formularios.

Formulario de datos de contacto para personal académico

Este formulario, representado en la figura 4.7, ha sido desarrollado de acuerdo a las características del tipo de información requerida para el personal académico.

**Formulario de Datos de Contacto
para Personal Académico**

Dependencia: Facultad de Ingeniería ▼

Division: Secretaría General ▼

Area o Departamento: Departamento de Información y Estadística ▼

Categoría: Ayudante A ▼ **Materia:**

Antigüedad:
aaaa/mm/dd

Otras

Horarios:

Figura 4.7 Formulario Datos de Contacto para Personal Académico

Permite ingresar datos de diferentes categorías de contratación de personal académico con el fin de que la información capturada sea lo más completa posible.

Además de que si se desea se puede seleccionar la opción Agregar Dirección que permitirá agregar la dirección de su oficina dentro de la UNAM.

El formulario de agregar dirección de oficina y datos de contacto dentro la UNAM (Figura 4.8) tiene la característica de además ofrecer la posibilidad de agregar una imagen referente a la ubicación de la oficina.

Agregar Dirección

Dirección de Oficina en la UNAM

Calle

no: int:

Colonia:

Delegación:

Estado: C.P.

Tel: Tel:

fax:

ext: ext:

horario

email:

imagen

Figura 4.8 Formulario Dirección de oficina en la UNAM

Después de agregar los datos que se piden en el formulario anterior, se oprime el botón Siguiente, mediante el cual la información será enviada para su almacenamiento en la Base de Datos, y visualizarse una pantalla como la siguiente.

Datos guardados exitosamente.

Número de Trabajador 29
Nombre Rafael Guillermo Suarez

<input checked="" type="radio"/> Nombre y grado <input checked="" type="radio"/> Dirección particular <input type="radio"/> Datos de Académico <input type="button" value="Modificar"/>	<input type="radio"/> Oficina ajena a la UNAM <input type="radio"/> Datos de Funcionario <input type="radio"/> Datos de Representante <input type="radio"/> Datos de Administrativo <input type="radio"/> Dirección UNAM <input type="button" value="Agregar"/>	<input type="radio"/> Datos de Académico <input type="button" value="Borrar"/>
--	--	---

Figura 4.9 Pantalla datos a Modificar, Agregar o Borrar

En esta última pantalla se observan tres columnas: la primera muestra que tipo de datos son los que tiene capturados esta persona y da la posibilidad de modificarlos mediante el botón de la parte inferior, la segunda columna nos muestra los tipos de datos que no han sido agregados y da la posibilidad de hacerlo mediante un botón llamado Agregar; en la última columna se observan los datos que se pueden borrar, cabe mencionar que el nombre, grado y dirección particular no se pueden borrar desde aquí sino con otro proceso llamado Baja.

Si optamos por modificar alguno de los datos, sólo seleccionamos que datos serán modificados y oprimimos el botón Modificar, con lo cual obtenemos el formulario correspondiente como es el caso siguiente del formulario para modificar datos de Funcionario (Figura 4.10).

MODIFICAR DATOS

Dependencia: Facultad de Ingeniería ▼

División: Ciencias Básicas ▼

Area o Departamento:

Otro ▼

Cargo:

Otro ▼

Antigüedad en el cargo:

aaaa/mm/dd

Figura 4.10 Formulario Modificar Datos Funcionario

Si lo que se desea es agregar datos al registro de una persona, simplemente se selecciona de la columna central los datos que se desean agregar y entonces se desplegará el formulario correspondiente como en este caso seleccionamos Datos de Funcionario damos clic en agregar y despliega el siguiente formulario (Figura 4.11).

Formulario de Datos de Contacto para Funcionarios

Dependencia: Facultad de Ingeniería

Division: Secretaría General

Area o Departamento: No Aplica

Carga: Director

Antigüedad en el cargo:
aaaa/mm/dd

Direccion de Oficina en la UNAM

Calle

no: int:

Colonia:

Delegación:

Estado: C.P.

Tel: Tel:

fax:

ext: ext:

horario

email:

imagen

Figura 4.11 Formulario Datos de Contacto Funcionario

Una vez concluida la captura en este formulario dar clic en Agregar para que entonces la información recabada sea almacenada en la base de datos, el sistema informa entonces que ha guardado la información al mismo tiempo que nos regresa a la pantalla anterior para realizar alguna acción relacionada con Agregar, Modificar y Borrar datos de la persona en cuestión.

