



**UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**“UTILIZACIÓN DE SOFTWARE LIBRE EN  
EL DESARROLLO DEL SISTEMA DE ACOPIO  
PERMANENTE PARA LA DIRECCIÓN  
GENERAL DE ESTADÍSTICA Y DESARROLLO  
INSTITUCIONAL DE LA UNAM”**

**T E S I S**

**QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN COMPUTACIÓN**

**P R E S E N T A :  
EDUARDO / ZARATE MEJIA**

**DIRECTORA DE TESIS:  
ING. LAURA SANDOVAL MONTAÑO**



**TESIS CON  
FALLA DE ORIGEN**

**MÉXICO, D.F. 2002**



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.

Con mucho cariño para toda mi familia y todos mis amigos.

## **Agradecimientos**

A la Universidad Nacional Autónoma de México y en especial a la Facultad de Ingeniería por la formación tan invaluable que me dieron.

A la Dirección General de Estadística y Desarrollo Institucional por permitirme ser y desarrollarme como profesional. Particularmente gracias a Longino y Jaime por darme esa oportunidad.

A mi directora de tesis, Ing. Laura Sandoval Montaña por el tiempo y el esfuerzo dedicado para hacer realidad este trabajo.

A los integrantes del jurado: Ing. Laura Sandoval Montaña, Ing. Lucila P. Arellano Mendoza, Ing. Juan J. Carreón Granados, Ing. Noé Cruz Marín y Fis. Raymundo H. Rangel González.

# Índice

INTRODUCCIÓN .....	1
<b>1. ANTECEDENTES</b>	
1.1    Conceptos básicos .....	4
1.1.1    Estadística .....	4
1.1.2    Indicador .....	5
1.1.3    Información estadística .....	4
1.2    Marco institucional .....	5
1.2.1    La estadística en la UNAM .....	5
1.3    La Dirección General de Estadística y Desarrollo Institucional (DGEDI) .....	6
1.3.1    Objetivo .....	6
1.3.2    Funciones .....	7
1.3.3    Organización .....	7
<b>2. PLANIFICACIÓN Y ANÁLISIS DEL SISTEMA DE ACOPIO PERMANENTE</b>	
2.1    Planteamiento del problema .....	10
2.2    Objetivos .....	11
2.3    Estrategia de desarrollo .....	11
2.4    Planificación .....	12
2.4.1    Estudio de factibilidad .....	12
2.4.2    Tiempo estimado de desarrollo .....	13
2.5    Análisis .....	14
2.5.1    Especificación de requerimientos .....	14
2.5.2    Análisis estructurado .....	15
2.5.3    Principales actividades del sistema .....	16
2.5.4    Diagrama de flujo de datos .....	19
2.5.5    Diccionario de datos .....	23
<b>3. DISEÑO DEL SISTEMA DE ACOPIO PERMANENTE</b>	
3.1    Especificación de procesos y procedimientos .....	49
3.2    Modelo de la base de datos .....	54
3.3    Diagramas entidad-relación .....	55
3.4    Diseño de la interfaz del usuario .....	66
<b>4. CONSTRUCCIÓN DEL SISTEMA DE ACOPIO PERMANENTE</b>	
4.1    Las aplicaciones web .....	81
4.2    Herramientas para acceder datos mediante el web .....	81
4.3    El software libre en el desarrollo de aplicaciones web .....	83
4.3.1    Sistema operativo Linux .....	84
4.3.2    Servidor web Apache .....	85
4.3.3    Manejador de bases de datos PostgreSQL .....	85
4.3.4    Lenguaje de programación PHP .....	86

4.4	Configuración del sistema operativo .....	88
4.5	Configuración del servidor web .....	91
4.6	Creación de la base de datos .....	94
4.7	Programación del sistema .....	121
4.8	Pruebas y mantenimiento .....	142
4.8.1	Pruebas .....	142
4.8.2	Mantenimiento .....	143
CONCLUSIONES .....		145
APÉNDICE A. CÓDIGO DE LA BASE DE DATOS .....		148
APÉNDICE B. GLOSARIO .....		177
APÉNDICE C. BIBLIOGRAFÍA .....		180
APÉNDICE D. SITIOS DE INTERÉS .....		181

## Introducción

En la actualidad la información estadística permite entre otras cosas, desarrollar políticas estatales e institucionales, así como prácticas orientadas al logro de las metas en instituciones de educación superior. Las estadísticas que generen dichas instituciones son de gran interés para múltiples usuarios cuando éstas ofrecen información valiosa, no sólo de números crudos de docencia, investigación, difusión cultural y servicios de apoyo a la comunidad, sino de la calidad de sus procesos educativos.

La tarea de la estadística en la Universidad Nacional Autónoma de México consiste hoy en proporcionar los indicadores y estadísticas que mejor comuniquen la relevancia y pertinencia de sus actividades, procesos y resultados, además de diseñar e instrumentar los procesos de evaluación para el mejoramiento de la institución en cada una de sus funciones substantivas.

La Dirección General de Estadística y Desarrollo Institucional (DGEDI) centra su labor en el acopio de información estadística relativa al funcionamiento de la Universidad y es la única instancia oficial que la proporciona de manera oportuna, suficiente y válida, con base en un sistema integral y confiable de información y procesos sistemáticos que brindan apoyo a la toma de decisiones y al desarrollo institucional.

Una de las principales funciones de la DGEDI consiste en mejorar y actualizar sus sistemas de acopio y procesamiento de la información estadística generada en la UNAM.

Con el propósito de cumplir con esta tarea en el año de 1998 la DGEDI, puso en marcha por primera vez el Sistema de Acopio Permanente (SAP), cuyo modelo de acopio y consulta de datos está sustentado en RedUNAM. El desarrollo de este sistema representó un avance importante hacia un modelo de acopio de información más dinámico y versátil.

Este sistema es alimentado por las distintas entidades académicas de la UNAM durante el proceso de acopio y desde entonces constituye la base para la construcción de los indicadores del desarrollo institucional como elementos de apoyo en los procesos de planeación y evaluación.

Si bien durante el periodo que ha estado funcionando el Sistema de Acopio Permanente se ha caracterizado por ser un sistema confiable y seguro con el manejo de los datos, se han observado también algunos problemas, el más importante está relacionado con el uso del sistema por parte de los usuarios, ya que la interfaz es poco clara y en algunas ocasiones un tanto confusa.

Por otra parte, de acuerdo a los requerimientos actuales que tiene la DGEDI, la estructura de la información procesada por el sistema tiene que ser actualizada, alguna de dicha información ya no será manejada, otra tiene que ser modificada, y otra más tiene que ser adicionada.

El objetivo de esta tesis es:

Desarrollar un sistema que mejore los mecanismos de acopio de información que realiza la DGEDI para poder contar con información estadística oportuna, confiable, suficiente y válida que brinde apoyo en la toma de decisiones, en la generación de indicadores institucionales, y en la construcción de los informes que le son requeridos por las distintas instancias.

Se pretende además lograr obtener un sistema de acopio que no sea únicamente seguro y confiable, sino que sea a la vez un sistema eficaz y funcional, para que permita hacer del proceso de acopio de información una tarea más simple.

La hipótesis que se plantea en este trabajo es que para poder construir un sistema que cumpla con los objetivos planteados se puede hacer uso de herramientas de software libre, que pueden cumplir con los requisitos de confiabilidad, seguridad, robustez y flexibilidad que el desarrollo del sistema necesita, y que deben de considerarse como alternativas bastante buenas con respecto a las aplicaciones propietarias.

En el capítulo 1 se presentan algunos conceptos básicos de la estadística, se establece cuál es la relevancia, y la trascendencia de la información estadística que se trabaja en la Universidad, la necesidad de contar con sistemas de información que hagan más ágil y flexible el manejo de dicha información, y por ello la importancia del desarrollo del Sistema de Acopio Permanente.

En el capítulo 2 se determina cuál es el problema que se tiene por resolver, se fijan los objetivos que se tienen que cumplir y se fija la estrategia de desarrollo para el desarrollo del sistema. La etapa de desarrollo que se trata principalmente en este capítulo, es la del análisis, se identifican los requerimientos y las principales actividades del sistema, con lo que se logran obtener el diagrama de flujo de datos, y el diccionario de datos.

En el capítulo 3, que trata sobre el diseño del sistema, se realiza la especificación de procesos y procedimientos con base en los diagramas de flujo de datos, se construye el diagrama entidad-relación para la posterior construcción de la base de datos, y se trabaja sobre el diseño de la interfaz del usuario.

El capítulo 4, la construcción del sistema, hace en primera instancia un breve estudio y análisis de las herramientas de software libre, y en particular de aquellas que son útiles en el desarrollo de aplicaciones web, mismas que resultan de suma importancia en la construcción del sistema. Se describe la instalación, configuración y uso del software utilizado, el procedimiento para la creación de la base de datos, y el mecanismo de creación de los programas y archivos necesarios para el funcionamiento del sistema.

La parte final de este trabajo consiste en presentar las conclusiones obtenidas durante la realización de cada una de las etapas de desarrollo del Sistema de Acopio Permanente, el análisis, el diseño y la construcción. Se platica cual ha sido la experiencia obtenida con la participación en este proyecto, y por último se realizan algunos comentarios relacionados a la hipótesis expuesta en esta tesis.



## CAPÍTULO 1

### ANTECEDENTES

## 1.1 Conceptos básicos

### 1.1.1 Estadística

La estadística es la rama de las matemáticas que se ocupa de reunir, organizar y analizar datos numéricos para ayudar a resolver problemas como el diseño de experimentos y la toma de decisiones. Constituye un medio para describir con exactitud datos físicos, económicos, políticos, sociales, psicológicos, entre otros, así como una herramienta para relacionar y analizar dichos datos.

Sin embargo, su importancia va más allá de la recolección y tabulación de la información, ya que su valor reside sobre todo en la interpretación experta de esa información, en la comprensión objetiva de los hechos y, particularmente, en las decisiones y acciones que se fundamentan en dicha información.

La estadística en las instituciones de educación superior tiene particular importancia, porque permite ver y analizar la situación actual que guarda el desarrollo de una institución, y permite sustentar las actividades de planeación, para programar las nuevas metas y compromisos.

### 1.1.2 Indicador

Los indicadores institucionales son algo más que una expresión numérica o un agregado de datos estadísticos, ya que tienen como finalidad informar sobre el funcionamiento de una institución a un público diverso: a los responsables de definir políticas, a organismos nacionales e internacionales, gubernamentales y no gubernamentales, a investigadores, profesores, estudiantes, y al público en general. Por consiguiente sirven también como fundamento para crear nuevas visiones y expectativas.

Para que los indicadores sean útiles deben ser válidos, relevantes, precisos, comprensibles, oportunos, comparables e interpretables. Por supuesto, disponer de datos estadísticos fiables es una condición indispensable para la construcción de indicadores.

Además de lo anterior, los indicadores deben de ser atractivos para una gran variedad de usuarios entre los cuales se encuentran los académicos, los expertos en estadística, los políticos, las autoridades responsables de mejorar las instituciones de educación superior y al público en general.

Una vez que cumplen con los requerimientos indispensables, los indicadores son útiles para informar sobre el estado que guarda una institución; dar seguimiento a los cambios que se producen en determinados periodos de tiempo; explicar causas de diferentes condiciones; predecir cambios probables; definir fortalezas y debilidades del sistema; informar a quienes toman decisiones sobre las formas más efectivas para mejorarlo; y para definir objetivos apropiados.

### 1.1.3 Información estadística

En la actualidad la información estadística permite desarrollar políticas estatales e institucionales, así como prácticas orientadas al logro de las metas de la educación superior. Hoy en día se dispone de más y mejor información para hacer juicios sobre educación superior, pero ésta debe difundirse no sólo a los especialistas y a los políticos, sino a la sociedad en general.

Las estadísticas que generen las instituciones de educación superior serán de gran interés para múltiples usuarios cuando éstas ofrezcan información valiosa, no sólo de números crudos de docencia, investigación, difusión cultural y servicios de apoyo a la comunidad, sino de la calidad de sus procesos y resultados educativos.

La educación superior debe demostrar públicamente su valor para los estudiantes, las empresas y la industria, para obtener un apoyo más amplio para el logro de sus objetivos.

## 1.2 Marco institucional

### 1.2.1 La estadística en la UNAM

La Universidad Nacional Autónoma de México tiene una larga tradición en la construcción de estadísticas. Desde la sexta década del siglo XX, la institución contaba con el Departamento de Estadística que funcionaba de manera muy elemental en la Secretaría Auxiliar de la Rectoría.

A partir de 1986 el Departamento de Estadística pasa de la Secretaría Administrativa a la Dirección General de Planeación como una Subdirección, hecho que le imprime un carácter más profesional y se buscan fuentes más confiables para la construcción de los datos. A partir de ese año se edita la Agenda Estadística que hoy es todavía la publicación oficial de la Institución.

A partir de 1993 la Subdirección de Estadística de la Dirección General de Planeación se convierte en la Dirección General de Estadística y Sistemas de Información Institucionales, que mejora sobre todo los aspectos de precisión, amplía horizontes y abre nuevos cauces de difusión al aprovechar Internet como una ventana adicional de acopio y publicación de datos e información.

En marzo del 2000, se crea la Dirección General de Estadística y Desarrollo Institucional, con adscripción a la Coordinación General de Reforma Universitaria, misma que fue creada en febrero del mismo año. Con esta nueva instancia, se fusionan las estructuras, recursos humanos, financieros y materiales, así como las funciones, de la Dirección General de Desarrollo Institucional y la Dirección General de Estadística y Sistemas de Información Institucionales.

En septiembre del 2001 se crea el Grupo Técnico de Responsables de Estadística y Planeación Institucional (GREPI), como sucesor del Grupo Técnico de Responsables de Estadística (RESPINFE), creado desde 1993. Este cuerpo asesor está constituido por los responsables de la estadística de cada una de las entidades académicas y dependencias administrativas.

El propósito de este cuerpo universitario es dar apoyo al Consejo de Planeación de la Universidad y asesoría a la Dirección General de Estadística y Desarrollo Institucional para concertar las acciones necesarias que faciliten las funciones de estadística, planeación y desarrollo institucional de la máxima casa de la Universidad.

Tiene también como objetivo establecer los criterios que permitan y aseguren permanentemente el acopio, flujo, uso, manejo y validación de la información, tanto al interior de las dependencias universitarias como la de carácter institucional de la Universidad Nacional Autónoma de México.

La tarea de la estadística en la Universidad consiste hoy en proporcionar los indicadores y estadísticas que mejor comuniquen la relevancia y pertinencia de sus actividades, procesos y resultados, además de diseñar e instrumentar los procesos de evaluación para el mejoramiento de la institución en cada una de sus funciones substantivas.

Muchas de las instituciones de educación superior públicas, entre ellas la Universidad se están viendo presionadas para realizar una mejor planeación de la aplicación de sus recursos financieros, para obtener cada vez mejores resultados de sus actividades educativas y de investigación, es decir, están siendo objeto de un creciente escrutinio no sólo por los gobiernos que las subsidian, sino por la sociedad en su conjunto.

La UNAM está comprometida con la rendición de cuentas y se encuentra técnicamente preparada para responder sobre sus logros y contribuciones ante las distintas audiencias: su propia comunidad, instancias hacendarias, legislativas, académicas y sociales. Desde hace varias décadas, sistemáticamente publica las estadísticas básicas de sus actividades sustantivas: docencia, investigación, difusión y administración universitarias

### **1.3 La Dirección General de Estadística y Desarrollo Institucional**

La Dirección General de Estadística y Desarrollo Institucional (DGEDI) centra su labor en el acopio de información estadística relativa al funcionamiento de la Universidad y es la única instancia oficial que la proporciona de manera oportuna, suficiente y válida, con base en un sistema integral y confiable de información y procesos sistemáticos que brindan apoyo a la toma de decisiones y al desarrollo institucional, tanto a las autoridades colegiadas como a las autoridades unipersonales.

#### **1.3.1 Objetivo**

La DGEDI tiene como objetivo promover y coordinar los esfuerzos del desarrollo institucional y su seguimiento estadístico, mediante diversos procesos participativos colegiados y permanentes.

### 1.3.2 Funciones

- ✓ Mejorar y actualizar permanentemente los mecanismos y sistemas de acopio y procesamiento de la información para generar los indicadores institucionales.
- ✓ Ser el conducto institucional para validar y difundir la información sobre indicadores y diversos elementos estadísticos del acontecer universitario.
- ✓ Fortalecer el área de estadística a través de estudios y proyectos innovadores que fomenten la cultura del uso y manejo de información estadística en las diversas instancias universitarias.
- ✓ Promover y apoyar un proceso permanente, participativo y colegiado de procesos de planeación y desarrollo institucional.
- ✓ Generar publicaciones periódicas con los reportes de información institucional, así como estudios y proyectos realizados por la dependencia.
- ✓ Constituirse como una fuente básica, bibliográfica y hemerográfica, en el área de planeación, evaluación y sistemas de información estadística de educación superior.

### 1.3.3 Organización

Desde marzo del 2000 la DGEDI está adscrita a la Coordinación General de Reforma Universitaria, y se constituye de la siguiente manera:

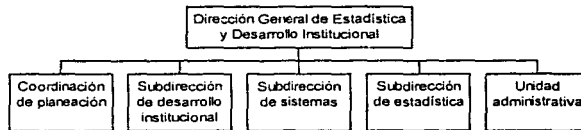


Figura 1.1 Organigrama DGEDI.

#### a) Coordinación de planeación

Proporciona el apoyo continuo que le solicite la Dirección de la dependencia en las labores de planeación y evaluación internas.

#### b) Subdirección de desarrollo institucional

Promueve y apoya un proceso permanente, participativo y colegiado de planeación y evaluación institucional a través de la asesoría requerida en estas labores, a las entidades académicas que así lo soliciten.

c) Subdirección de sistemas

Diseña, desarrolla, mantiene y supervisa la operación de los sistemas de información para el acopio, procesamiento y difusión en medios digitales, de la información estadística institucional.

d) Subdirección de estadística

Analiza la información estadística recabada, y la procesa para la creación de los indicadores institucionales. Genera estudios y proyectos que permitan difundir dicha información. Fomenta y difunde la cultura del uso y manejo de la información estadística en las diversas instancias universitarias, para el apoyo de la toma de decisiones, así como de los procesos de desarrollo y seguimiento institucional.

e) Unidad administrativa

Es la encargada de la administración de los recursos humanos y técnicos con los que cuenta la DGEDI.

## CAPÍTULO 2

# PLANIFICACIÓN Y ANÁLISIS DEL SISTEMA DE ACOPIO PERMANENTE

## 2.1 Planteamiento del problema

Una de las principales funciones de la Dirección General de Estadística y Desarrollo Institucional (DGEDI) consiste en mejorar y actualizar sus sistemas de acopio y procesamiento de la información estadística generada en la UNAM.

Con el propósito de cumplir con esta tarea en el año de 1998 la DGEDI<sup>1</sup>, puso en marcha por primera vez el Sistema de Acopio Permanente (SAP), cuyo modelo de acopio y consulta de datos está sustentado en RedUNAM. El desarrollo de este sistema representó un avance importante hacia un modelo de acopio de información más dinámico y versátil.

Este sistema es alimentado por las distintas entidades académicas de la UNAM durante el proceso de acopio y desde entonces constituye la base para la construcción de los indicadores del desarrollo institucional como elementos de apoyo en los procesos de planeación y evaluación.

El Sistema de Acopio Permanente fue diseñado como una aplicación web siguiendo un modelo trifásico. De acuerdo con este modelo, tres procesos separados y perfectamente definidos se ejecutan en distintas plataformas:

1. La interfaz gráfica de usuario (GUI), el navegador, que se ejecuta en la computadora del usuario.
2. El programa o programas de aplicación que se ejecutan en el servidor web y que se encargan de procesar los datos.
3. Un sistema de base de datos que almacena los datos que se requieren.

Este modelo presenta una serie de ventajas frente a los modelos tradicionales de una o dos fases. En particular para las aplicaciones web cabe destacar:

- ✓ Los navegadores web están presentes en cualquier parte, por lo que puede accederse a las aplicaciones desde cualquier plataforma.
- ✓ Las aplicaciones pueden compartir un mismo aspecto y finalidad.
- ✓ Su estructura en forma de módulos facilita la modificación o sustitución de uno de los módulos sin que ello afecte a los demás.

Si bien durante el periodo que ha estado funcionando el Sistema de Acopio Permanente se ha caracterizado por ser un sistema confiable y seguro con el manejo de los datos, se han observado también algunos problemas, el más importante está relacionado con el uso del sistema por parte de los usuarios, ya que la interfaz es poco clara y en algunas ocasiones un tanto confusa, en otras palabras, tiene problemas de usabilidad<sup>2</sup>.

---

<sup>1</sup> En el año de 1998, la DGEDI era todavía la Dirección General de Estadística y Sistemas de Información Institucionales.

<sup>2</sup> La facilidad de aprendizaje y utilización de un sistema de información es lo que se denomina usabilidad.



Por otra parte, de acuerdo a los requerimientos actuales que tiene la DGEDI, la estructura de la información procesada por el sistema tendrá que ser actualizada, alguna de dicha información ya no será manejada, otra tiene que ser modificada, y otra más tiene que ser adicionada.

## 2.2 Objetivos

Desarrollar un sistema que mejore los mecanismos de acopio de información que realiza la DGEDI para poder contar con información estadística oportuna, confiable, suficiente y válida que brinde apoyo en la toma de decisiones, en la generación de indicadores institucionales, y en la construcción de los informes que le son requeridos por las distintas instancias.

Lograr obtener un sistema de acopio que no sea únicamente seguro y confiable, sino que sea a la vez un sistema eficaz y funcional, para que permita hacer del proceso de acopio de información una tarea más simple.

## 2.3 Estrategia de desarrollo

El desarrollo de sistemas de información es una tarea que se puede manejar como un proyecto; es decir, debe tener un inicio y un fin, así como contar con una secuencia de pasos o etapas.

La estrategia elegida para el desarrollo del Sistema de Acopio Permanente consistió en seleccionar la metodología denominada "Método de ciclo de vida del desarrollo de sistemas", pues se consideró que ésta permitía cumplir con los objetivos planteados de forma efectiva.

A continuación se listan algunas de las características del método de ciclo de vida para el desarrollo de sistemas:

- ✓ Es el conjunto de actividades que realizan los analistas, diseñadores y usuarios para desarrollar e implantar sistemas de información.
- ✓ Este método concibe el desarrollo de sistemas como un ciclo específico de actividades.
- ✓ Los analistas no están de acuerdo respecto al número exacto de etapas que conforman este método; sin embargo, reconocen la importancia de su enfoque sistemático.
- ✓ Aunque cada etapa se encuentra definida, nunca se lleva a cabo como un elemento independiente.
- ✓ Es recomendable para proyectos de gran escala y cuando se tienen los procedimientos bien establecidos.

De manera general, se puede decir que este método cuenta con las siguientes etapas:

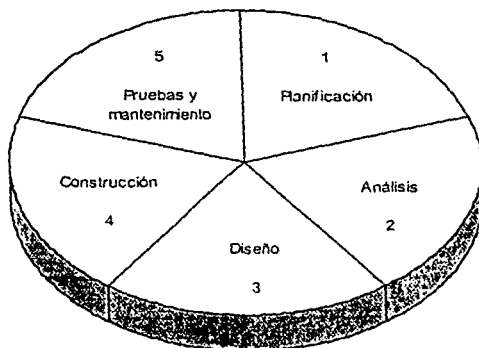


Figura 2.1 Método de ciclo de vida del desarrollo de sistemas.

TESIS CON  
 FALLA DE ORIGEN

## 2.4 Planificación

La creación de un sistema de información requiere el mismo grado de planificación que cualquier proyecto importante, de tal manera que se puedan prevenir las cosas que pueden fallar y cómo pueden solucionarse.

La planificación incluye también a todas las actividades que se requieren para la selección del equipo de desarrollo, la asignación de tareas apropiadas a los miembros de dicho equipo, la estimación del tiempo que cada tarea requiere para su ejecución, y la programación del proyecto, de tal forma que las tareas se concluyan oportunamente.

### 2.4.1 Estudio de factibilidad

Un estudio de factibilidad es una breve mirada a los principales factores que influirán en la capacidad del sistema para alcanzar los objetivos deseados. A continuación se muestra el estudio de factibilidad que se llevó a cabo para determinar que era posible desarrollar un sistema de información que permitiera alcanzar los objetivos deseados.

El estudio de factibilidad se realizó desde las siguientes perspectivas:

#### a) Factibilidad operativa.

- ✓ El sistema promete mejorar en muchos aspectos los mecanismos de acopio de información.
- ✓ Con el nuevo sistema se verá beneficiada la DGEDI, así como los responsables de la información estadística de las distintas entidades académicas de la UNAM, principales usuarios del sistema.

## b) Factibilidad técnica.

- ✓ No hay ninguna duda en que se cuenta con la tecnología necesaria para cubrir con las demandas del proyecto.
- ✓ Una vez puesto en marcha el sistema, éste puede crecer y ser actualizado con facilidad.
- ✓ Existen las garantías técnicas para asegurar la confiabilidad y seguridad de los datos.

## c) Factibilidad económica o financiera

- ✓ El costo del sistema será muy bajo puesto que la DGEDI cuenta ya con equipo de cómputo que puede ser utilizado para el desarrollo del sistema y la adquisición del software que se utilizará no constituye ningún costo adicional.
- ✓ El desarrollo del sistema será llevado a cabo por el personal del departamento de la Subdirección de Sistemas de la DGEDI por lo que tampoco habrá gastos adicionales en la contratación de personal.

## 2.4.2 Tiempo estimado de desarrollo

Mediante el siguiente diagrama de Gantt, se especifica cuál es el tiempo estimado de desarrollo del Sistema de Acopio Permanente.

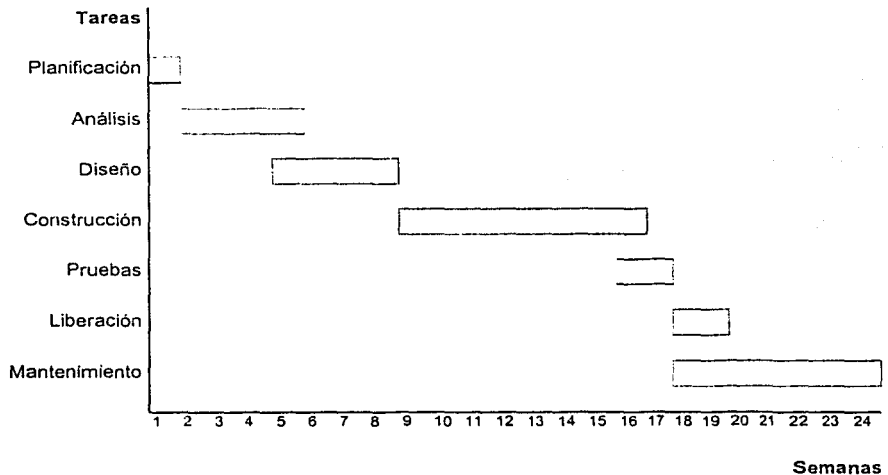


Figura 2.2 Diagrama de Gantt para el tiempo estimado de desarrollo del sistema.

## 2.5 Análisis

Una vez completada la fase de planificación, el siguiente paso es la fase de análisis. El análisis de sistemas es el estudio de un sistema existente con el propósito de diseñar un sistema nuevo o mejorarlo con procedimientos más adecuados.

El análisis es el proceso de clasificar e interpretar hechos, diagnosticar problemas y emplear la información para hacer las mejoras necesarias a un sistema. El análisis de sistemas de información específica qué es lo que el sistema debe hacer.

### 2.5.1 Especificación de requerimientos

Un requerimiento es una característica que debe incluirse en un nuevo sistema. La determinación de requerimientos es el estudio de un sistema para conocer cómo trabaja y dónde es necesario efectuar mejoras. La determinación de requerimientos vincula el estudio de un sistema existente con la recopilación de detalles relacionados con él.

Los datos obtenidos durante el estudio de un sistema se analizan para determinar las especificaciones de los requerimientos, es decir la descripción de las características del nuevo sistema.

A continuación se enumera la lista de requerimientos para el Sistema de Acopio Permanente:

1. El sistema debe de ser capaz de facilitar el proceso de acopio de información que realiza la DGEDI.

Una de las tareas más importantes para el nuevo sistema consiste en mejorar de manera sustancial la usabilidad del mismo. Se tiene que mejorar la navegación y el uso del sistema, para que resulte más eficaz, y el proceso de acopio sea más dinámico y versátil.

2. El acopio de información deberá ser mediante procesos automatizados.

La experiencia con el uso del Sistema de Acopio Permanente demostró que el proceso de acopio de información puede lograrse mediante procesos automatizados. El nuevo sistema debe de seguir cumpliendo con esta característica.

3. El sistema debe de proporcionar datos confiables y oportunos.

La seguridad, la integridad y la confiabilidad de la información manejada por el sistema debe de seguir siendo un punto importante que debe de conservar el nuevo sistema, inclusive, se tienen que examinar los mecanismos de acopio para verificar que la información que sea capturada es confiable.

El sistema debe de generar reportes que proporcionen información oportuna y válida para que pueda ser útil en la construcción de los indicadores institucionales, y en el proceso de toma de decisiones.

4. El sistema debe de proporcionar acceso al sistema sólo a los responsables de información estadística.

Los responsables de información estadística de las distintas entidades académicas de la UNAM son los principales usuarios del sistema, por lo que se debe de garantizar que únicamente éstos cuenten con acceso al sistema y que puedan realizar la tarea del proceso de acopio sin ningún inconveniente.

5. La información generada por el sistema debe de ser útil para la creación de las publicaciones periódicas realizadas en la DGEDI, tal como la Agenda Estadística, así como también para difundir la información estadística institucional a través del sitio web de la DGEDI.
6. La estructura de la organización de la información tiene que ser modificada.

Debido a los nuevos requerimientos de información que tiene la DGEDI, la organización de la información en la base de datos tiene que ser modificada. Alguna de esta información ya no será útil, otra tendrá que ser modificada, y otra más tiene que ser agregada.

Todos estos cambios tienen que hacerse de tal manera que permitan conservar de manera segura y confiable la información que aún será necesaria, y que la información que va a ser agregada no afecte el funcionamiento general del sistema.

7. Para el desarrollo del sistema se deben de seguir utilizando las mismas herramientas de software.

Durante el tiempo que lleva funcionando el Sistema de Acopio Permanente, se ha podido observar que las herramientas de software que se escogieron para el desarrollo, incluyendo sistema operativo, servidor de web, servidor de base de datos, y lenguaje de programación, fueron una adecuada elección.

Nunca se han tenido problemas graves, ni ha habido "caídas del sistema" ha causa del uso del software, por lo que se considera conveniente seguir utilizando las mismas herramientas de software en el desarrollo del nuevo sistema.

### 2.5.2 Análisis estructurado

Aunque el método del ciclo de vida fue base para el desarrollo del Sistema de Acopio Permanente, también fue utilizado el "Método de desarrollo por análisis estructurado".

El método de desarrollo por análisis estructurado es especialmente útil para facilitar la comprensión de sistemas grandes y complejos. Para lograrlo, se divide el sistema en componentes y se construye un modelo del sistema.

Este método permite que los usuarios observen los elementos lógicos separados de los elementos físicos. Los elementos lógicos se refieren a las funciones del sistema. Los elementos físicos son las computadoras, sistemas de almacenamiento, etcétera. Como este método sólo enfatiza la funcionalidad del sistema, fue utilizado en combinación con el método del ciclo de vida para el desarrollo de sistemas.

### 2.5.3 Principales actividades del sistema

La información estadística institucional que acopia el sistema, procede de las entidades académicas universitarias que se listan a continuación.

- ✓ Casa del Lago
- ✓ Centro Coordinador y Difusor de los Estudios Latinoamericanos
- ✓ Centro de Ciencias de la Atmósfera
- ✓ Centro de Ciencias de la Materia Condensada en Ensenada B. C.
- ✓ Centro de Ciencias Físicas
- ✓ Centro de Enseñanza de Lenguas Extranjeras
- ✓ Centro de Enseñanza para Extranjeros
- ✓ Centro de Estudios sobre la Universidad
- ✓ Centro de Instrumentos
- ✓ Centro de Investigación en Energía
- ✓ Centro de Investigaciones Interdisciplinarias en Ciencias y Humanidades
- ✓ Centro de Investigaciones sobre América del Norte
- ✓ Centro de Investigación sobre Fijación de Nitrógeno
- ✓ Centro de Neurobiología en Querétaro
- ✓ Centro Regional de Investigaciones Multidisciplinarias de la UNAM
- ✓ Centro Universitario de Estudios Cinematográficos
- ✓ Centro Universitario de Investigaciones Bibliotecológicas
- ✓ Centro Universitario de Teatro
- ✓ Colegio de Ciencias y Humanidades
- ✓ Consejo Técnico y Coordinación de Investigación Científica
- ✓ Coordinación de Difusión Cultural
- ✓ Coordinación y Consejo Técnico de Humanidades
- ✓ Dirección de Literatura
- ✓ Dirección de Teatro y Danza
- ✓ Dirección General de Actividades Cinematográficas
- ✓ Dirección General de Actividades Musicales
- ✓ Dirección General de Artes Plásticas
- ✓ Dirección General de Asuntos del Personal Académico
- ✓ Dirección General de Divulgación de la Ciencia
- ✓ Dirección General de Radio UNAM
- ✓ Dirección General de Servicios de Cómputo Académico
- ✓ Dirección General de Televisión Universitaria
- ✓ ENP: Antonio Caso
- ✓ ENP: Erasmo Castellanos Quinto
- ✓ ENP: Ezequiel A. Chávez
- ✓ ENP: Gabino Barreda
- ✓ ENP: José Vasconcelos
- ✓ ENP: Justo Sierra

- ✓ ENP: Miguel E. Schulz
- ✓ ENP: Pedro de Alba
- ✓ ENP: Vidal Castañeda y Nájera
- ✓ Escuela Nacional de Artes Plásticas
- ✓ Escuela Nacional de Enfermería y Obstetricia
- ✓ Escuela Nacional de Estudios Profesionales Acatlán
- ✓ Escuela Nacional de Estudios Profesionales Aragón
- ✓ Escuela Nacional de Estudios Profesionales Iztacala
- ✓ Escuela Nacional de Música
- ✓ Escuela Nacional de Trabajo Social
- ✓ Escuela Nacional Preparatoria
- ✓ Facultad de Arquitectura
- ✓ Facultad de Ciencias
- ✓ Facultad de Ciencias Políticas y Sociales
- ✓ Facultad de Contaduría y Administración
- ✓ Facultad de Derecho
- ✓ Facultad de Economía
- ✓ Facultad de Estudios Superiores Cuautitlán
- ✓ Facultad de Estudios Superiores Zaragoza
- ✓ Facultad de Filosofía y Letras
- ✓ Facultad de Ingeniería
- ✓ Facultad de Medicina
- ✓ Facultad de Medicina Veterinaria y Zootecnia
- ✓ Facultad de Odontología
- ✓ Facultad de Psicología
- ✓ Facultad de Química
- ✓ Instituto de Astronomía
- ✓ Instituto de Biología
- ✓ Instituto de Biotecnología
- ✓ Instituto de Ciencias del Mar y Limnología
- ✓ Instituto de Ciencias Nucleares
- ✓ Instituto de Ecología
- ✓ Instituto de Física
- ✓ Instituto de Fisiología Celular
- ✓ Instituto de Geofísica
- ✓ Instituto de Geografía
- ✓ Instituto de Geología
- ✓ Instituto de Ingeniería
- ✓ Instituto de Investigaciones Antropológicas
- ✓ Instituto de Investigaciones Bibliográficas
- ✓ Instituto de Investigaciones Biomédicas
- ✓ Instituto de Investigaciones Económicas
- ✓ Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas
- ✓ Instituto de Investigaciones en Materiales
- ✓ Instituto de Investigaciones Estéticas
- ✓ Instituto de Investigaciones Filológicas
- ✓ Instituto de Investigaciones Filosóficas
- ✓ Instituto de Investigaciones Históricas
- ✓ Instituto de Investigaciones Jurídicas
- ✓ Instituto de Investigaciones Sociales
- ✓ Instituto de Matemáticas
- ✓ Instituto de Química

- ✓ Museo Universitario del Chopo
- ✓ Programa Universitario de Alimentos
- ✓ Programa Universitario de Energía
- ✓ Programa Universitario de Estudios sobre la Ciudad
- ✓ Programa Universitario de Investigaciones en Salud
- ✓ Programa Universitario de Medio Ambiente

Cada elemento de la lista anterior, se identifica en el sistema como **dependencia**, y tiene la responsabilidad de proporcionar la información que le es requerida por la DGEDI.

Para facilitar el uso y manejo de la información el sistema clasifica la información de la siguiente manera:

- ✓ Proyectos de investigación
- ✓ Productos editoriales
- ✓ Publicaciones del personal
- ✓ Planta académica
- ✓ Actividades académicas
- ✓ Actividades de difusión
- ✓ Actividades de comunicación
- ✓ Actividades de educación continua
- ✓ Premios y distinciones recibidos
- ✓ Premios y distinciones otorgados

Cada elemento de esta lista se identifica en el sistema como un **módulo**.

El sistema valida a cada dependencia, antes de permitirle realizar cualquier tipo de operación en cada uno de estos módulos. Una dependencia siempre puede escoger sobre cuál módulo va a trabajar.

La operaciones que el sistema permite realizar a una dependencia son siempre las mismas. Se pueden dar de alta nuevos registros, eliminar registros, o actualizar registros.

El sistema proporciona a la dependencia los reportes de cada uno de estos módulos para verificar que la información que ha incorporado es la correcta.

El sistema proporciona a la DGEDI los reportes globales sobre la información que ha incorporado cada dependencia en cada uno de los módulos.



### 2.5.4 Diagrama de flujo de datos

El diagrama de flujo de datos (DFD) es una herramienta que permite representar un sistema como una red de procesos funcionales conectados entre sí por "flujos" de datos y "almacenes" de datos.

El DFD da más importancia a las funciones de un sistema de información que a los datos que maneja, por eso se dice que está orientado a las funciones. Los diagramas de flujo de datos se pueden dibujar con sólo cuatro notaciones sencillas. El uso de esta notación para cada elemento depende de qué enfoque se utilice, el de Yourdon o el de Gane y Sarson.

La siguiente tabla muestra los componentes del diagrama de flujo de datos y sus notaciones más comunes:







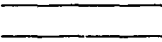
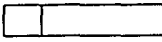
	Yourdon	Gane/Sarson
Proceso		
Flujo de datos		
Origen / destino		
Almacén		

Tabla 2.1 Elementos en un diagrama de flujo de datos.

**Proceso.** Muestra la parte del sistema que transforma las entradas de datos en salidas. Puede ser utilizado como sinónimo de función.

El proceso se identifica con un número y un nombre que se compone de un verbo-objeto para describir lo que hace. El número del proceso es para su identificación, no para definir una secuencia.

**Flujo de datos.** El flujo describe el movimiento de paquetes de datos de una parte del sistema a otra. Representa datos en movimiento.

El flujo de datos se representa con una flecha porque los datos viajan en una dirección específica desde su origen hacia su destino. Todo flujo debe tener un nombre que define la información que transporta. La única excepción a esta regla es cuando el flujo se dirige a un almacén, ya que se supone que transportará la información guardada en él.