Es entonces que, si se requiere borrar algunos de los datos que tiene esta persona en el sistema se debe seleccionar que información será borrada en la tercera columna y oprimir el botón Borrar, con lo cual aparecerá la siguiente pantalla (Figura 4.12) para confirmar los datos que se eliminarán, para este ejemplo se borrará la información de Datos de Oficina Ajena a la UNAM. Así que sólo hará falta oprimir el botón Borrar para que esta acción se realice. Es importante mencionar que esta pantalla de confirmación cuenta con un botón llamado Regresar como una característica de navegación para que el usuario pueda corregir su selección si es que le fuera necesario devolviéndolo a la pantalla anterior.

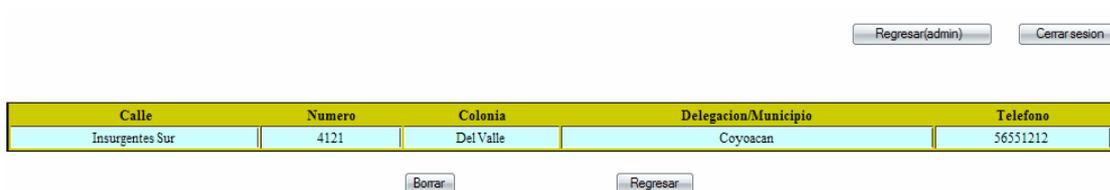


Figura 4.12 Pantalla Confirmar datos a Borrar

Como se observa, la captura de datos es sencilla y se puede realizar eficazmente ya que los datos se pueden Modificar, Agregar y Borrar con solo seleccionarlos después de haber elegido a la persona que se requiere.

Con las acciones descritas anteriormente, también podemos asegurar que una modificación en los datos se puede realizar en el momento que sea necesario al igual que agregar datos como en el caso de que la persona tenga un nuevo nombramiento o desee agregar información de contacto.

Ahora se regresa a la pantalla siguiente de la validación de usuario en donde existe como segunda opción a seleccionar la Baja (Figura 4.13).

PROPORCIONE LOS DATOS REQUERIDOS Y LA ACCIÓN QUE DESEA REALIZAR

Número de Empleado:

División o Secretaría:

ALTA BAJA MODIFICAR

Figura 4.13 Pantalla Acción a Realizar, seleccionando Baja

Aquí después de haber introducido el Número de Empleado y la División o Secretaría a la que pertenece, además de seleccionar la opción BAJA, solo falta dar clic en el botón COMENZAR para realizar esta acción, obteniendo la pantalla siguiente (Figura 4.14).

Desea ELIMINAR todos los registros de:

Número de Trabajador	Grado	Nombre	Apellido	Apellido
27	Doctor	José	Peréz	Ruiz

Figura 4.14 Pantalla confirmar Baja

En esta pantalla es posible visualizar el nombre de la persona que se seleccionó mediante su Número de Empleado para confirmar que se trata de la persona cuya información se quiere eliminar, de no ser así se puede salir de aquí sin realizar ninguna acción mediante el botón Regresar situado en la parte superior derecha. Si se desea continuar con este proceso dar clic en BORRAR, los registros referentes a esta persona serán eliminados de la base de datos y se observa en la pantalla un mensaje de confirmación diciendo “Registro Eliminado Exitosamente”.

Por otra parte si se opta por la tercera opción: MODIFICAR, sólo podrá continuar en el caso de un trabajador dado de alta con anterioridad, de no ser así el sistema enviará un mensaje para notificarle que el trabajador no existe. En caso contrario las opciones de modificación

serán las referentes a la pantalla siguiente de acuerdo a los datos que el trabajador ya tenga registrados.

o

Número de Trabajador 83452
Nombre Carmen Mendoza Miranda

<ul style="list-style-type: none"><input type="radio"/> Nombre y grado<input type="radio"/> Dirección particular<input type="radio"/> Oficina ajena a la UNAM <p><input type="radio"/> Datos de Académico</p> <p><input type="radio"/> Dirección UNAM</p> <p><input type="button" value="Modificar"/></p>	<ul style="list-style-type: none"><input type="radio"/> Datos de Funcionario<input type="radio"/> Datos de Representante<input type="radio"/> Datos de Administrativo <p><input type="button" value="Agregar"/></p>	<ul style="list-style-type: none"><input type="radio"/> Oficina ajena a la UNAM<input type="radio"/> Datos de Académico<input type="radio"/> Dirección UNAM <p><input type="button" value="Borrar"/></p>
---	---	--

Figura 4.9 Pantalla datos a Modificar, Agregar o Borrar

Como se puede observar el sistema ha sido desarrollado para ser funcional de manera que un usuario que comienza a usarlo puede familiarizarse pronto con los procesos y trabajar eficazmente con él.