**Almacén.** El almacén se utiliza para representar una colección de paquetes de datos en reposo. No se refiere exclusivamente a los medios de almacenamiento electrónicos como bases de datos en discos duros o discos ópticos, sino también a archivos metálicos o cualquier medio que permita guardar datos en tarjetas u hojas de papel.

Los flujos que van hacia el almacén representan los datos que serán guardados en él. Los flujos que vienen desde el almacén representan los datos que serán extraídos de él.

El almacén se identifica con un nombre que representa su contenido.

**Origen/destino de los datos.** Representa entidades externas al sistema que se comunican con él y que están fuera de su control. También se les conoce como terminadores.

Un terminador puede ser una persona o grupo de personas, por ejemplo, una organización o un departamento. El origen/destino de los datos se identifica con un rectángulo y un nombre representativo de la entidad.

Es importante tomar en cuenta que los terminadores:

- ✓ Son externos al sistema.
- ✓ No se pueden cambiar sus contenidos ni funciones.

**Niveles de un DFD.** El diagrama de flujo de datos representa el sistema de información desde lo general hacia lo específico.

Los niveles de un DFD señalan el grado de detalle de la representación del sistema:

- a) El diagrama de contexto. Este diagrama muestra el sistema como un solo proceso para hacer énfasis en los terminadores y los flujos de datos.
- b) El diagrama de nivel 0. Este diagrama muestra los procesos más importantes del sistema para resaltar la transformación de los datos.
- c) Los diagramas de nivel 1. Desde los diagramas de nivel 1 y subsecuentes se describirán los procesos de manera más detallada.

Utilizando la notación de Gane/Sarson, a continuación se muestra el diagrama de contexto, el diagrama de nivel 0, y los diagramas de nivel 1 para el Sistema de Acopio Permanente.

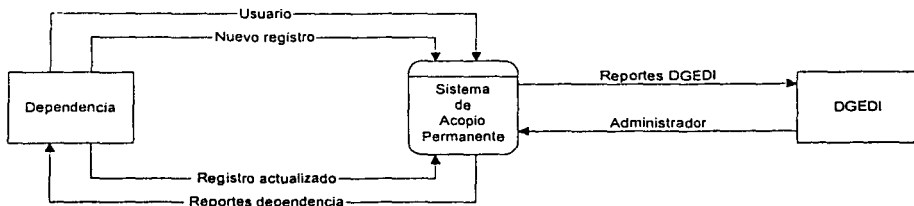


Figura 2.3 Diagrama de contexto, SAP.

La figura 2.4 muestra el diagrama de nivel 0. Como se podrá observar, los flujos que van de y hacia los almacenes no se han nombrado. Se está haciendo la consideración que transportan los datos que van de y hacia ellos.

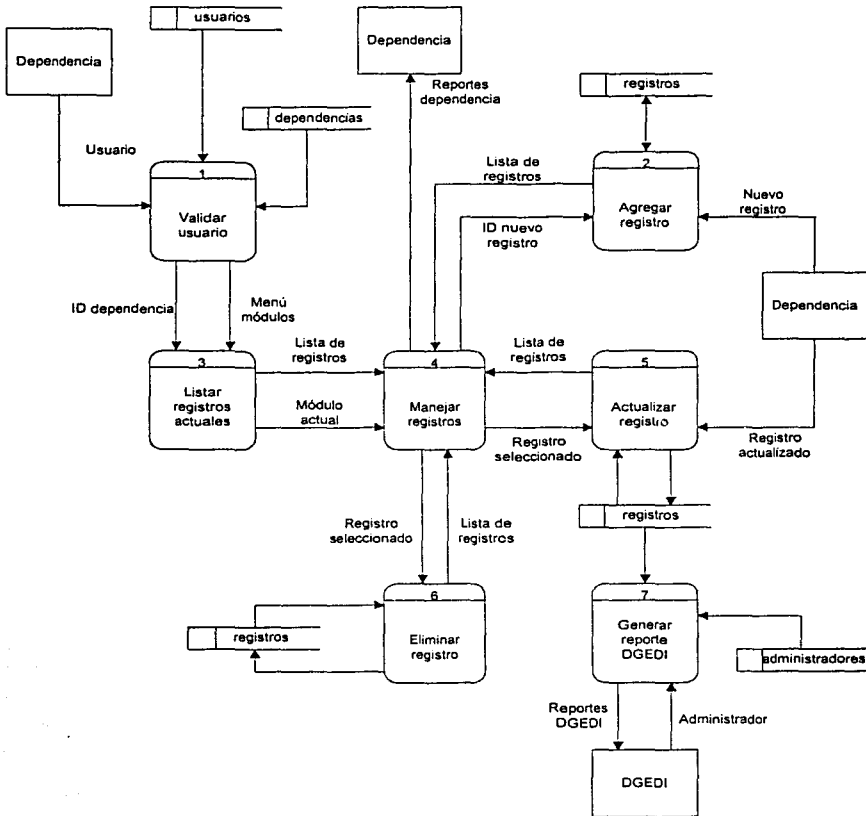


Figura 2.4 Diagrama de nivel 0, Sistema de Acopio Permanente.

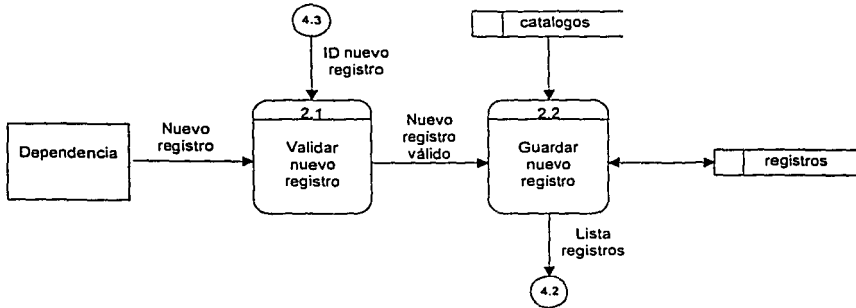


Figura 2.5 Diagrama de nivel 1 proceso 2.

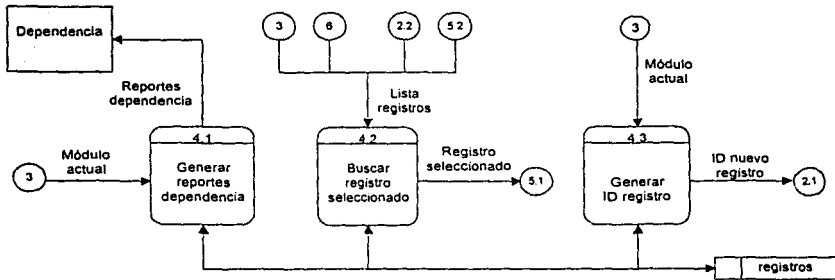


Figura 2.6 Diagrama de nivel 1 proceso 4.

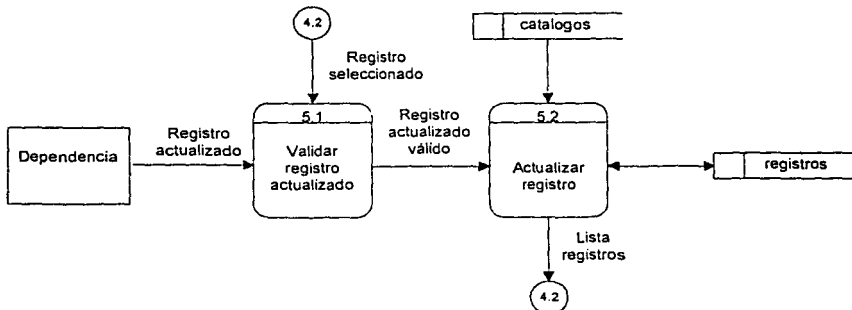


Figura 2.7 Diagrama de nivel 1 proceso 5.

### 2.5.5 Diccionario de datos

Después de que se han creado los diagramas de flujos de datos en sucesivos niveles, éstos resultan de gran utilidad para ayudar a catalogar los procesos, flujos, almacenes, estructuras y elementos de datos en un diccionario de datos.

El diccionario de datos es un trabajo de referencia acerca de ellos que son compilados para servir de guía a través del análisis y diseño. Los diagramas de flujos de datos son un punto de arranque excelente para la recolección de entradas en el diccionario de datos, puesto que éste es creado y examinado describiendo el contenido de los flujos de datos, almacenes y procesos.

Además de proporcionar documentación y eliminar redundancia, el diccionario de datos puede ser usado para:

- ✓ Validar el diagrama de flujo de datos y para confirmar que esté completo y preciso.
- ✓ Proporcionar un punto inicial para el desarrollo de pantallas y reportes.
- ✓ Determinar el contenido de datos almacenados en archivos.
- ✓ Desarrollar la lógica para los diagramas de flujos de datos de procesos.

#### Descripción de los flujos de datos

El flujo de datos es, por lo general, el primer componente a ser definido. La información relacionada a cada flujo de datos del diagrama de nivel 0 se muestra en las siguientes tablas.

<i>ID</i>	Fd1			
<i>Nombre</i>	Usuario			
<i>Descripción</i>	Contiene el nombre de usuario y contraseña de un usuario que desea ingresar al sistema. Esta información es utilizada para validar a dicho usuario.			
<i>Origen del flujo de datos</i>	Entidad externa Dependencia			
<i>Destino del flujo de datos</i>	Proceso 1, Validar usuario			
<i>Estructura de datos para este flujo</i>	Información del usuario			
<i>Tipo de flujo de datos</i>	Archivo	Pantalla	Reporte	✓ Forma Interno

Tabla 2.2 Flujo de datos Usuario.

<b>ID</b>	<b>Fd2</b>			
<b>Nombre</b>	<b>ID dependencia</b>			
<b>Descripción</b>	Contiene el identificador de la dependencia para el usuario que ha sido validado. Cada dependencia tiene un identificador único, que se utiliza con cada registro que se maneja en el sistema.			
<b>Origen del flujo de datos</b>	Proceso 1, Validar usuario			
<b>Destino del flujo de datos</b>	Proceso 3, Listar registros actuales			
<b>Estructura de datos para este flujo</b>	Información del id de la dependencia			
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	Reporte	Forma
				✓ Interno

Tabla 2.3 Flujo de datos ID dependencia.

<b>ID</b>	<b>Fd3</b>			
<b>Nombre</b>	<b>Menú módulos</b>			
<b>Descripción</b>	Contiene una lista de los principales módulos en los cuales se clasifica la información dentro del sistema.			
<b>Origen del flujo de datos</b>	Proceso 1, Validar usuario			
<b>Destino del flujo de datos</b>	Proceso 3, Listar registros actuales			
<b>Estructura de datos para este flujo</b>	Información de la lista de módulos			
<b>Tipo de flujo de datos</b>	Archivo	✓ Pantalla	Reporte	Forma
				Interno

Tabla 2.4 Flujo de datos Menú módulos.

<b>ID</b>	<b>Fd4</b>			
<b>Nombre</b>	<b>Módulo actual</b>			
<b>Descripción</b>	Contiene la información del módulo que ha seleccionado una dependencia, y se utiliza para que el sistema pueda trabajar sobre los registros adecuados.			
<b>Origen del flujo de datos</b>	Proceso 3, Listar registros actuales			
<b>Destino del flujo de datos</b>	Proceso 4, Manejar registros			
<b>Estructura de datos para este flujo</b>	Información del módulo actual			
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	Reporte	Forma
				✓ Interno

Tabla 2.5 Flujo de datos Módulo actual.

<b>ID</b>	Fd5				
<b>Nombre</b>	Lista de registros				
<b>Descripción</b>	Contiene una lista de los registros que actualmente tiene una dependencia para alguno de los módulos.				
<b>Origen del flujo de datos</b>	Proceso 3, Listar registros actuales Proceso 2, Agregar registro Proceso 5, Actualizar registro Proceso 6, Eliminar registro				
<b>Destino del flujo de datos</b>	Proceso 4, Manejar registros				
<b>Estructura de datos para este flujo</b>	Información de la lista de registros				
<b>Tipo de flujo de datos</b>	Archivo	✓ Pantalla	Reporte	Forma	Interno

Tabla 2.6 Flujo de datos Lista de registros.

<b>ID</b>	Fd7				
<b>Nombre</b>	ID nuevo registro				
<b>Descripción</b>	Contiene un identificador para un registro, y se utiliza cuando se va a agregar un nuevo registro en alguno de los módulos.				
<b>Origen del flujo de datos</b>	Proceso 4, Manejar registros				
<b>Destino del flujo de datos</b>	Proceso 2, Agregar registro				
<b>Estructura de datos para este flujo</b>	Información del id nuevo registro				
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	Reporte	Forma	✓ Interno

Tabla 2.7 Flujo de datos ID nuevo registro.

<b>ID</b>	Fd8				
<b>Nombre</b>	Nuevo registro				
<b>Descripción</b>	Contiene los datos para el nuevo registro que se va a agregar en alguno de los módulos.				
<b>Origen del flujo de datos</b>	Entidad externa Dependencia				
<b>Destino del flujo de datos</b>	Proceso 2, Agregar registro				
<b>Estructura de datos para este flujo</b>	Información del nuevo registro				
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	Reporte	✓ Forma	Interno

Tabla 2.8 Flujo de datos Nuevo registro.

<i>ID</i>	Fd9				
<i>Nombre</i>	Registro seleccionado				
<i>Descripción</i>	Contiene los datos de un registro que va a ser eliminado o actualizado.				
<i>Origen del flujo de datos</i>	Proceso 4, Manejar registros				
<i>Destino del flujo de datos</i>	Proceso 5, Actualizar registro Proceso 6, Eliminar registro				
<i>Estructura de datos para este flujo</i>	Información del registro seleccionado				
<i>Tipo de flujo de datos</i>	Archivo	Pantalla	Reporte	✓ Forma	Interno

Tabla 2.9 Flujo de datos Registro seleccionado.

<i>ID</i>	Fd10				
<i>Nombre</i>	Registro actualizado				
<i>Descripción</i>	Contiene los datos para un registro que va a ser actualizado.				
<i>Origen del flujo de datos</i>	Entidad externa Dependencia				
<i>Destino del flujo de datos</i>	Proceso 5, Actualizar registro				
<i>Estructura de datos para este flujo</i>	Información del registro actualizado				
<i>Tipo de flujo de datos</i>	Archivo	Pantalla	Reporte	✓ Forma	Interno

Tabla 2.10 Flujo de datos Registro actualizado.

<i>ID</i>	Fd6				
<i>Nombre</i>	Reportes dependencia				
<i>Descripción</i>	Contiene la información de los registros que tiene una dependencia en alguno de los módulos.				
<i>Origen del flujo de datos</i>	Proceso 4, Manejar registros				
<i>Destino del flujo de datos</i>	Entidad externa Dependencia				
<i>Estructura de datos para este flujo</i>	Información de los reportes de la dependencia				
<i>Tipo de flujo de datos</i>	Archivo	Pantalla	✓ Reporte	Forma	Interno

Tabla 2.11 Flujo de datos Reportes dependencia.



<b>ID</b>	Fd11				
<b>Nombre</b>	Administrador				
<b>Descripción</b>	Contiene el nombre de usuario y contraseña de alguna persona de la DGEDI. Se utilizan para validar a la persona que quiere acceder a los reportes globales.				
<b>Origen del flujo de datos</b>	Entidad externa DGEDI				
<b>Destino del flujo de datos</b>	Proceso 7, Generar reporte DGEDI				
<b>Estructura de datos para este flujo</b>	Información del administrador				
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	Reporte	✓ Forma	Interno

Tabla 2.12 Flujo de datos Administrador.

<b>ID</b>	Fd12				
<b>Nombre</b>	Reportes DGEDI				
<b>Descripción</b>	Contiene la información de los datos que han capturado las dependencias en cada uno de los módulos. Esta información es utilizada por la DGEDI.				
<b>Origen del flujo de datos</b>	Proceso 7, Generar reporte DGEDI				
<b>Destino del flujo de datos</b>	Entidad externa DGEDI				
<b>Estructura de datos para este flujo</b>	Información de reportes DGEDI				
<b>Tipo de flujo de datos</b>	Archivo	Pantalla	✓ Reporte	Forma	Interno

Tabla 2.13 Flujo de datos Reportes DGEDI.

### Descripción de las estructuras de datos

Las estructuras de datos son descritas por lo general usando notación algebraica. Esto permite producir una lista de los elementos que conforman la estructura de datos, junto con la información acerca de esos elementos. La notación algebraica usa los siguientes símbolos:

1. Un símbolo de igual (=) significa "está compuesta de".
2. Un signo de más (+) significa "y".
3. Las llaves {} indican elementos repetidos, también llamados grupos repetidos o tablas.
4. Los corchetes [] representan una situación disyuntiva. Puede estar presente un elemento u otro, pero no ambos. Los elementos listados entre corchetes son mutuamente excluyentes.
5. Los paréntesis () representan un elemento opcional. Los elementos opcionales pueden ser dejados en blanco en las pantallas de captura.

A continuación se describen las estructuras de datos para cada uno de los flujos de datos.

Usuario =	Login + Password
-----------	---------------------

Tabla 2.14 Información del usuario.

ID dependencia =	{ Dígito }
Dígito =	[ 1 - 9 ]

Tabla 2.15 Información del id de la dependencia.

Menú módulos =	[ Proyectos investigación   Productos editoriales   Publicaciones del personal   Personal académico   Actividades académicas   Actividades difusión   Actividades comunicación   Actividades educación continua   Premios y distinciones recibidos   Premios y distinciones otorgados ]
----------------	--

Tabla 2.16 Información de la lista de módulos.

Módulo actual =	[ Proyectos investigación   Productos editoriales   Publicaciones del personal   Planta académica   Actividades académicas   Actividades difusión   Actividades comunicación   Actividades educación continua   Premios y distinciones recibidos   Premios y distinciones otorgados ]
-----------------	--

Tabla 2.17 Información del módulo actual.

Productos editoriales =	[ Boletines   Carteles   Cartográficos   Catálogos   Cuadernos   Folletos   Informes   Libros   Materiales audiovisuales   Materiales de apoyo docente   Memorias   Revista y anales   Software ]
-------------------------	---

Tabla 2.17 (continuación) Información del módulo actual.

Publicaciones del personal =	[ Artículos en memoria   Artículos en periódico   Artículos en revista arbitrada   Artículos en revista no arbitrada   Capítulos en libro   Catálogos   Informes   Introducciones   Libros   Materiales de apoyo docente   Prólogos   Reportes ]
------------------------------	---

Tabla 2.17 (continuación) Información del módulo actual.

Actividades académicas =	[ Coloquios   Concursos   Conferencias   Congresos   Cursos   Encuentros   Foros   Seminarios   Simposio   Talleres ]
--------------------------	--

Tabla 2.17 (continuación) Información del módulo actual.

Actividades difusión =	[ Conciertos   Exposiciones   Ferias   Festivales   Jornadas   Lecturas   Mesas redondas   Muestras   Obras de danza   Obras de teatro   Obras filmica y/o video   Performance   Presentación de publicaciones   Visitas guiadas ]
------------------------	---

Tabla 2.17 (continuación) Información del módulo actual.

Actividades comunicación =	[ Cápsulas   Programas   Promocionales ]
----------------------------	--

Tabla 2.17 (continuación) Información del módulo actual.

Lista de registros =	<ul style="list-style-type: none"> <li>[ Lista proyectos investigación  </li> <li>Lista productos editoriales  </li> <li>Lista publicaciones del personal  </li> <li>Lista personal académico  </li> <li>Lista actividades académicas  </li> <li>Lista actividades difusión  </li> <li>Lista actividades comunicación  </li> <li>Lista actividades educación continua  </li> <li>Lista premios y distinciones recibidos  </li> <li>Lista premios y distinciones otorgados ]</li> </ul>
----------------------	--

Tabla 2.18 Información de la lista de registros.