Conclusiones y Recomendaciones

A raíz de la existencia de una situación problemática de tipo administrativo para que conformar un directorio de contactos del personal de la Facultad de Ingeniería, tarea costosa en tiempo, esfuerzo y horas hombre que se caracterizaba por falta de oportunidad, veracidad y utilidad de la información se sugirió una solución de tipo ingenieril basado en el desarrollo de un sistema, que ha cubierto la información del directorio de contactos, y además proporciona información adicional referente a la situación contractual de las personas listadas.

Esta solución ha ocupado conocimientos de ingeniería en computación contemplando el aprovechamiento de recursos humanos y materiales de la Universidad con el fin de colaborar en la inclusión de los servicios que ofrece la Facultad de Ingeniería en el mundo global por medio de Internet. La tecnología se ha empleado para crear un sistema en donde se tienen procesos que comienzan por la recopilación de información, la validación, el almacenamiento y la disponibilidad de la información en cualquier momento desde cualquier parte del mundo.

Los beneficios de la existencia de este sistema son tangibles ya que permite realizar actividades académicas con más eficiencia, disminuir pérdida de información, disminuir carga de trabajo en las áreas correspondientes y costos de papelería y logística.

El proceso ingenieril no termina aquí la tecnología (hardware y software) y las necesidades cambian y el sistema se debe ir adaptando a los tiempos mediante otros procesos necesarios y contemplados en la vida de un sistema como lo es el mantenimiento.

El empleo de los conocimientos adquiridos en la Facultad de Ingeniería ha sido de gran apoyo, no sólo porque se ha resuelto un problema de manera profesional sino porque se ha tenido el orgullo de colaborar en la vida y desarrollo de nuestra alma mater.

Mantenimiento

En el ciclo de vida del software existe una etapa que pareciera menos importante que las demás y que sin embargo es esencial ya que está ampliamente ligada con la supervivencia de un sistema que está ampliamente ligado a la mejora y a la optimización del mismo, para su permanencia futura y requiere adaptaciones y correcciones desde el momento en que se pone en marcha.

La fase de mantenimiento involucra cambios al software en el sentido de corregir defectos y dependencias encontradas durante su uso, además del mejoramiento de su utilidad y aplicabilidad.

La fase de mantenimiento puede contemplar cuatro tipos de cambios:

Corrección. Incluso cuando se han llevado a cabo las mejores actividades ofreciendo garantía de calidad, existe la posibilidad de que el cliente encuentre defectos en el software, es entonces cuando el mantenimiento correctivo cambia el software para corregir los defectos detectados durante su uso.

Adaptación. El mantenimiento adaptativo consiste en modificar el software para que sea compatible a los cambios de su entorno externo, debido a la constante evolución de la tecnología a través del tiempo ya sea en hardware o software.

Mejora. El mantenimiento perfectivo amplía el software más allá de sus requisitos funcionales originales. Conforme el paso del tiempo y el constante uso del software, el usuario puede estar interesado en funciones adicionales para que sean incorporadas en el mismo sistema con el objetivo de hacer más funcional, eficiente y completo su trabajo.

Preventivo. El objetivo de éste es facilitar el mantenimiento futuro del sistema (verificar condiciones, mejorar legibilidad).

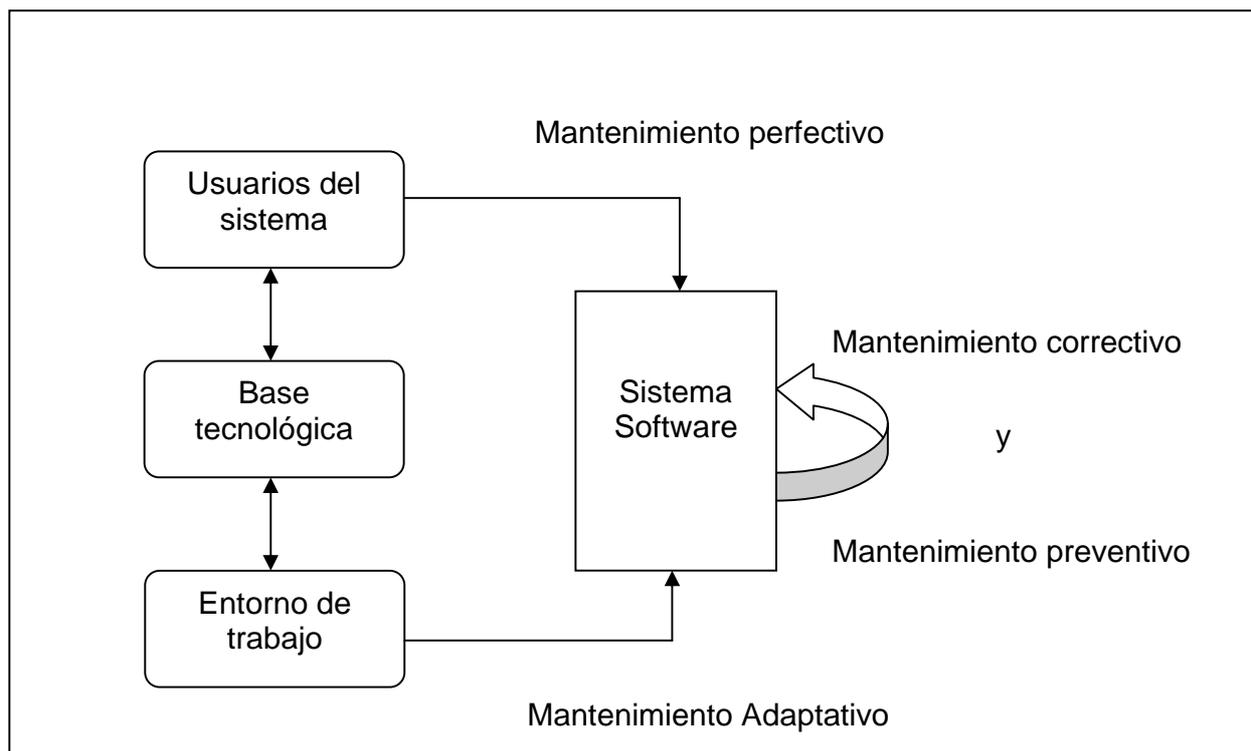


Figura A. Mantenimiento

El sistema debe ser de carácter intuitivo para un mejor manejo por parte de los usuarios finales, además de contar con la adecuada documentación permita resolver problemas tanto de funcionamiento y administración como de adecuación a posibles necesidades futuras. Para esto último, se debe monitorear el funcionamiento del sistema así como sus capacidades de uso y el comportamiento de los usuarios con el fin de realizar los ajustes necesarios.

Glosario

Apache: Servidor web bajo la filosofía Open Source. Desarrollado en 1995 por Rob McCool procurando mejorar el servidor que se tenía en el NCSA. El 60% de los servidores web del mundo lo utilizan.

Applet: Programa o aplicación pequeña creada en lenguaje Java.

Aplicación: Programas con los cuales el usuario final interactúa, es decir, son aquellos programas que permiten la interacción entre el usuario y la computadora. Esta comunicación se lleva a cabo cuando el usuario elige entre las diferentes opciones o realiza actividades que le ofrece el programa.

Atributo: En bases de datos es una característica que le interesa al programador sobre una entidad, representa alguna propiedad que es de utilidad almacenar.

Base de datos: Conjunto de datos organizados de tal forma que resulta fácil su gestión, actualización y acceso.

Browser: También llamado navegador o visualizador es el programa que permite leer documentos y seguir sus enlaces hacia otros documentos de Hipertexto.

Campo: En computación, es la unidad básica de las bases de datos. Se puede describir a un campo como cada uno de los distintos tipos de datos que componen una tabla. Para su definición se requiere, nombre, tipo de dato longitud de dato, etc.

Caracter: Número, letra o símbolo en la computadora, conformado por un byte.

Correo electrónico o email: Servicio de Internet que permite el envío de mensajes privados (semejantes al correo común) entre usuarios. Basado en el SMTP. Más rápido, económico y versátil que ningún otro medio de comunicación actual.

Dato: Unidad mínima de información no elaborada. Es la representación general de la característica de una entidad y por si sólo, el dato no tiene sentido, pero al procesarlo, es de utilidad en la toma de decisiones.