Lista productos editoriales =	<ul style="list-style-type: none"> <li>[ Lista carteles  </li> <li>Lista cartográficos  </li> <li>Lista catálogos  </li> <li>Lista cuadernos  </li> <li>Lista folletos  </li> <li>Lista informes  </li> <li>Lista libros  </li> <li>Lista materiales audiovisuales  </li> <li>Lista materiales de apoyo docente  </li> <li>Lista memorias  </li> <li>Lista revista y anales  </li> <li>Lista software ]</li> </ul>
-------------------------------	--

Tabla 2.18 (continuación) Información de la lista de registros.

Lista publicaciones del personal =	<ul style="list-style-type: none"> <li>[ Lista artículos en memoria  </li> <li>Lista artículos en periódico  </li> <li>Lista artículos en revista arbitrada  </li> <li>Lista artículos en revista no arbitrada  </li> <li>Lista capítulos en libro  </li> <li>Lista catálogos  </li> <li>Lista informes  </li> <li>Lista introducciones  </li> <li>Lista libros  </li> <li>Lista materiales de apoyo docente  </li> <li>Lista prólogos  </li> <li>Lista reportes ]</li> </ul>
------------------------------------	---

Tabla 2.18 (continuación) Información de la lista de registros.

Lista actividades académicas =	<ul style="list-style-type: none"> <li>[ Lista coloquios  </li> <li>Lista concursos  </li> <li>Lista conferencias  </li> <li>Lista congresos  </li> <li>Lista cursos  </li> <li>Lista encuentros  </li> <li>Lista foros  </li> <li>Lista seminarios  </li> <li>Lista simposio  </li> <li>Lista talleres ]</li> </ul>
--------------------------------	--

Tabla 2.18 (continuación) Información de la lista de registros.

Lista actividades difusión =	[ Lista conciertos   Lista exposiciones   Lista ferias   Lista festivales   Lista jornadas   Lista lecturas   Lista mesas redondas   Lista muestras   Lista obras de danza   Lista obras de teatro   Lista obras filmicas y/o video   Lista performance   Lista presentación de publicaciones   Lista visitas guiadas ]
------------------------------	--

Tabla 2.18 (continuación) Información de la lista de registros.

Lista actividades de comunicación =	[ Lista cápsulas   Lista programas   Lista promocionales ]
-------------------------------------	--

Tabla 2.18 (continuación) Información de la lista de registros.

Lista proyectos investigación =	{ Nombre del proyecto + Id }
Lista boletines = Lista carteles = Lista cartográficos = Lista catálogos = Lista cuadernos = Lista folletos = Lista informes = Lista libros = Lista materiales audiovisuales = Lista materiales apoyo docente = Lista memorias = Lista revista y anales = Lista software =	{ Título de producto editorial + Id del producto editorial }
Lista artículos en memoria = Lista artículos en periódico = Lista artículos en revista arbitrada = Lista artículos en revista no arbitrada = Lista capítulos en libro = Lista catálogos = Lista informes = Lista introducciones = Lista libros = Lista materiales de apoyo docente = Lista prólogos = Lista reportes =	{ Título de la publicación del personal + Id de la publicación del personal }

Lista personal académico =	{ Nombre del académico + Id académico }
Lista coloquios =	{ Nombre de la actividad académica + Id de la actividad académica }
Lista concursos =	
Lista conferencias =	
Lista congresos =	
Lista cursos =	
Lista encuentros =	
Lista seminarios =	
Lista simposio =	
Lista talleres =	
Lista conciertos =	{ Nombre de la actividad de difusión + Id de la actividad de difusión }
Lista exposiciones =	
Lista ferias =	
Lista festivales =	
Lista jornadas =	
Lista lecturas =	
Lista mesas redondas =	
Lista muestras =	
Lista obras de danza =	
Lista obras de teatro =	
Lista obras fílmicas y/o video =	
Lista performance =	
Lista presentación de publicaciones =	
Lista visitas guiadas =	
Lista cápsulas =	{ Nombre de la actividad de comunicación + Id de la actividad de comunicación }
Lista programas =	
Lista promocionales =	
Lista actividades educación continua =	{ Nombre actividad de educación continua + Id actividad de educación continua }
Lista premios y distinciones recibidos =	{ Nombre premio recibido + Id premio }
Lista premios y distinciones otorgados =	{ Nombre premio otorgado + Id premio }

Tabla 2.18 (continuación) Información de la lista de registros.

ID nuevo registro =	{ Dígito }
Dígito =	[ 1 – 9 ]

Tabla 2.19 Información del id nuevo registro.

Nuevo registro =	[ Datos proyecto de investigación
Registro actualizado =	Datos producto editorial
	Datos publicación del personal
	Datos personal académico
	Datos actividad académica
	Datos actividad de difusión
	Datos actividad de comunicación
	Datos actividad de educación continua
	Datos premio y distinción recibido
	Datos premio y distinción otorgado ]
Registro seleccionado =	[ Datos proyecto de investigación +
	identificador del registro
	Datos producto editorial +
	identificador del registro
	Datos publicación del personal +
	identificador del registro
	Datos personal académico +
	identificador del registro
	Datos actividad académica +
	identificador del registro
	Datos actividad de difusión +
	identificador del registro
	Datos actividad de comunicación +
	identificador del registro
	Datos actividad de educación continua +
	identificador del registro
	Datos premio y distinción recibido +
	identificador del registro
	Datos premio y distinción otorgado +
	identificador del registro ]

Tabla 2.20 Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Datos producto editorial =	{ Datos boletín   Datos cartel
	Datos cartográfico   Datos catálogo
	Datos cuaderno   Datos folleto
	Datos informe   Datos libro
	Datos material audiovisual
	Datos material de apoyo docente
	Datos memoria   Datos revista y anales
	Datos software }

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Datos publicación del personal =	[ Datos artículo en memoria   Datos artículo en periódico   Datos artículo en revista arbitrada   Datos artículo en revista no arbitrada   Datos capítulo en libro   Datos catálogo   Datos informe   Datos introducción   Datos libro   Datos material de apoyo docente   Datos prólogo   Datos reporte ]
----------------------------------	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Datos actividad académica =	[ Datos coloquio   Datos concurso   Datos conferencia   Datos congreso   Datos curso   Datos encuentro   Datos foro   Datos seminario   Datos simposio   Datos taller ]
-----------------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Datos actividad de difusión =	[ Datos concierto   Datos exposición   Datos feria   Datos festival   Datos jornada   Datos lectura   Datos mesa redonda   Datos muestra   Datos obra de danza   Datos obra de teatro   Datos obra filmica y/o video   Datos performance   Datos presentación de publicaciones   Datos visita guiada ]
-------------------------------	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Datos actividad de comunicación =	[ Datos cápsula   Datos programa   Datos promocional ]
-----------------------------------	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.



<i>Datos proyecto de investigación =</i>	Nombre del proyecto + Línea de investigación + Responsable + RFC del responsable + Número de trabajador del responsable + { Colaborador } + Campo de la ciencia + { Fuente de financiamiento } Tipo de actividad + Status del proyecto + Proyecto terminado + Alumnos de bachillerato involucrados + Alumnos de licenciatura involucrados + Alumnos de postgrado involucrados + Comentarios
Colaborador =	Título académico + Nombre + Institución de origen + (País de origen )
Campo de la ciencia =	[ Ciencias naturales   Ingeniería y tecnología   Ciencias agrícolas   Ciencias médicas   Ciencias sociales   Humanidades ]
Fuente de financiamiento =	[ Presupuestal directo (UNAM)   Programas institucionales   Intercambio académico   Fondos propios   Gobierno   Privado   Educación superior   Instituciones privadas no lucrativas   Exterior   Sin financiamiento   No definida ]
Tipo de actividad =	[ Investigación básica   Investigación aplicada   Investigación de desarrollo experimental   No definida ]
Status del proyecto =	[ Nuevo   De continuación ]
Proyecto terminado =	[ Si   No ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos boletín =</i>	Título + Editorial + Ciudad + Fecha de tiraje + Periodicidad + Total de páginas + Tiraje + Volumen + Número + Autor + {Co-autor}
Periodicidad =	[ Anual   Bimestral   Cuatrimestral   Irregular   Mensual   Quincenal   Semanal   Semestral   Trimestral   No definida   Otra ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos cartel =</i>	Título + Año + Tiraje + Evento + Autor + { Co-autor }
-----------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos cartográfico =</i>	Título + Año + Escala + Autor + { Co-autor }
-----------------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos catálogo =</i>	Título + ISSN + Editorial + Ciudad + Año + Total de páginas + Tiraje + Volumen + Número + Autor + { Co-autor }
-------------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos cuaderno =*

Título +  
 Colección +  
 Año +  
 Tiraje +  
 Edición +  
 Total de páginas +  
 Autor +  
 { Co-autor }

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos folleto =*

Título +  
 Editorial +  
 Ciudad +  
 Año +  
 Total de páginas +  
 Tiraje +  
 Número +  
 Autor +  
 { Co-autor }

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos informe =*

Título +  
 Año +  
 Total de páginas +  
 Autor +  
 { Co-autor }

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos libro =*

Título +  
 ISBN +  
 Editorial +  
 Colección +  
 Ciudad +  
 Año +  
 Total de páginas +  
 Tiraje +  
 Edición +  
 Autor +  
 { Co-autor }

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos material audiovisual =</i>	Título + Año + Tipo material + Autor + { Co-autor }
Tipo material =	[ CD   CD ROM   Videocasete   Audio casete   Otro ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos material de apoyo docente =</i>	Título + Editorial + Colección + Ciudad + Año + ISBN + Tipo de material + Receptor + Total de páginas Tiraje + Edición + Autor + { Co-autor }
Receptor =	[ Bachillerato   Licenciatura   Posgrado   Técnico   No definido ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos memoria =</i>	Título + Editorial + Ciudad + Año + Total de páginas + Tiraje + Volumen + Número + Autor + { Co-autor }
------------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos revista y anales =</i>	Título + ISSN + Editorial + Ciudad + Año + Periodicidad + Arbitrada + Total de páginas + Tiraje + Volumen + Número + Autor + { Colaborador }
Arbitrada =	[ Si   No ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos software =</i>	Título + Año + Versión + Lenguaje + Tamaño + Autor + { Co-autor }
-------------------------	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos artículo en memoria =</i> <i>Datos artículo en revista arbitrada =</i> <i>Datos artículo en revista no arbitrada =</i>	Título + Nombre + Año + Página inicial + Página final + Volumen + Número + Autor + { Co-autor }
---	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos artículo en periódico =</i>	Título + Nombre del periódico + Fecha + Página inicial + Página final + Época + Número + Autor + { Co-Autor }
--------------------------------------	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos capítulo en libro =</i> <i>Datos introducción =</i> <i>Datos prólogo =</i>	Título + Nombre del libro + ISBN + Editorial + Colección + ( Editor )+ Ciudad + Año + Página inicial + Página final + Tiraje + Edición + Autor + { Co-autor }
---	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos catálogo =</i>	Título + ISSN + Editorial + Ciudad + Año + Total de páginas + Tiraje + Volumen + Número + Autor + { Co-autor }
-------------------------	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos libro =</i> <i>Datos material de apoyo docente =</i>	Título + ISBN + Editorial + Colección + Ciudad + Total de páginas + Año + Tiraje + Edición + Autor + { Co-autor }
--	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos reporte =</i> <i>Datos informe =</i>	Título + Año + Total de páginas + Autor + { Co-autor }
--	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos personal académico =</i>	Nombre del académico + RFC + Nivel + { Categoría asignada } + Realizando estudios + Lugar estudios + { Tipo de beca } + Tesis licenciatura dirigidas + Tesis maestría dirigidas + Tesis doctorado dirigidas + Alumnos de servicio social
Nivel =	[ Licenciatura   Especialización   Maestría   Doctorado   No especificado   Menos de Licenciatura ]
Realizando estudios =	[ Licenciatura   Especialización   Maestría   Doctorado   Post-doctorado   No está realizando estudios ]
Categoría asignada =	[ Profesor de Asignatura A   Profesor de Asignatura B   Ayudante de Profesor de Asignatura A   Ayudante de Profesor de Asignatura B   Profesor de Enseñanza Media Superior A   Profesor de Enseñanza Media Superior B   Profesor de Carrera Asociado A Medio Tiempo   Profesor de Carrera Asociado B Medio Tiempo   Profesor de Carrera Asociado C Medio Tiempo   Profesor de Carrera Asociado A Tiempo Completo   Profesor de Carrera Asociado B Tiempo Completo   Profesor de Carrera Asociado C Tiempo Completo   Profesor de Carrera Titular A Medio Tiempo   Profesor de Carrera Titular B Medio Tiempo   Profesor de Carrera Titular C Medio Tiempo   Profesor de Carrera Titular A Tiempo Completo   Profesor de Carrera Titular B Tiempo Completo   Profesor de Carrera Titular C Tiempo Completo   Ayudante de Profesor A Medio Tiempo   Ayudante de Profesor B Medio Tiempo   Ayudante de Profesor C Medio Tiempo   Ayudante de Profesor A Tiempo Completo   Ayudante de Profesor B Tiempo Completo   Ayudante de Profesor C Tiempo Completo   Técnico Académico Asociado B Tiempo Completo Investigación   Técnico Académico Asociado C Medio Tiempo Investigación   Técnico Académico Asociado C Tiempo Completo Investigación   Técnico Académico Auxiliar A Medio Tiempo Investigación   Técnico Académico Auxiliar A Tiempo Completo Investigación   Técnico Académico Auxiliar B Medio Tiempo Investigación   Técnico Académico Auxiliar B Tiempo Completo Investigación   Técnico Académico Auxiliar C Medio Tiempo Investigación   Técnico Académico Auxiliar C Tiempo Completo Investigación   Técnico Académico Titular A Medio Tiempo Investigación   Técnico Académico Titular A Tiempo Completo Investigación   Técnico Académico Titular B Medio Tiempo Investigación   Técnico Académico Titular B Tiempo Completo Investigación   Técnico Académico Titular C Medio Tiempo Investigación   Técnico Académico Titular C Tiempo Completo Investigación ]

	Investigador de Carrera Titular A Medio Tiempo   Técnico Académico Asociado A Medio Tiempo Docencia   Técnico Académico Asociado A Tiempo Completo Docencia   Investigador de Carrera Asociado A Medio Tiempo   Investigador de Carrera Asociado B Medio Tiempo   Investigador de Carrera Asociado C Medio Tiempo   Investigador de Carrera Asociado A Tiempo Completo   Investigador de Carrera Asociado B Tiempo Completo   Investigador de Carrera Asociado C Tiempo Completo   Técnico Académico Asociado B Medio Tiempo Docencia   Investigador de Carrera Titular B Medio Tiempo   Investigador de Carrera Titular C Medio Tiempo   Investigador de Carrera Titular A Tiempo Completo   Investigador de Carrera Titular B Tiempo Completo   Investigador de Carrera Titular C Tiempo Completo   Ayudante de Investigador A Medio Tiempo   Ayudante de Investigador B Medio Tiempo   Ayudante de Investigador C Medio Tiempo   Ayudante de Investigador A Tiempo Completo   Ayudante de Investigador B Tiempo Completo   Ayudante de Investigador C Tiempo Completo   Técnico Académico Asociado B Tiempo Completo Docencia   Técnico Académico Asociado C Medio Tiempo Docencia   Técnico Académico Asociado C Tiempo Completo Docencia   Técnico Académico Auxiliar A Medio Tiempo Docencia   Técnico Académico Auxiliar A Tiempo Completo Docencia   Técnico Académico Auxiliar B Medio Tiempo Docencia   Técnico Académico Auxiliar B Tiempo Completo Docencia   Técnico Académico Auxiliar C Medio Tiempo Docencia   Técnico Académico Auxiliar C Tiempo Completo Docencia   Técnico Académico Titular A Medio Tiempo Docencia   Técnico Académico Titular A Tiempo Completo Docencia   Técnico Académico Titular B Medio Tiempo Docencia   Técnico Académico Titular B Tiempo Completo Docencia   Técnico Académico Titular C Medio Tiempo Docencia   Técnico Académico Titular C Tiempo Completo Docencia   Técnico Académico Asociado A Medio Tiempo Investigación   Técnico Académico Asociado A Tiempo Completo Investigación   Técnico Académico Asociado B Medio Tiempo Investigación   Profesor o Investigador Emérito   Profesor e Investigador visitante)
Lugar de estudios =	[ Nacional   Extranjero   No especificado ]
Tipo de beca =	[ Presupuestal directo (UNAM)   Programas institucionales   Intercambio académico   Fondos propios   Gobierno   Privado   Educación superior   Instituciones privadas no lucrativas   Exterior   Sin financiamiento   No definida ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.



*Datos coloquio* =  
*Datos concurso* =  
*Datos conferencia* =  
*Datos congreso* =  
*Datos encuentro* =  
*Datos foro* =  
*Datos seminario* =  
*Datos simposio* =

Nombre +  
 Sede +  
 Fecha de inicio +  
 Fecha de término +  
 Número de asistentes

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos curso* =  
*Datos taller* =

Nombre +  
 Ponente +  
 Sede +  
 Fecha de inicio +  
 Fecha de término +  
 Número de sesiones +  
 Número de asistentes

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos concierto* =  
*Datos muestra* =  
*Datos obra de danza* =  
*Datos obra de teatro* =  
*Datos obra filmica y/o video* =

Nombre +  
 Sede +  
 Recinto +  
 Fecha de inicio +  
 Fecha de término +  
 Número de asistentes +  
 Número de funciones

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos exposición* =  
*Datos feria* =  
*Datos festival* =  
*Datos jornada* =  
*Datos mesa redonda* =  
*Datos performance* =  
*Datos visita guiada* =

Nombre +  
 Sede +  
 Recinto +  
 Fecha de inicio +  
 Fecha de término +  
 Número de asistentes

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

*Datos lectura* =

Título +  
 Sede +  
 Recinto +  
 Tipo lectura +  
 Fecha +  
 Número de asistentes

Tipo lectura =

[ Cuento |  
 Poesía |  
 Dramatización ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos presentación de publicaciones</i> =	Nombre + Sede + Recinto + Tipo de presentación + Fecha + Número de asistentes
Tipo de presentación =	[ Cdrom   Libro   Revista   Video   Catálogo   Escritores   Otro ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos cápsula</i> =	Nombre +
<i>Datos programa</i> =	Medio +
<i>Datos promocional</i> =	Fecha de inicio + Fecha de término + Duración + Emisiones + Coproducida
Medio =	[ Radio   Televisión ]
Coproducida =	[ Si   No ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos actividad de educación continua</i> =	Nombre + Modalidad + Número de ponentes + Número de módulos + Número de asistentes + Número de constancias + Número de becados + Sede + Duración + Fecha de inicio + Fecha de término + Tipo de actividad + Diplomado + Comentarios
Modalidad =	[ Presencial   Teleconferencia   Videoconferencia   Multimedia   Internet   Otro ]
Tipo de actividad =	[ Curso   Taller   Seminario   Otro ]
Diplomado =	[ Si   No ]

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos premio y distinción recibido =</i>	Nombre del premio o distinción + Receptor + RFC + Fecha + Institución que otorga
---	--

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

<i>Datos premio y distinción otorgado =</i>	Nombre del premio o distinción + Receptor + RFC + Fecha + Institución beneficiada
---	---

Tabla 2.20 (continuación) Información del nuevo registro, información del registro seleccionado e información del registro actualizado.

Reportes dependencia =	[ Datos proyecto de investigación   Datos producto editorial   Datos publicación del personal   Datos personal académico   Datos actividad académica   Datos actividad de difusión   Datos actividad de comunicación   Datos actividad de educación continua   Datos premio y distinción recibido   Datos premio y distinción otorgado ]
------------------------	---

Tabla 2.21 Información de los reportes de la dependencia.

Administrador =	Login + Password
-----------------	---------------------

Tabla 2.22 Información del administrador.