Diagrama de Flujo de Datos: Muestra el movimiento de datos a través de un sistema de procesamiento de datos. Es un modelo lógico-gráfico para representar el funcionamiento de un sistema. Los rectángulos representan entidades externas, los rectángulos abiertos almacenamiento, los círculos procesos y la flechas un flujo de datos desde cualquier elemento a un proceso o viceversa y se puede detallar más dividiendo algunos de sus procesos en subprocesos.

Entidad: Es un objeto que existe y que es distinguible de otros objetos, puede ser concreta, como una persona o puede ser abstracta como un algoritmo o una dirección.

Formulario: Conjunto de campos solicitados por un determinado programa, los cuales se almacenarán para su posterior uso o manipulación.

Frames: También llamados marcos, son las divisiones en las páginas de Internet, donde cada segmento contiene a su vez otra página.

Hardware: Componentes físicos de la computadora y sus periféricos.

HTML: Hyper Text Mark-up Language. Lenguaje estándar para crear páginas web, Consta de un grupo de etiquetas que describen el contenido y dan apariencia a las páginas en Internet.

Internet: Habitualmente llamada la red de redes mundial, es el conjunto de redes que se comunican entre sí a través de un protocolo llamado TCP/IP (Transmission Control Protocol/ Internet Protocol). Tiene sus inicios al final de los sesentas creada por el Departamento de Defensa de los Estados Unidos. Hoy en día es un espacio público utilizado por millones de personas alrededor del mundo como herramienta de comunicación e información.

Información: Conjunto organizado de datos, que constituyen un mensaje sobre una determinada entidad. Otra definición se puede dar de información es que es un fenómeno que proporciona significado o sentido a las cosas.

Ingeniería: Profesión en la cual los conocimientos de las matemáticas y las ciencias naturales obtenidos a través del estudio, la experiencia y la práctica, son aplicados con criterio y con conciencia al desarrollo de medios para utilizar económicamente con responsabilidad social y basados en una ética profesional, los materiales y las fuerzas de la naturaleza para beneficio de la humanidad.

Ingeniería de Software: Es el conjunto de métodos, técnicas y herramientas que controlan el proceso integral del desarrollo de software y suministra las bases para construir software de calidad de forma eficiente en los plazos adecuados.

Interfase: Elemento de transición o conexión que facilita el intercambio de datos.

ISO: International Organization for Standardization. Fundada en 1946, es una federación internacional que unifica normas en más de cien países.

Lenguaje de programación: Conjunto de normas lingüísticas que permiten escribir un programa y que éste sea entendido por la computadora.

ODBC: Open Data Base Connection o Conexión abierta a base de datos, es una forma de conexión a bases de datos independiente del lenguaje o programa que se utilice. Cada fabricante provee su propia librería con las características de conexión a las bases de datos.

OSI: Interconexión de Sistemas Abiertos, es el modelo de referencia universal para protocolos de comunicación, es la norma universal para protocolos de comunicación.

Plataforma: Es el principio, ya sea de hardware o software, sobre el cual un programa puede ejecutarse. Ejemplos típicos incluyen: arquitectura de hardware, sistema operativo, lenguajes de programación y sus librerías de tiempo de ejecución. La plataforma es el estándar con el

que los diseñadores de software escriben sus programas. Este término se utiliza para referirse al sistema operativo incluido con el hardware.

Portal: Sitio web que sirve de punto de partida para navegar por Internet. Los portales ofrecen una gran diversidad de servicios: listado de sitios web, noticias, e-mail, información meteorológica, chat, newgroups (grupos de discusión) y comercio electrónico.

Protocolo: Conjunto de reglas o procedimientos que controlan la secuencia de mensajes que ocurren durante una comunicación entre entidades que forman una red. Estos protocolos rigen formatos, modos de acceso, secuencias temporales, etc.

Relación: En bases de datos, es una asociación entre entidades que puede ocurrir entre dos entidades de un mismo conjunto de entidades o entre entidades de conjuntos distintos. Pueden existir relaciones entre más de dos conjuntos de entidades.

Sistema: Conjunto de elementos interrelacionados e interactuantes entre sí. El concepto tiene dos usos muy diferenciados, que se refieren respectivamente a los sistemas de conceptos y a los objetos reales más o menos complejos y dotados de organización.