Reportes DGEDI =	Nombre de la dependencia + Número de registros
Número de registros	Número proyecto de investigación   Número producto editoriales   Número publicaciones del personal   Número personal académico   Número actividades académicas   Número actividades difusión   Número actividades comunicación   Número actividades educación continua   Número premios y distinciones recibidos   Número premios y distinciones otorgados ]

Tabla 2.23 Información de reportes DGEDI.

Número productos editoriales =	[ Número software   Número carteles   Número cartográficos   Número catálogos   Número cuadernos   Número folletos   Número informes   Número libros   Número materiales audiovisuales   Número materiales de apoyo docente   Número memorias   Número revista y anales   Número software ]
--------------------------------	---

Tabla 2.23 (continuación) Información de reportes DGEDI.

Número publicaciones del personal =	[ Número artículos en memoria   Número artículos en periódico   Número artículos en revista arbitrada   Número artículos en revista no arbitrada   Número capítulos en libro   Número catálogos   Número informes   Número introducciones   Número libros   Número materiales de apoyo docente   Número prólogos   Número reportes ]
-------------------------------------	---

Tabla 2.23 (continuación) Información de reportes DGEDI.

Número actividades académicas =	[ Número coloquios   Número concursos   Número conferencias   Número congresos   Número cursos   Número encuentros   Número foros   Número seminarios   Número simposio   Número talleres ]
---------------------------------	--

Tabla 2.23 (continuación) Información de reportes DGEDI.

Número actividades difusión =	[ Número conciertos   Número exposiciones   Número ferias   Número festivales   Número jornadas   Número lecturas   Número mesas redondas   Número muestras   Número obras de danza   Número obras de teatro   Número obras filmicas y/o video   Número performance   Número presentación de publicaciones   Número visitas guiadas ]
-------------------------------	--

Tabla 2.23 (continuación) Información de reportes DGEDI.

Número actividades de comunicación =	[ Número cápsulas   Número programas   Número promocionales ]
--------------------------------------	---

Tabla 2.23 (continuación) Información de reportes DGEDI.

## CAPÍTULO 3

# DISEÑO DEL SISTEMA DE ACOPIO PERMANENTE

### 3.1 Especificación de procesos y procedimientos

Las especificaciones de proceso, a veces llamadas miniespecificaciones, debido a que son una parte pequeña del total de especificaciones del proyecto, son creadas para procesos primitivos en los diagramas de flujo de datos, así como para algunos procesos de más alto nivel que explotan hacia un diagrama hijo.

Los tres objetivos principales de las especificaciones del proceso son:

1. Reducir la ambigüedad del proceso. Esto lleva a aprender los detalles acerca de la manera en que trabajan los procesos.
2. Obtener una descripción precisa de lo que se logra con un proceso, que está, por lo general, incluido en un paquete de especificaciones para el programador.
3. Validar el diseño del sistema. Esto incluye el asegurarse que un proceso tenga todos los flujos de datos necesarios para producir la salida. Además, toda la entrada y salida debe estar representada en el diagrama de flujo de datos.

Las especificaciones de los principales procesos, los del diagrama de nivel 1, se describen en las siguientes tablas.

<i>ID</i>	1	
<i>Nombre</i>	Validar usuario	
<i>Descripción</i>	Se validan los datos del Usuario para determinar si tiene acceso al sistema.	
<i>Flujo de datos de entrada</i>	Usuario de entidad externa Dependencia	
<i>Flujo de datos de salida</i>	<u>ID dependencia</u> al proceso 3 <u>Menú módulos</u> al proceso 3	
<i>Tipo de proceso</i>	> En línea	Por lote                      Manual
<i>Lógica del proceso</i>	<p>Cada Dependencia asignará a una persona, identificada en el sistema como Usuario, que sea la responsable de manejar la información de dicha dependencia, siendo esta persona la única que tendrá acceso al sistema durante el proceso de acopio de información.</p> <p>Si los datos del Usuario son válidos, se le permite el acceso al sistema, y se le muestra a continuación el Menú módulos. Una vez validado el usuario, se obtiene también el valor para ID dependencia, y saber a qué dependencia pertenece, y cuáles son los registros que puede modificar.</p> <p>En caso de que los datos del Usuario no sean válidos, se le deniega el acceso, y se le manda un mensaje de error.</p>	

Tabla 3.1 Proceso 1, Validar usuarios.





**Lógica del proceso**

Cuando el usuario tiene presente la Lista de registros en el Módulo actual, éste tiene la opción de eliminar o actualizar alguno de esos registros, de agregar un nuevo registro, o de generar un reporte para obtener la información de los registros del Módulo actual.

Para eliminar o actualizar un registro, el usuario lo debe de seleccionar de la Lista de registros, y a continuación se le muestra la información completa del Registro seleccionado.

Si decide agregar un nuevo registro, este proceso genera un ID nuevo registro y se lo pasa como dato al proceso Agregar registro.

Si la operación escogida es generar reportes, este proceso genera los Reportes dependencia, que contienen la información completa de los registros que tiene actualmente una dependencia en el Módulo actual.

Tabla 3.3 Proceso 4, Manejar registros.

<b>ID</b>	2	
<b>Nombre</b>	Agregar registro	
<b>Descripción</b>	Agrega un nuevo registro en el Módulo actual.	
<b>Flujo de datos de entrada</b>	<u>ID nuevo registro</u> del proceso 4	
	<u>Nuevo registro</u> de la entidad externa Dependencia	
<b>Flujo de datos de salida</b>	<u>Lista de registros</u> al proceso 4	
<b>Tipo de proceso</b>		
> En línea	Por lote	Manual
<b>Lógica del proceso</b>		
Este proceso recibe el valor de ID nuevo registro por el proceso 4, y los datos del Nuevo registro por parte de la Dependencia, con estos datos forma el registro completo para agregarlo en la base de datos, en el Módulo actual.		
Si el registro pudo ser agregado de manera satisfactoria, este proceso muestra entonces la nueva Lista de registros, en donde debe de aparecer ya el registro que acaba de agregarse.		
En caso contrario, se le muestra al usuario un mensaje de error, y a continuación la misma Lista de registros sin cambios.		

Tabla 3.4 Proceso 2, Agregar registro.

<b>ID</b>	5	
<b>Nombre</b>	Actualizar registro	
<b>Descripción</b>	Actualiza los datos de un registro del Módulo actual.	
<b>Flujo de datos de entrada</b>	<u>Registro seleccionado</u> del proceso 4 <u>Registro actualizado</u> de la entidad externa Dependencia.	
<b>Flujo de datos de salida</b>	<u>Lista de registros</u> al proceso 4	
<b>Tipo de proceso</b>		
	> En línea	Por lote Manual
<b>Lógica del proceso</b>	Este proceso recibe los datos del Registro seleccionado por el usuario de la Lista de registros, para mostrar los datos completos que tiene actualmente el registro, y para permitir modificarlos con los datos del Registro actualizado, que son proporcionados por la Dependencia.  Cuando los datos han sido modificados, se pueden entonces actualizar en la base de datos.  Si los datos del registro pudieron ser actualizados de manera satisfactoria, se le pasa al proceso 4 la Lista de registros ya actualizada, para que le sean mostrados al usuario.  En caso contrario, se le muestra al usuario un mensaje de error, y a continuación se le muestra al usuario la Lista de registros sin cambios.	

Tabla 3.5 Proceso 5, Actualizar registro.

<b>ID</b>	6	
<b>Nombre</b>	Eliminar registro	
<b>Descripción</b>	Elimina un registro del Módulo actual	
<b>Flujo de datos de entrada</b>	<u>Registro seleccionado</u> del proceso 4	
<b>Flujo de datos de salida</b>	<u>Lista de registros</u> al proceso 4	
<b>Tipo de proceso</b>		
	> En línea	Por lote Manual
<b>Lógica del proceso</b>	Este proceso recibe los datos del Registro seleccionado por el usuario de la Lista de registros, para mostrar los datos completos que tiene actualmente el registro, y para permitir eliminar dicho registro.  Si el Registro seleccionado pudo ser eliminado de manera satisfactoria, se le pasa al proceso 4 la Lista de registros ya actualizada, para que le sean mostrados al usuario.  En caso contrario, se le muestra al usuario un mensaje de error, y a continuación se le muestra al usuario la Lista de registros sin cambios.	

Tabla 3.6 Proceso 6, Eliminar registro.

<b>ID</b>	<b>7</b>		
<b>Nombre</b>	Generar reporte DGEDI		
<b>Descripción</b>	Genera reportes globales para la DGEDI		
<b>Flujo de datos de entrada</b>	<u>Administrador</u> de la entidad externa DGEDI		
<b>Flujo de datos de salida</b>	<u>Reportes DGEDI</u> a la entidad externa DGEDI		
<b>Tipo de proceso</b>			
> En línea	Por lote		Manual
<b>Lógica del proceso</b>			
La DGEDI es la única entidad que puede acceder a la información que ha sido copiada por la distintas entidades académicas, en los distintos módulos.			
Para que a la DGEDI se le permita acceder a los reportes de estos datos, es necesario que se autentique al Administrador, que es una persona de la DGEDI que tiene permitido consultar los reportes.			
Si el Administrador es un autenticado de manera satisfactoria, se le muestran a continuación los Reportes DGEDI, que contienen los números de los registros capturados en cada uno de los módulos del Menú módulos.			
En caso contrario, se manda un mensaje de error al Administrador para que verifique su nombre de usuario, y su contraseña.			

Tabla 3.7 Proceso 7, Generar reporte DGEDI.

### 3.2 Modelo de la base de datos

El almacenamiento de datos siempre será considerado una parte medular en cualquier sistema de información, por pequeño que éste sea. Los objetivos generales que se tienen que cumplir para un buen almacenamiento de los datos son:

- ✓ Los datos tienen que estar disponibles cuando el usuario quiere o tiene que usarlos.
- ✓ Los datos tienen que ser precisos y consistentes, es decir, deben poseer integridad.
- ✓ El almacenamiento de los datos debe de ser eficiente, tanto para la actualización, como para la recuperación.
- ✓ La información obtenida de los datos almacenados debe estar en un formato útil para la administración, planeación, control o toma de decisiones.

Hay dos enfoques para el almacenamiento de datos en un sistema basado en computadora. El primero consiste en guardar los datos en archivos individuales, cada uno de ellos único para un aplicación particular. El segundo enfoque consiste en la construcción de una base de datos. Una base de datos es una colección integrada de datos de computadora, organizados y almacenados de manera tal que se facilita su recuperación.

Las bases de datos no son simplemente un conjunto de archivos. En vez de ello, una base de datos es una fuente central de datos que está pensada para que sea compartida por muchos usuarios con una diversidad de aplicaciones. La parte medular de la base de datos es el DBMS (sistema de administración de base de datos) que permite la creación, modificación y actualización de la base de datos, la recuperación de datos y la generación de reportes.

Los objetivos de la efectividad de la base de datos incluyen:

- ✓ Asegurarse de que la base de datos pueda ser compartida entre los diversos usuarios.
- ✓ Mantener datos que sean precisos y consistentes.
- ✓ Asegurarse de que todos los datos requeridos para las aplicaciones actuales y futuras estén fácilmente disponibles.
- ✓ Permitir que la base de datos evolucione y que las necesidades de los usuarios crezcan.
- ✓ Permitir que los usuarios construyan su vista personal de los datos sin preocuparse de la forma en que estén físicamente guardados los datos.

Adicional a esto, dos objetivos primordiales del concepto de base de datos son minimizar la redundancia de los datos y lograr la independencia de los datos. La redundancia de datos es la duplicidad de datos; es decir, el almacenamiento de los mismos datos en varios archivos. La independencia de los datos es la capacidad de modificar la estructura de los datos sin tener que modificar los programas que procesan los datos. La independencia de los datos se logra colocando las especificaciones de los datos en tablas y diccionarios físicamente aparte de los programas. Los programas consultan las tablas para poder acceder a los datos. Las modificaciones de la estructura de los datos sólo se efectúan una vez: en las tablas.

### 3.3 Diagramas entidad-relación

Antes de considerar el diseño de la base de datos, es importante comprender cómo son representados los datos, y para ello se incluyen las siguientes definiciones.

**Entidades.** Es cualquier objeto o evento acerca del cual se tienen que recolectar datos.

**Relaciones.** Las relaciones son asociaciones entre entidades.

**Atributos.** Un atributo es alguna característica de una entidad. Puede haber muchos atributos para una entidad. La palabra *concepto de datos* puede ser usada de manera indistinta en lugar de atributo. Un atributo puede tener valores que sean de longitud fija o variable, y ser alfabéticos, numéricos o alfanuméricos.

**Registros.** Un registro es un conjunto de conceptos de datos que tienen algo en común con la entidad descrita.

**Llaves.** Una llave es uno de los conceptos de datos de un registro que se usa para identificar un registro. Cuando una llave identifica en forma única a un registro es llamada la llave primaria. Una llave se denomina secundaria si no puede identificar en forma única a un registro. Las llaves secundarias pueden usarse para seleccionar un grupo de registros que pertenecen a un mismo conjunto.

**Metadatos.** Los metadatos son datos acerca de los datos de una base de datos. Los metadatos describen el nombre dado y la longitud asignada a cada concepto de datos. Los metadatos también describen la longitud y composición de cada uno de los registros.

Todos estos conceptos corresponden a un modelo de datos que se denomina modelo entidad-relación, y se emplea para interpretar, especificar y documentar los requerimientos para sistemas de bases de datos, ya que proporciona estructuras que muestran el diseño general de los requerimientos de datos de los usuarios.

Dado que el diagrama completo se ha optado por presentar los diagramas de la siguiente manera.

En primera instancia, se muestra únicamente un diagrama entidad-relación en el que se observan las principales entidades, y sus relaciones. En los posteriores diagramas se analiza de manera más detallada cada una de las entidades, que corresponden a cada uno de los módulos en los cuales se organiza la información dentro del sistema.

Como se puede observar en el siguiente diagrama, todas las entidades que representan a cada uno de los módulos, tiene el mismo tipo de relación con la entidad *DEPENDENCIA*, que representa a cada una de las entidades académicas de la Universidad responsables de llevar a cabo el proceso de acopio de información.

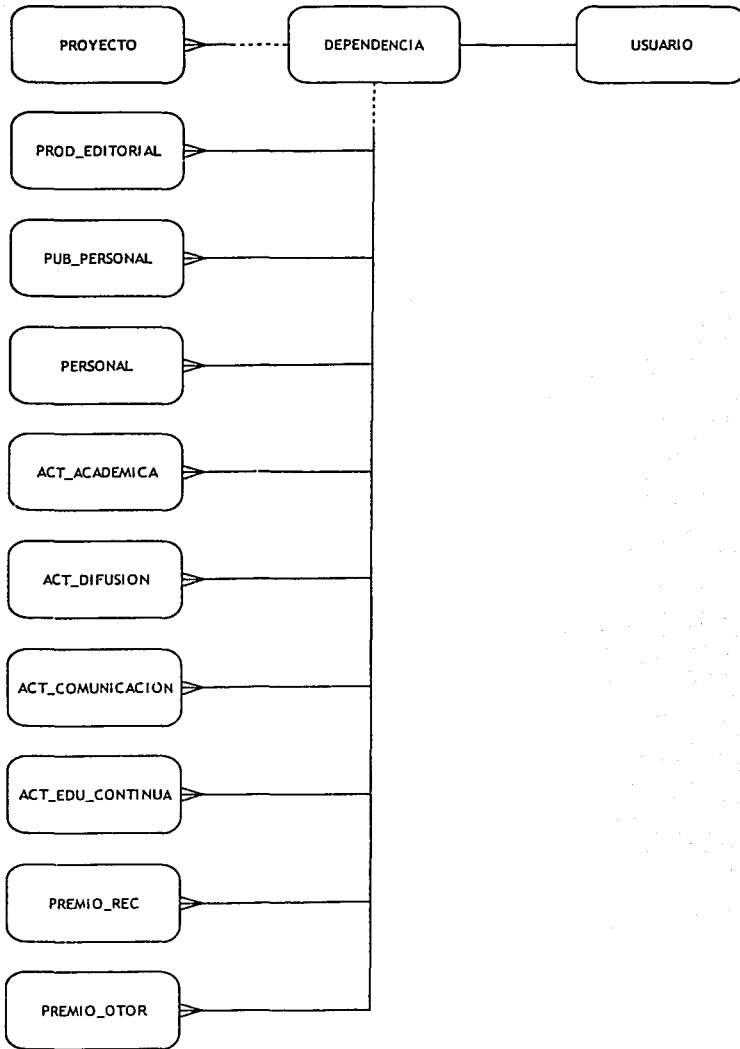


Figura 3.1 Diagrama E-R Sistema de Acopio Permanente.

Por simplicidad, en el diagrama anterior no se han señalado los atributos de cada una de las entidades, esto se hará en cada uno de los siguientes diagramas.

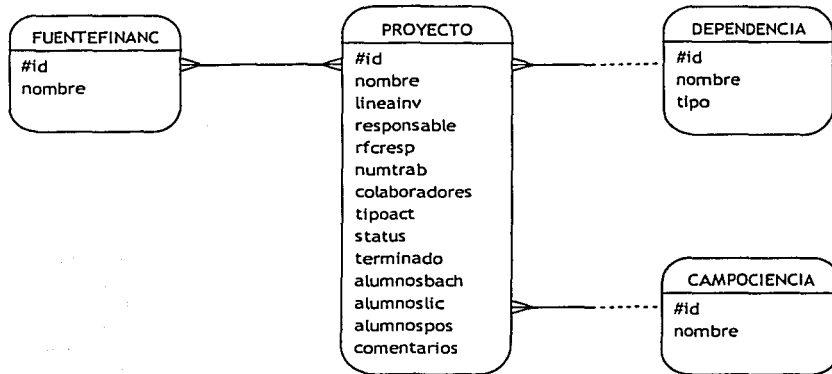


Figura 3.2 Diagrama E-R Proyectos de investigación

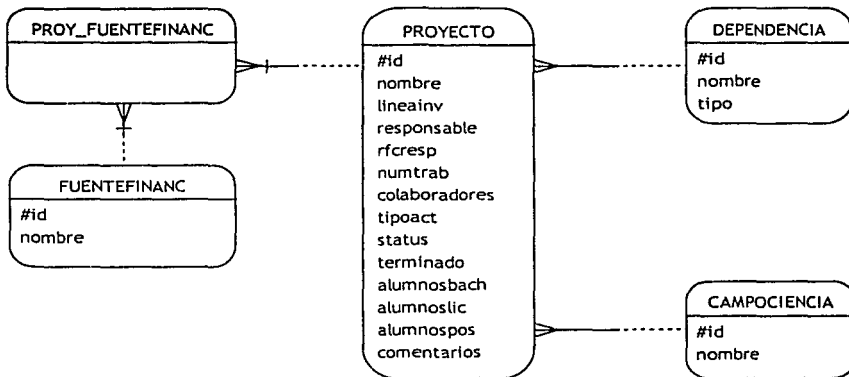


Figura 3.3 Diagrama E-R Proyectos de investigación sin relaciones M:M.

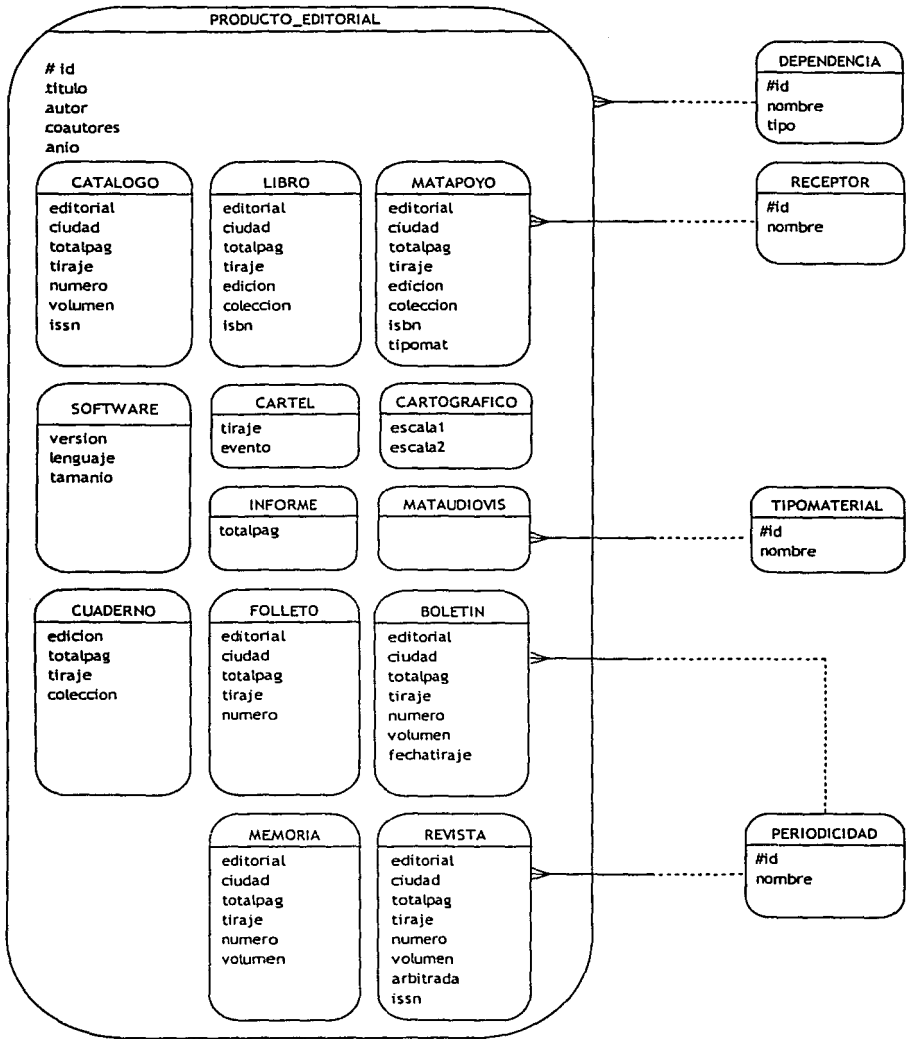


Figura 3.4 Diagrama E-R Productos editoriales.



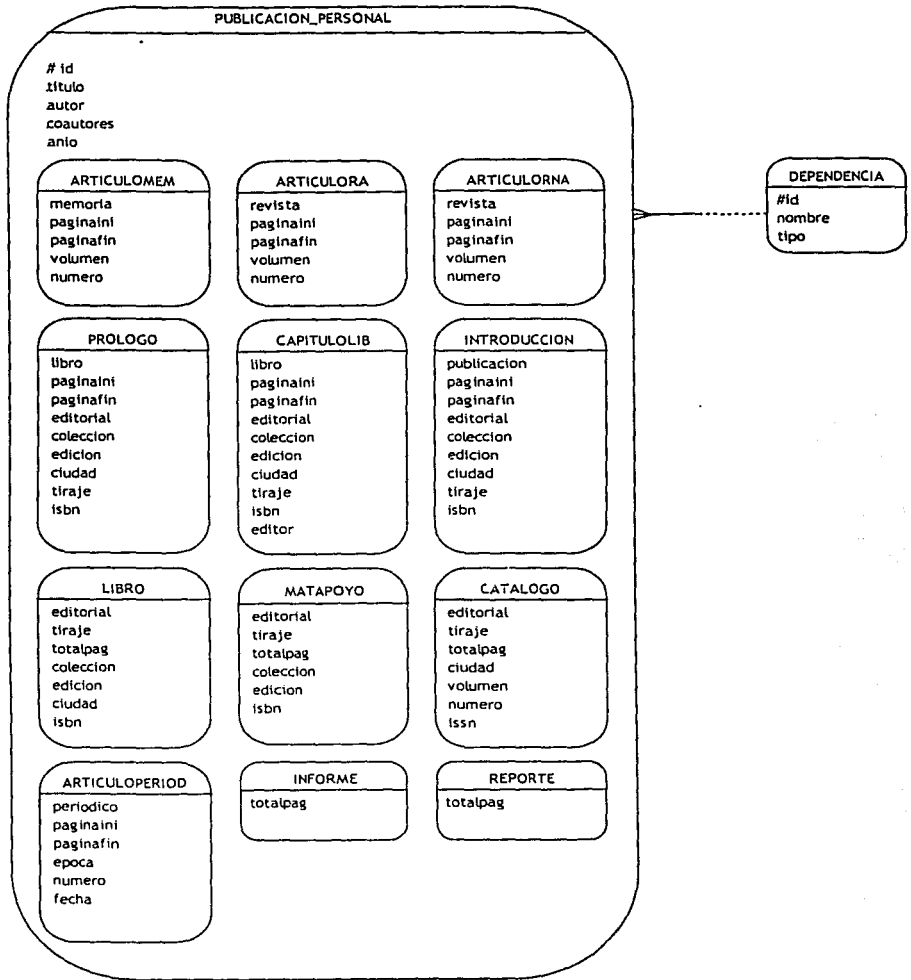


Figura 3.5 Diagrama E-R Publicaciones del personal.

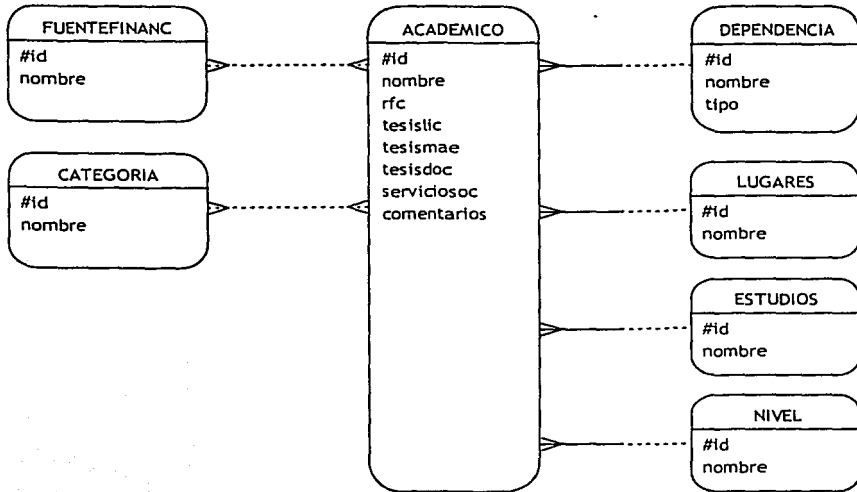


Figura 3.6 Diagrama E-R Personal académico.

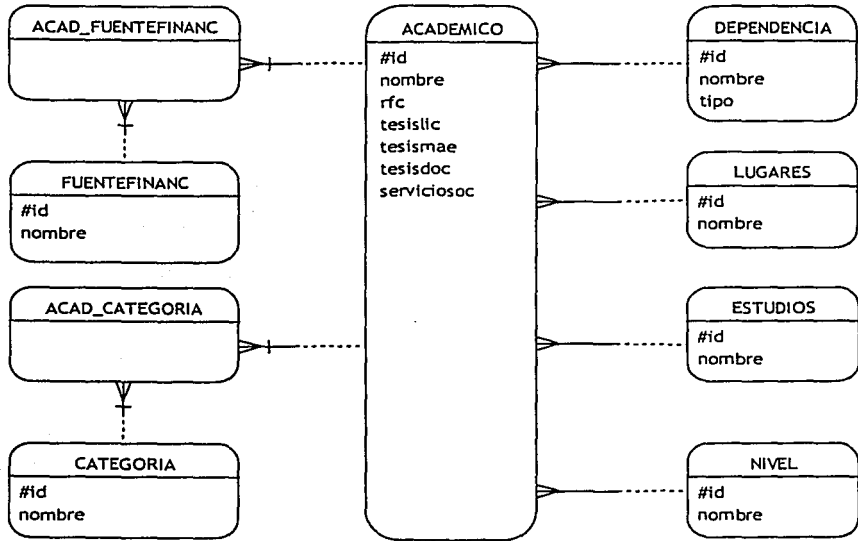


Figura 3.7 Diagrama E-R Personal académico sin relaciones M:M.

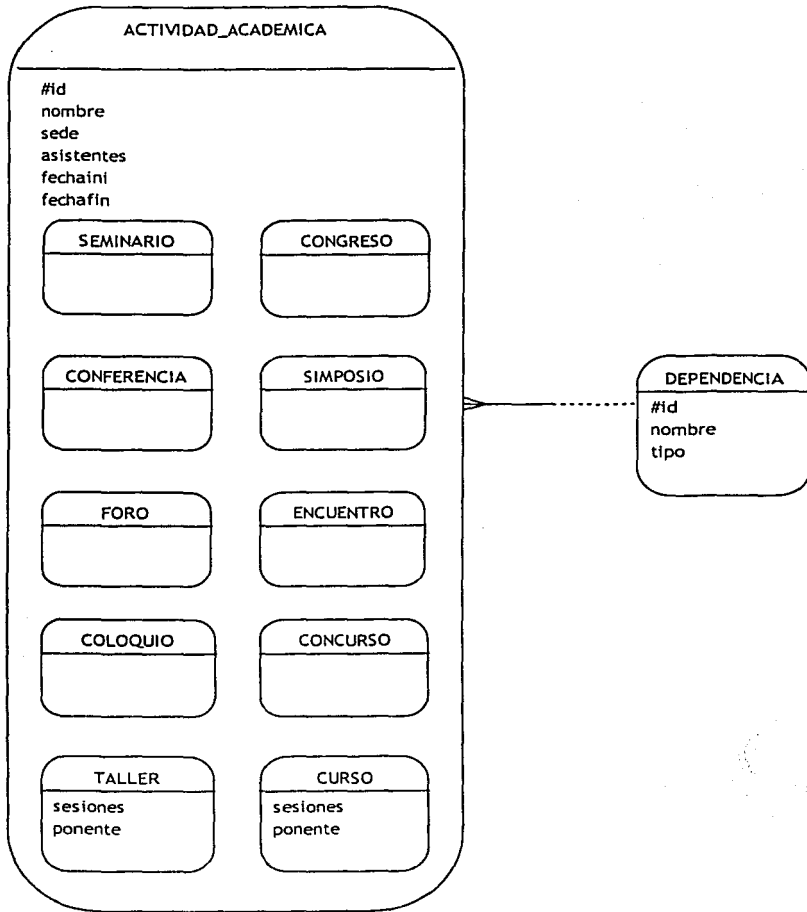


Figura 3.8 Diagrama E-R Actividades académicas.

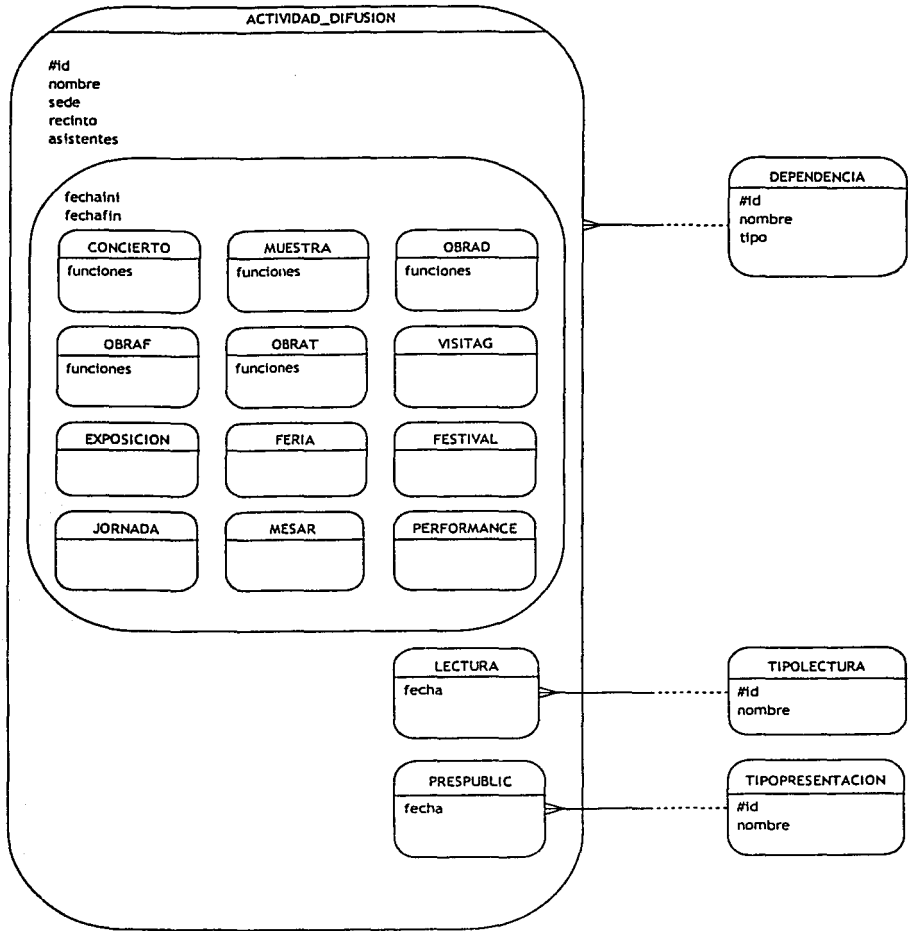


Figura 3.9 Diagrama E-R Actividades de difusión.

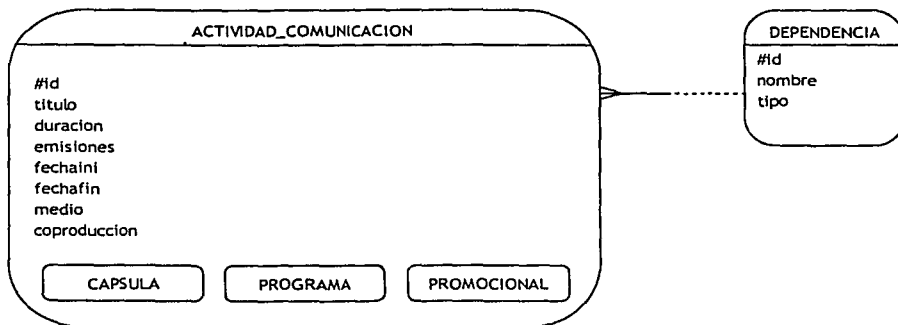


Figura 3.10 Diagrama E-R Actividades de comunicación.

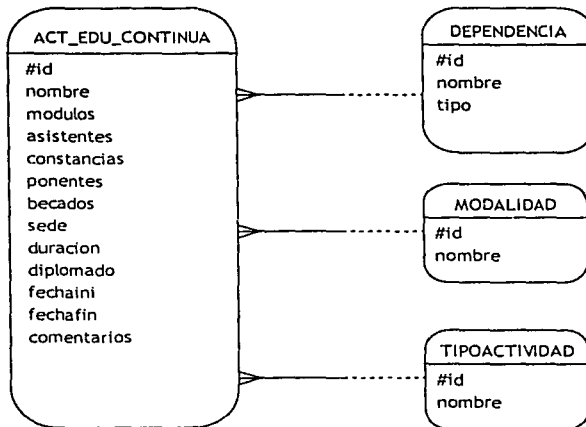


Figura 3.11 Diagrama E-R Actividades de educación continua.

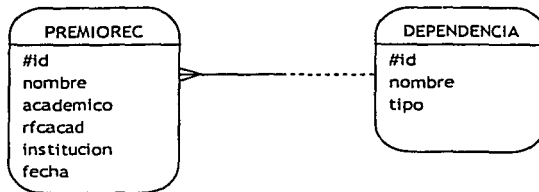


Figura 3.12 Diagrama E-R Premios y distinciones recibidos.

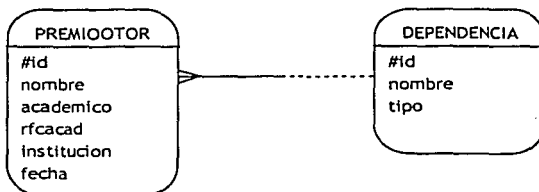


Figura 3.13 Diagrama E-R Premios y distinciones otorgados.

Por último, en la siguiente figura se muestra de forma más detallada la relación entre las entidades DEPENDENCIA y USUARIO.



Figura 3.14 Diagrama E-R usuario.

### 3.4 Diseño de la interfaz del usuario

Pensando en que uno de los principales objetivos consiste en mejorar la usabilidad del sistema, es decir, su facilidad de uso, se ha planteado la necesidad de rediseñar la interfaz para hacerla más consistente en todos los aspectos, diseño, navegación, contenidos, etc.

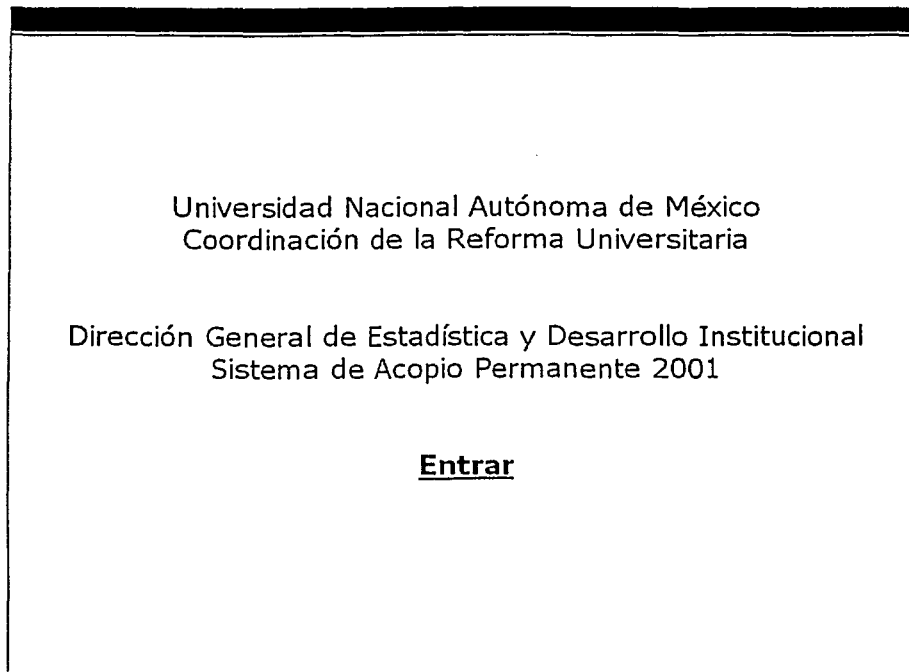


Figura 3.15 Entrada al sistema.

En la figura 3.15 se muestra la pantalla de bienvenida y de presentación del sistema que será mostrada cuando el usuario quiera acceder al sistema.

Al seleccionar la opción "Entrar", se pedirá al usuario que proporcione su login y su password, en caso de que estos datos no sean válidos, se le denegará el acceso.



Quando los datos proporcionado por el usuario son correctos, el usuario está en la posibilidad de seleccionar el módulo con el que desea trabajar.