Sistema de Información: Se puede definir como el conjunto de funciones y procedimientos encaminadas a la captación, desarrollo, recuperación, almacenamiento, etc., de información en el seno de una organización.

Sistema Operativo: Conjunto de programas que administra los recursos de una computadora tales como la memoria, la capacidad de procesamiento, las interrupciones y el envío de información a los periféricos entre otros.

Software: Conjunto de programas de distinto tipo (sistema operativo y aplicaciones) que hacen posible operar una computadora.

Usuario: Persona que hará uso de un sistema o aplicación.

Bibliografía

ANTECEDENTES

DOCUMENTOS OFICIALES

Informes Facultad de Ingeniería, UNAM. 2003 – 2005

FUENTES ELECTRÓNICAS

Portal de la Facultad de Ingeniería

www.ingenieria.unam.mx

Portal de la Secretaría General de la Facultad de Ingeniería

<http://sgeneral.fi-a.unam.mx>

METODOLOGÍAS PARA EL DESARROLLO DE SISTEMAS BIBLIOGRAFÍA

SOMERVILLE, Ian. *Ingeniería de Programación*. Editorial Addison-Wesley, 7ª Edición, Estados Unidos, 2005.

PRESUMAN, Rogers. *Ingeniería del Software, Un enfoque práctico*. Editorial Mc Graw Hill, 5ta. Edición, Madrid-España, 2002. 852 pp.

ZALDÍVAR ZAMORATEGUI, Orlando. *Apuntes de Ingeniería de Programación*. Facultad de Ingeniería UNAM, México, 2002.

Diagramas de Flujos de Datos

<http://www.gestiopolis.com/recursos2/documentos/fulldocs/fin/modproolgaarchivos/frame.htm>

LENGUAJES DE PROGRAMACIÓN

FUENTES ELECTRÓNICAS

PERL

<http://www.webtaller.com>

<http://www.oreilly.com>

<http://www.perl.com>

ASP Y .NET

www.microsoft.com/latam

www.mctekk.com/

www.devjoker.com

Libro Electrónico:

GONZÁLEZ SECO, José Antonio. *El Lenguaje de Programación C#*. 2001.

<http://www.josanguapo.com>

JSP Y PHP

[http://java.ciberaula.com/articulo/que_es_java/
www.desarrolloweb.com](http://java.ciberaula.com/articulo/que_es_java/www.desarrolloweb.com)

REDES DE COMPUTADORA

BIBLIOGRAFÍA

STOLZ, Kevin. *Todo Acerca de las Redes de Computadoras*. Editorial Prentice Hall Hispanoamericana, México, 1995.

ZALDÍVAR ZAMORATEGUI, Orlando. *Apuntes de Ingeniería de Programación*. Facultad de Ingeniería UNAM, México, 2002.

Tesis 110-04-49

Tesis 110-04-101

BASES DE DATOS

BIBLIOGRAFÍA

BATINI, CERI, NAVATHE. *Diseño Conceptual de Bases de Datos. Un enfoque de entidades-interrelaciones*. Editorial Addison-Wesley /Díaz de Santos, Madrid, 1994. 574 pp.

CONNOLLY, BEGG. *Database Systems. A Practical Approach to Design, Implementation and Management*. Editorial Addison-Wesley, 2ª Edición, 1998. 1374 pp.

DATE, C.J. *Introducción a los Sistemas de Bases de Datos*. Volumen I. Editorial Addison-Wesley Iberoamericana, 5ª Edición, 1995. (en inglés por Addison-Wesley)

ELMASRI, NAVATHE. *Sistemas de Bases de Datos. Conceptos Fundamentales*. Editorial Addison-Wesley Iberoamericana, 3ª Edición, 1999. (en inglés por Addison-Wesley)

FOLK, ZOELLICK. *File Structures*. Editorial Addison-Wesley, 2ª Edición, 1992. 195 pp.

HANSEN, G.W; HANSEN J.V. *Diseño y Administración de Bases de Datos*. Editorial Prentice Hall, 2ª Edición, 1997.

HERNÁNDEZ, Michael James. Colaboración de Kent Getz. *Database Design for Mere Mortals*. Editorial Addison-Wesley Developers Press, 2003. 672 pp.

MANEJADORES DE BASES DE DATOS

Manuales de Usuario de cada DBMS

<http://manuales.astalaweb.com/Manuales/UML.asp>