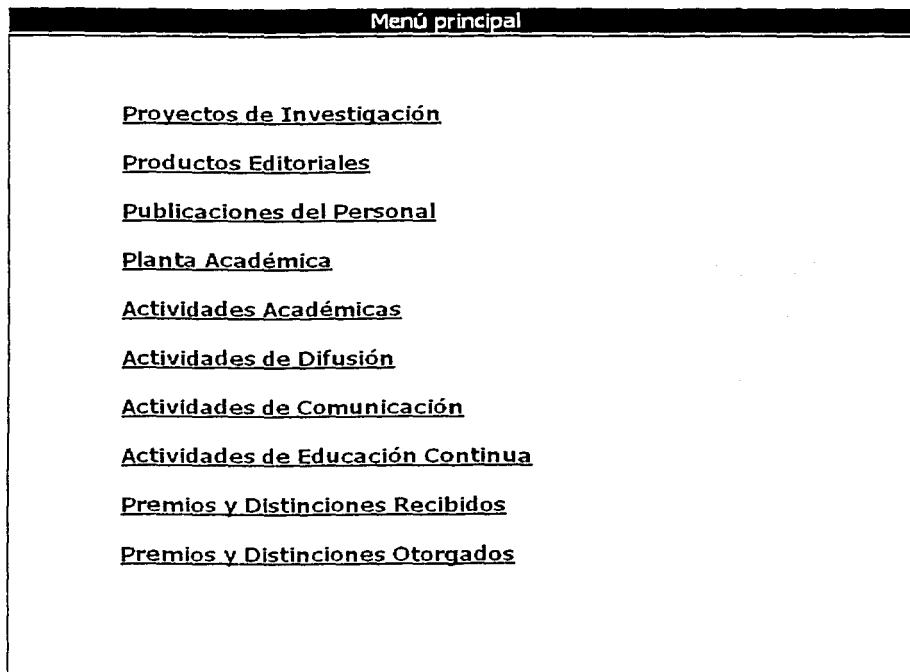


Figura 3.16 Principales módulos del sistema.

En la figura 3.16 se muestra la lista de los principales módulos en los cuales se clasifica la información.

Para los siguientes módulos, se tiene una clasificación adicional:

- ✓ Productos editoriales
- ✓ Publicaciones del personal
- ✓ Actividades académicas
- ✓ Actividades de difusión
- ✓ Actividades de comunicación

En el caso de que el módulo seleccionado por el usuario sea alguno de los listados anteriormente, se mostrará la clasificación adicional.

En la figura 3.17 se muestra un ejemplo de la clasificación adicional para el módulo de Actividades Académicas.

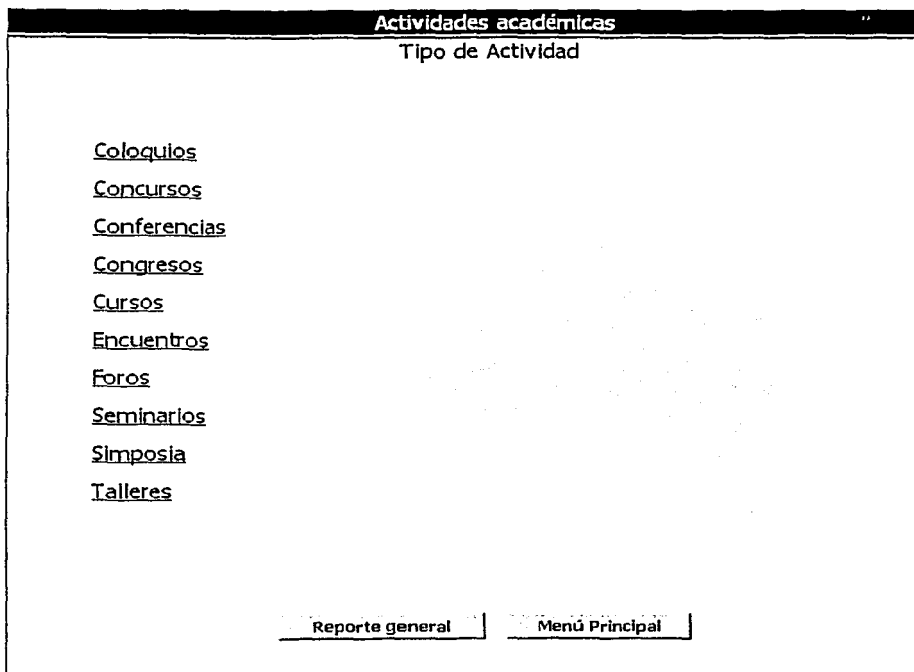


Figura 3.17 Clasificación adicional en el módulo Actividades Académicas.

En la figura 3.18 se muestra otro ejemplo de una clasificación adicional, en este caso para el módulo de Productos Editoriales.

Si el usuario se encuentra en alguno de estos módulos, siempre puede regresar al menú principal, el que se muestra en la figura 3.16.

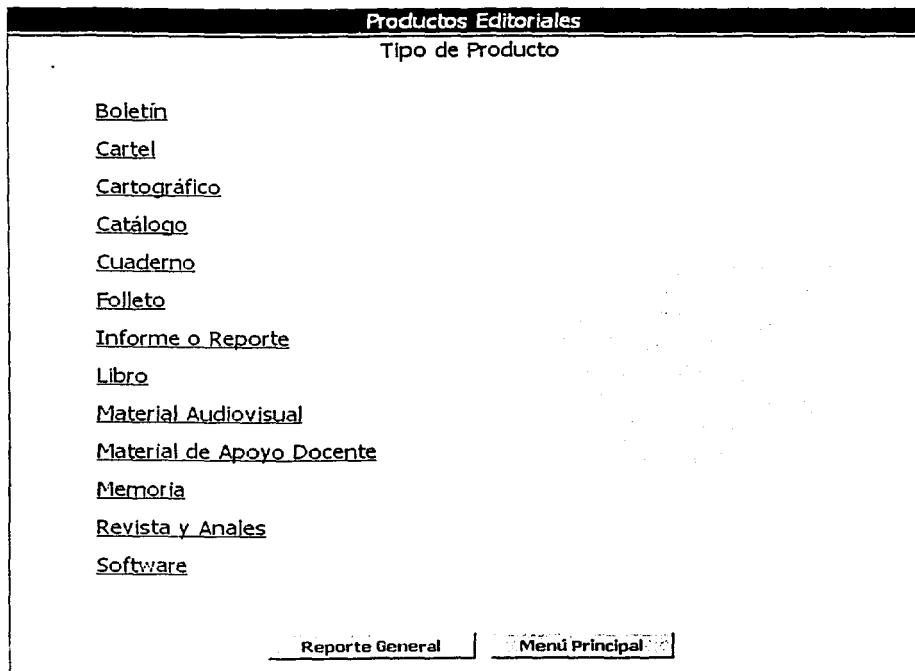


Figura 3.18 Clasificación adicional en el módulo Productos Editoriales.

La siguiente lista muestra los módulos que no contienen una clasificación adicional:

- ✓ Proyectos de investigación
- ✓ Personal académico
- ✓ Actividades de educación continua
- ✓ Premios y distinciones recibidos
- ✓ Premios y distinciones otorgados

Ya sea que el usuario seleccione alguno de los módulos de la lista anterior, o alguna de las opciones para los módulos con clasificación adicional, lo que se mostrará a continuación es la lista de registros que tiene actualmente capturados.

Esta lista muestra únicamente los registros que pertenecen a una dependencia en particular para la cual el usuario está realizando el acopio de información.

En la figura 3.19 se muestra un ejemplo de la lista de registros para la clasificación *Cartográfico* del módulo de Productos Editoriales.

**Productos Editoriales: Cartográfico**

Total registros: 8

1. Mapas y figuras de la geomorfología y procesos costeros en mares de México
2. Mapa vectorial de las carreteras de la República Mexicana
3. Mapa vectorial de aeropuertos de la República Mexicana
4. Mapa vectorial de Ferrocarriles de la República Mexicana
5. Mapa vectorial de la carretera México-Querétaro
6. Mapa vectorial de puertos de la República Mexicana
7. Rutas óptimas
8. Sistemas de información geográfica de las vialidades de Ciudad Universitaria

Figura 3.19 Lista de registros Productos Editoriales – Cartográfico.

La lista de registros se presenta ordenada alfabéticamente, y siempre se muestra el total de los registros capturados.

En la figura 3.20 se muestra otro ejemplo de la lista de registros para la opción *Cartel* del módulo de Productos Editoriales.

Lo que se debe de notar en la figura 3.20 es que únicamente se muestran 10 registros a la vez, y en el caso de que se tengan más de 10 registros capturados, se deben mostrar las opciones para pasar a otra página y ver otros diez registros.

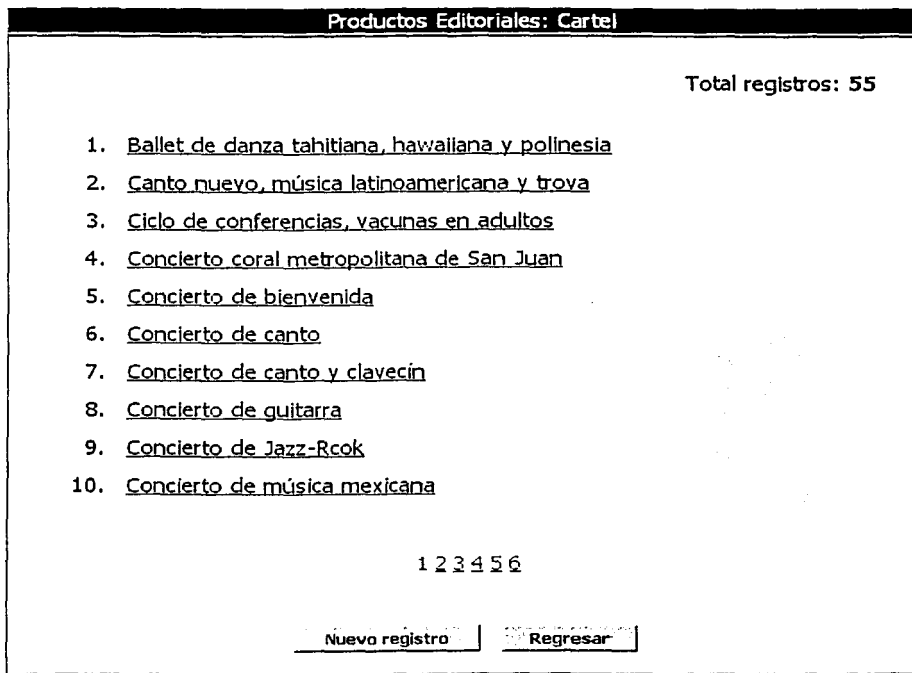


Figura 3.20 Lista de registros Productos Editoriales – Cartel.

En la figura 3.21 se muestra un ejemplo de la lista de registros para el módulo Actividades de Educación Continua.

Lo que se debe de notar en esta figura es que cuando el número de registros es demasiado grande, en lugar de mostrar 10 registros, se muestran en este caso 15 registros.

Cuando se tiene presente la lista de registros, siempre se está en la posibilidad de agregar un nuevo registro al seleccionar la opción "Nuevo registro".

Si en lugar de ello, se selecciona algún registro de la lista, se muestran los datos del registro completo permitiendo eliminarlo o actualizarlo.

Actividades de Educación Continua																															
																															Total registros: 476
1.	<u>Access 2000 avanzado</u>																														
2.	<u>Access 2000 básico</u>																														
3.	<u>Active server pages</u>																														
4.	<u>Actualización de estructuras hidráulicas</u>																														
5.	<u>Actualización en la planeación, desarrollo y evaluación de obras públicas</u>																														
6.	<u>Administración de la calidad (diplomado en gerencia de proyectos)</u>																														
7.	<u>Administración de la conservación de carreteras</u>																														
8.	<u>Administración de la energía, finanzas y medio ambiente</u>																														
9.	<u>Administración del mantenimiento (de la especialización virtual del</u>																														
10.	<u>Administración del mantenimiento (diplomado en administración del</u>																														
11.	<u>Administración del mantenimiento, segunda generación de alumnos</u>																														
12.	<u>Administración de proyectos</u>																														
13.	<u>Administración de recursos humanos y relaciones laborales</u>																														
14.	<u>Aguas superficiales (diplomado en uso eficiente del agua en ciudades)</u>																														
15.	<u>Análisis de costos, edificación y obra pesada</u>																														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
<input type="button" value="Nuevo registro"/>											<input type="button" value="Generar Reporte"/>											<input type="button" value="Menú principal"/>									

Figura 3.21 Lista de registros Actividades de Educación Continua.

En la figura 3.22 se muestra un ejemplo de la forma de captura de un nuevo registro, que se presenta cuando se selecciona la opción "Nuevo registro" en el módulo de Actividades de Educación Continua.

**Actividades de Educación Continua**

Nombre:

Modalidad: 

- Presencial
- Teleconferencia
- Videoconferencia**
- Multimedia
- Internet (en línea)
- Otro

 No. ponentes:

No. módulos:

No. asistentes:

No. constancias ó diplomas:

No. becados:

Sede:

Duración:

Fecha de Inicio:

Fecha de Término:

Tipo: 

- Curso
- Taller**
- Seminario
- Otro

 Diplomado: Si  No

Comentarios:

Figura 3.22 Nuevo registro en Actividades de Educación Continua.

En la figura 3.23 se muestra un ejemplo de la forma de captura de un nuevo registro, que se presenta cuando se selecciona la opción "Nuevo registro" en la clasificación *Boletín de Productos Editoriales*.

Siempre que se muestra la forma de captura de un nuevo registro las opciones principales son "Guardar cambios" para guardar los datos del registro, y "Cancelar", en el caso de que se decida no guardar los datos del nuevo registro.

Cuando se selecciona la opción "Guardar cambios", se procede a la validación de los campos del registro que son necesarios. Cuando los datos han sido validados, éstos son guardados, y a continuación se pasa a la lista de registros actualizada.

Si la opción elegida es "Cancelar", entonces se regresa a la lista de registros sin cambios.

**Forma de Productos Editoriales**

Tipo: Boletín

Título:

Editorial:

Ciudad:

Fecha de Tiraje:  Año:

Periodicidad:  s:  Tiraje:

n:  Número:

Autor:

Colaboradores / Co-autores:

Figura 3.23 Nuevo registro en Productos Editoriales – Boletín.

En la figura 3.24 se muestra un ejemplo de los datos que se presentan cuando se selecciona un registro de alguna lista de registros, en este caso, del módulo Actividades de Educación Continua.



**Actividades de Educación Continua**

Nombre:

Modalidad:  No. ponentes:   
No. módulos:   
No. asistentes:   
No. constancias ó diplomas:   
No. becados:

Sede:

Duración:

Fecha de Inicio:   Año

Fecha de Término:   Año

Tipo:  Diplomado:

Comentarios:

Figura 3.24 Datos de un registro Actividades de Educación Continua.

En la figura 3.25 se muestra otro ejemplo de los datos que se presentan cuando se selecciona un registro de alguna lista de registros, en este caso, de la clasificación "Boletín" del módulo Productos Editoriales.

Forma de Productos Editoriales	
Tipo: Boletín	
Título:	¿Qué tanto sabemos de los genios?
Editorial:	Facultad de Ingeniería, UNAM
Ciudad:	México, D.F.
Fecha de Tiraje:	1 Agosto Año 2001
Periodicidad:	Irregular
Total de Páginas:	16
Tiraje:	255
Volumen:	1
Número:	1
Autor:	Sánchez Sánchez, Fernando
Colaboradores / Co-autores:	
<input type="button" value="Guardar Cambios"/> <input type="button" value="Cancelar"/> <input type="button" value="Eliminar"/>	

Figura 3.25 Datos de un registro Productos Editoriales - Boletín.

Cuando se selecciona un registro de alguna lista de registros, se muestran los datos que contiene el registro, y se muestran además las opciones "Guardar cambios", "Eliminar" y "Cancelar".

Si se selecciona la opción "Guardar cambios", se realiza una actualización de los datos del registro, y a continuación se pasa a la lista de registros.

Si se selecciona la opción "Eliminar", se realiza la eliminación del registro, y se pasa a la lista de registros.

Si la opción seleccionada es "Cancelar", se regresa a la lista de registros que no debe de haber sido modificada.

De acuerdo a las pantallas mostradas anteriormente, la navegación a través del sistema es de la siguiente manera.

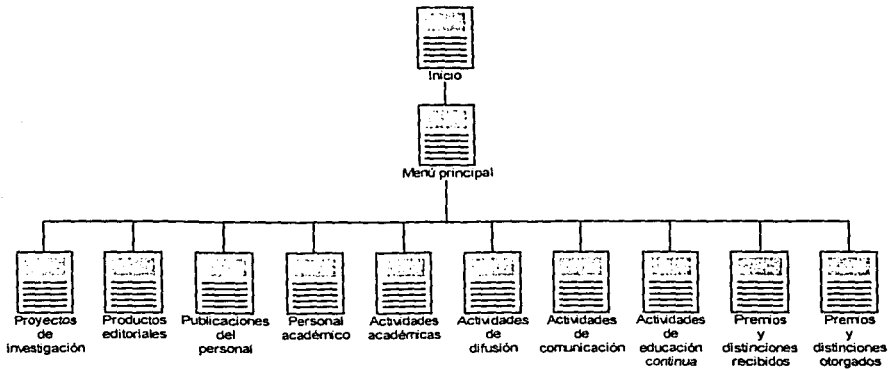


Figura 3.26 Navegación a través del sistema (a).

La figura 3.26 muestra la entrada al sistema, y la selección de alguno de los módulos en el que se va a trabajar.

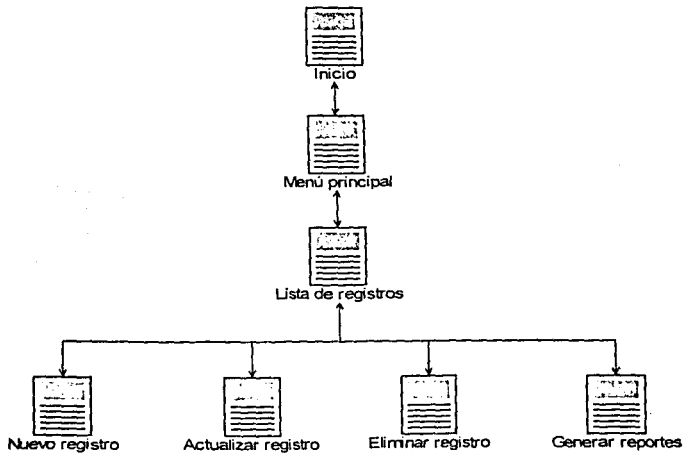


Figura 3.27 Navegación a través del sistema (b).

La figura 3.27 muestra la navegación a través del sistema cuando se selecciona alguno de los siguientes módulos.

- ✓ Proyectos de investigación
- ✓ Personal académico
- ✓ Actividades de educación continua
- ✓ Premios y distinciones recibidos
- ✓ Premios y distinciones otorgados

La información de la lista de registros, y de cada una de las opciones subsecuentes corresponde a alguno de estos módulos.

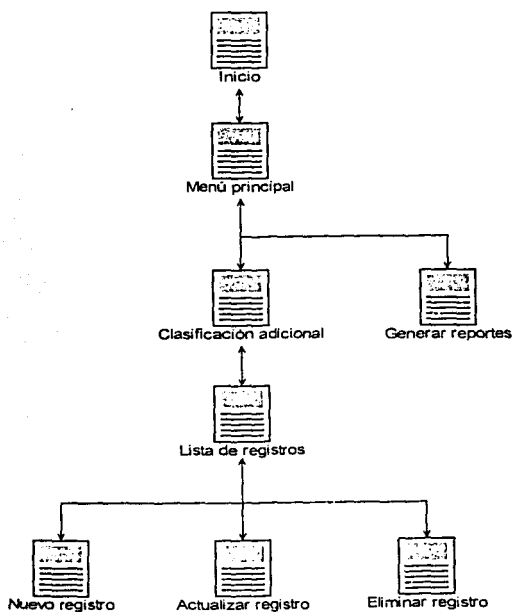


Figura 3.28 Navegación a través del sistema (b).

La figura 3.28 muestra la navegación a través del sistema cuando se selecciona alguno de los siguientes módulos.

- ✓ Productos editoriales
- ✓ Publicaciones del personal
- ✓ Actividades académicas
- ✓ Actividades de difusión
- ✓ Actividades de comunicación

La información de la clasificación adicional, de la lista de registros, y de cada una de las opciones subsecuentes corresponde a alguno de estos módulos.

**ESTA TESIS NO SALE  
DE LA BIBLIOTECA**

## CAPÍTULO 4

# CONSTRUCCIÓN DEL SISTEMA DE ACOPIO PERMANENTE

## 4.1 Las aplicaciones web

World Wide Web (WWW, el web) es uno de los principales servicios que se pueden encontrar en Internet. Su éxito ha sido tal, que a menudo se identifica Internet con el web, pero son cosas diferentes.

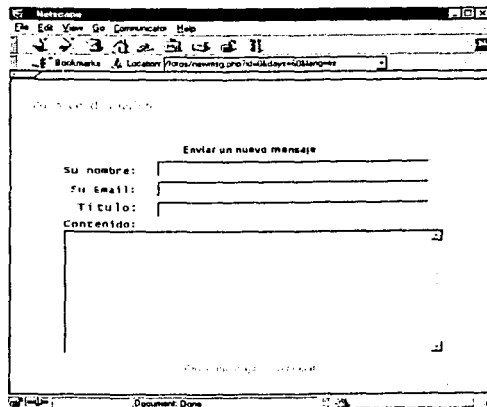
Internet es una red mundial de computadoras que intercambian información basándose en el protocolo TCP/IP. Todas las computadoras utilizan de alguna manera este protocolo cuando están conectadas a Internet.

La cantidad de información disponible a través de Internet es inmensa. El acceso a la misma se lleva a cabo de diversas maneras y formas de acceso: correo electrónico, ftp, videoconferencia, news, y el web, entre otros. El web es pues una de las formas de acceder a la información que se encuentra en Internet, información que en su gran mayoría se compone de documentos HTML, comúnmente llamados páginas web.

Una aplicación web es cualquier aplicación que utiliza el Protocolo de Transferencia de Hipertexto o HTTP como principal protocolo de transporte y de intercambio de información. HTTP es el protocolo fundamental en el web, por lo que su definición abarca las aplicaciones que se utilizan en el web. El acceso a la información a través del web es tan versátil y potente que no es de extrañar que en poco tiempo se haya convertido en uno de los servicios preferidos por la mayoría de los usuarios de Internet.

## 4.2 Herramientas para acceder datos mediante el web

Una página web es construida fundamentalmente utilizando el lenguaje HTML, por lo que el HTML es el instrumento más elemental en la construcción de cualquier aplicación web. Una de las características más útiles y prácticas de este lenguaje son los formularios.



The image shows a screenshot of a Netscape browser window. The title bar reads "Netscape". The address bar contains "http://www.mozilla.org/". The main content area displays a form titled "Enviar un nuevo mensaje" (Send a new message). The form includes the following fields:

- Su nombre: [text input]
- Su email: [text input]
- Título: [text input]
- Contenido: [text area]

At the bottom of the form, there is a "Enviar" (Send) button. The browser's status bar at the bottom indicates "Document Done".

TESIS CON  
FALTA DE CÍGEN

Figura 4.1 Ejemplo de un formulario HTML.

Las figuras 4.1 y 4.2 muestran dos ejemplos de formularios que están siendo utilizados en alguna aplicación.

The image shows a Netscape browser window with a menu bar (File, Edit, View, Go, Communicate, Help) and a toolbar. The address bar contains 'http://www.admin/asegpa\_grupo.php'. The main content area displays a form with the following elements:

- Title: Ingrese los datos del asesor para el nuevo grupo
- Form fields:
  - Nombre: [text input]
  - Correo electrónico: [text input]
  - Acerca del asesor: [text area]
  - Fotografía del asesor: (Formato GIF) [text input] with a 'Browse...' button
- Buttons: 'Aceptar' and 'Cancelar' at the bottom.

Figura 4.2 Otro ejemplo de un formulario HTML.

Los formularios permiten a un usuario introducir información en una página web para que los datos se transfieran a otra página, o a un programa externo para su procesamiento.

La herramienta que proporciona la interfaz gráfica para el usuario, y que permite acceder al contenido de las páginas web, es el navegador web. Existen varios tipos disponibles de navegadores y todos proporcionan por lo menos un conjunto mínimo de funciones.

El navegador constituye la mitad de lo necesario en una aplicación web, la otra mitad es un servidor web. Un servidor web es un programa que atiende las peticiones enviadas por un navegador, e independientemente de cuál haya sido esta petición, lo que el servidor web le manda como respuesta en la gran mayoría de los casos es un documento HTML.

De hecho, el web se diseñó originariamente como un medio para publicar información en forma de documentos HTML. Cuando un navegador web envía una petición de un documento HTML a un servidor web, éste último extrae la información del documento solicitado de su sistema de archivos y lo devuelve a través de la conexión HTTP.

Los documentos HTML se suministran en los sistemas de archivos tradicionales, por lo que no se requieren sistemas de gestión para almacenar, editar y recuperar estos documentos. Sin embargo, los sistemas de archivos sólo ofrecen capacidades mínimas de gestión, como el control de acceso y la exclusión mutua, por lo que los documentos HTML tienden a ser estáticos, y por lo tanto, se les hace poco apropiados para almacenar información que cambia constantemente.



Sin embargo, lo que devuelve el servidor web no tiene que ser siempre un documento HTML estático almacenado en el sistema de archivos, sino un documento cuyo contenido ha sido generado dinámicamente, por ejemplo al extraer información de una base de datos, y es en este caso cuando resulta necesario emplear alguna herramienta que sea capaz de generar las páginas web dinámicamente.

### 4.3 El software libre en el desarrollo de aplicaciones web

- ✓ Cuando se habla de software libre, se debe de hablar desde el sentido más filosófico de la palabra libertad. Se habla de la libertad de tener un programa completo (incluyendo su código fuente), la libertad de usarlo, de copiarlo, de modificarlo, de venderlo, de compartirlo. Este es el espíritu de muchos proyectos de software que se desarrollan actualmente.
- ✓ El software libre presenta una innumerable cantidad de ventajas para el desarrollador frente a otros sistemas formados bajo modelos cerrados. La primera y principal es que el desarrollador puede obtener ayuda y retroalimentación de personas a las que quizás ni siquiera conoce personalmente, pero con las cuales puede tener contacto gracias a Internet.
- ✓ El desarrollo de proyectos bajo estas características permite que crezcan mucho más rápido, gracias a la cantidad de colaboradores que en un momento dado pueden tener. Es decir, los proyectos de software libre se basan generalmente en la colaboración de cientos o miles de personas alrededor del mundo que tienen una sola cosa en común: el espíritu de colaboración.

Los últimos años han sido buenos para el software libre, los desarrolladores que lo crearon, y los usuarios que lo usan confían en él. El crecimiento exponencial de sistemas instalados basados en Linux continúa sin parar, y las tecnologías clave del lado del servidor basadas en el software libre se consideran componentes de misión crítica en infinidad de aplicaciones.

Linux como sistema operativo, Apache como servidor web, PostgreSQL como manejador de bases de datos y PHP como lenguaje de programación son sólo algunos ejemplos de proyectos de software libre realmente competitivos, que permiten desarrollar aplicaciones web bastante robustas, y con un grado de desempeño y confiabilidad realmente muy bueno.

Gobiernos e instituciones de todo el mundo, especialmente en países en vías de desarrollo, están implantando soluciones basadas en Linux para minimizar sus costes. Y todavía más notable es que ya se pueden señalar exitosos modelos de negocios construidos sobre el desarrollo, el servicio y el soporte del software libre.

Conforme las compañías que usan el web comienzan a necesitar sistemas de tecnología más flexibles, potentes y bien equipados, muchas están prestando atención a las opciones del software libre. Desde los sistemas operativos y servidores web hasta sus sistemas de bases de datos, un creciente número de empresas e instituciones están encontrando que las aplicaciones de software libre proporcionan alternativas bastante buenas con respecto a los exorbitantes costos de licencias y a los lentos ciclos de desarrollo que caracterizan a las aplicaciones propietarias.

El movimiento del software libre es, en pocas palabras, una de las revoluciones informáticas más grandes de los últimos años. Cabe señalar que en el fondo, el movimiento del software libre tiene algo de complejidad, ya que no se puede explicar, ni resumir en una frase. No obstante, usar el término *Open Source* para referirse al software libre suministra algún tipo de indicación única desde el punto de vista funcional.

Así, es muy común referirse al software libre como software *Open Source*, ya que proporciona un punto de partida neutral desde el que se puede explicar de qué se trata este movimiento. Bajo este concepto, el beneficio real del software *Open Source* no es el apreciado ideal de la libertad, sino más bien su tendencia a ser muy fiable y flexible.

### 4.3.1 Sistema operativo Linux

El fenómeno Linux proviene de toda una serie de factores, pero los más destacados son sus orígenes en una comunidad internacional de programadores que trabajan en un entorno abierto y en continua evolución para la que resulta más importante tener un sistema operativo que concentre la inteligencia y la experiencia colectiva, que estar pensando en derechos y copyrights.

Mientras que la industria de las tecnologías de la información ha visto emerger sistemas operativos con mucho bombo en el pasado OS/360, VMS, Windows, Solaris y AIX por mencionar algunos, ninguno se ha desarrollado tan rápido y se ha extendido tan ampliamente como Linux. En su relativamente corta vida, Linux se ha convertido ya en uno de los sistemas operativos más importantes para servidores.

Linux es una base ideal para innovaciones alrededor de las últimas tecnologías. La comunidad Linux continúa mejorando la funcionalidad básica de Linux, lo que permite contar con una plataforma abierta, escalable y económica para desarrollar clases completamente nuevas de aplicaciones que hasta ahora no han sido factibles.

De hecho, Linux ya no es sólo esto, en estos momentos es para muchos algo más que un sistema operativo, es casi una filosofía de vida, es una nueva forma de ver y entender el mundo de los sistemas operativos, de las aplicaciones y en definitiva una manera nueva y revolucionaria de entender el mundo de la informática.

El gran esfuerzo de desarrollo y la amplia aceptación y distribución del sistema operativo Linux aseguran que tome posición como una alternativa real, viable y significativa a los sistemas operativos comerciales.

### 4.3.2 Servidor web Apache

Según las últimas encuestas de Netcraft <sup>1</sup> el 60 % de todas las páginas web son servidas por servidores web Apache. Los competidores en este campo son Microsoft y Netscape/iPlanet con unos índices muy rezagados del 19.6 % y 7% respectivamente. Una explicación preliminar puede encontrarse en que su precio lo convierte en un ganador instantáneo, ya que puede usarse sin la necesidad de pagar un solo centavo, si embargo ésta no es la razón principal de su amplia aceptación.

Apache fue diseñado con las siguientes metas en mente; que implementase completamente los estandartes HTTP, que fuese sólido, fiable, escalable y exportable a casi cualquier sistema operativo. Apache ha demostrado ser sólido y flexible en su uso para infinidad de sitios web que tienen gran carga y demanda.

En sus inicios, Apache era un derivado del clásico NCSA HTTPD, un popular servidor web que luego fue descontinuado. Hoy, Apache ha logrado convertirse en un producto más robusto, gracias a los cientos de usuarios que colaboran con el proyecto día a día.

### 4.3.3 Manejador de bases de datos PostgreSQL

PostgreSQL empezó como un proyecto de la Universidad de California, en Berkeley entre 1977 y 1985, y a lo largo de este tiempo ha pasado por una serie de etapas que lo han convertido en uno de los manejadores de bases de datos más populares hoy en día. Por fortuna, su edad no vuelve a PostgreSQL un producto anticuado, por el contrario, ha tenido tiempo para madurar hacia un sistema de bases de datos relacionales fiable y robusto. PostgreSQL está disponible bajo la licencia *Open Source* de Berkeley, y forma parte de muchas distribuciones de Linux.

De las opciones *Open Source*, PostgreSQL es el manejador de bases de datos objeto-relacional más robusto. PostgreSQL, también conocido como Postgres, ha sido *Open Source* desde sus inicios y evoluciona rápidamente debido a la activa y comprometida comunidad de desarrolladores que tiene detrás. De una revisión a otra se añaden grandes características en cuestión de meses, equivalente a actualizaciones de compañías de software comercial que tardan años en introducir. La inaudita frecuencia de revisiones procede de cientos de leales y entusiastas desarrolladores y usuarios cuyo torrente de ideas arregla y prueba los últimos detalles, comunicando rápidamente los resultados obtenidos desde toda clase de aplicaciones, configuraciones y plataformas.

En su última versión PostgreSQL es de entrada totalmente compatible con ANSI SQL-92, soportando la mayoría de las construcciones SQL, incluyendo transacciones y subconsultas, así como tipos y funciones definidas por el usuario.

---

<sup>1</sup> <http://www.netcraft.com>

La estabilidad y la redundancia son siempre consideraciones importantes a tener en cuenta cuando se quiere implantar una base de datos en sistemas críticos. Durante los últimos años, mucha de la labor de PostgreSQL se ha concentrado en estas áreas. El cierre al nivel de filas y el soporte para la replicación hacen de PostgreSQL bastante robusto a la vez que aumenta los buenos resultados en entornos con alto número de transacciones.

Estas prestaciones permiten entre otras cosas que resulte más fácil y confiable obtener respaldos de las bases de datos.

PostgreSQL corre en una amplia variedad de sistemas operativos, incluyendo Linux, Solaris, Iris, AIX y Windows NT/2000. PostgreSQL proporciona una poderosa infraestructura técnica para un amplio rango de aplicaciones que requieren sitios web, intranets y extranets.

Un entorno potente de bases de datos no sirve de nada sin buenas herramientas de desarrollo, de interfaces de programación de aplicaciones, es decir, de bibliotecas que ayuden a construir aplicaciones para sacar partido de las características y de la funcionalidad que ofrecen los manejadores de bases de datos.

Como cualquier manejador de bases de datos importante, PostgreSQL soporta métodos de acceso a través de ODBC y JDBC, además de ofrecer soporte para otros lenguajes de programación como Perl, Python, PHP, Tcl/Tk, C y C++, con lo que se tiene la oportunidad de decidir cuál es el lenguaje de programación más indicado, de acuerdo al tipo de aplicación que se vaya a desarrollar.

#### 4.3.4 Lenguaje de programación PHP

El desarrollo de un sitio web que sea capaz de generar contenido dinámico requiere sin duda de la adecuada elección de un lenguaje de programación que se ejecute del lado del servidor web. Las opciones disponibles son diversas, ASP, JSP, Cold Fusion, etc, cada uno de las cuales puede producir una solución satisfactoria para determinadas necesidades. Sin embargo, en Linux y otros sistemas Unix destaca PHP, uno de los lenguajes de programación web más importantes del momento, libre, gratuito y abierto. PHP al igual que los lenguajes mencionados anteriormente, se denominan lenguajes de *script*.

Se mencionan a continuación algunas de las características de estos lenguajes tratando de mostrar algunas de sus ventajas y desventajas.

##### Cold Fusion

Éste es un producto comercial de Allaire Corporation, está basado en un lenguaje de etiquetas que se intercalan con el código HTML, Cold Fusion está disponible en las siguientes plataformas: Win32, Solaris, Linux y HP/UX y tiene un excelente conjunto de herramientas de administración, monitoreo, desarrollo y *debugging*. El costo actual es de aproximadamente \$1295 dólares. Cold Fusion está diseñado para no programadores.

## ASP

ASP no es realmente un lenguaje de programación por sí mismo, ASP es el acrónimo de *Active Server Pages*, los lenguajes mayormente usados para programar ASP son Visual Basic Script (VBScript) y Jscript, la versión de Microsoft del lenguaje Java Script. Uno de los principales puntos débiles de ASP, es que éste es un sistema propietario que es nativamente usado sólo en los servidores web de Microsoft, el Internet Information Server (IIS) y el Personal Web Server (PWS), bajo su plataforma Windows. Aunque es cierto que es posible usar ASP en otras plataformas, esto sólo es posible a través de otros productos comerciales: InstantASP de Halcyon, y ChiliSoft ASP de ChiliSoft, además la falta de soporte para componentes COM fuera de Windows limita muchísimo la practicidad de esta alternativa.

Una de las ventajas de ASP, es que es muy fácil de usar si ya se conoce como programar en Visual Basic, además de que el soporte para ASP está habilitado por omisión en el IIS, por lo que es muy fácil comenzar a desarrollar y ejecutar aplicaciones en un entorno Windows. Por otra parte, el número de componentes que se proporcionan de manera gratuita es muy limitado, así que si se necesita alguna herramienta "avanzada", se tienen que comprar los componentes adecuados.

## JSP

JSP es el acrónimo de *Java Server Pages*, y es la alternativa de SUN Microsystems en los lenguajes de *script*, para crear páginas web interactivas que forman parte de una aplicación web. Al ser una tecnología Java, los programas en JSP están diseñados para ser independientes de la plataforma, y a su vez, independientes del servidor, es decir, se adhiere a la filosofía "Escríbelo una vez, Córrelo donde sea".

Cabe mencionar que Sun está licenciando el código fuente de las tecnologías JSP y Java Servlet a la fundación Apache, para que sean desarrolladas y liberadas bajo el proceso de desarrollo de Apache, por lo que se presenta como una tecnología flexible y abierta que trabaja con un amplia variedad de herramientas ofrecidas por los vendedores y que refleja la cooperación y colaboración de la industria.

De manera análoga a Visual Basic y ASP, el desarrollo de aplicaciones en JSP debe de ser algo muy fácil para los programadores en Java.

## PHP

PHP (*Hypertext Preprocessor*, o bien, preprocesador de hipertexto) es un lenguaje de programación *Open Source* que ofrece una solución simple y completa para la creación de aplicaciones que requieran del manejo de información de manera dinámica e interactiva a través del web. PHP es un lenguaje de script rápido, estable, simple, configurable, independiente de la plataforma, y que se ejecuta del lado del servidor.

PHP se ha convertido en uno de los lenguajes de script para el desarrollo de aplicaciones web más utilizados, ya que permite a un desarrollador crear sitios web más potentes y flexibles de una manera mucho más fácil. PHP es utilizado hoy en día por más de 7 millones de sitios web en Internet, y este número seguramente seguirá creciendo.

Apache debería de ser la elección ideal para desarrollar aplicaciones web bajo Linux, sin embargo PHP se ejecuta igualmente bien en cualquier otro Unix, o sistema Windows, corriendo algún otro servidor web.

La extensa librería de funciones que posee PHP facilitan la programación, y permiten realizar prácticamente cualquier tipo de aplicación web. De hecho, una de las características más importantes que tiene, es el soporte que ofrece para un gran número de manejadores de bases de datos, entre los que se incluyen Oracle, Sybase, Informix, DB2, MySQL y PostgreSQL.

Al igual que PHP, ASP, Cold Fusion y JSP pueden funcionar de manera conjunta con el servidor web sin tener que expandir procesos adicionales por lo que no tienen problemas cuando sucede un tráfico denso a través de múltiple sitios web.

Cuando se compara PHP con otras tecnologías como JSP, ASP o Cold Fusion, de manera general PHP tiene un menor costo, con un mayor rendimiento y se integra muy bien con otras tecnologías, además de proporcionar un entorno de desarrollo de producto rápido.

PHP toma características de lenguaje C, Perl, y Java, por lo que el aprendizaje de este lenguaje debe de ser fácil para los programadores que tengan experiencia en alguno de estos lenguajes.

El futuro parece brillar para el software *Open Source*, y en particular para PHP, lo que le permitirá seguir estando a la vanguardia en las tecnologías web durante los años venideros.

#### 4.4 Configuración del sistema operativo

Linux es el núcleo del sistema operativo, el kernel. Linux y su colección de software se conocen formalmente con el nombre de distribución. Debido a que Linux es de carácter libre, no hay ninguna organización concreta que asuma la normalización de sistema. La consecuencia es la gran variedad de distribuciones que se pueden encontrar hoy en día, sin embargo la mayoría de las distribuciones más populares tienen básicamente las mismas aplicaciones, por lo que la elección de una distribución concreta es más o menos arbitraria.

Algunas de las distribuciones de Linux más populares son: SuSE<sup>2</sup>, Mandrake<sup>3</sup>, Debian<sup>4</sup>, Slackware<sup>5</sup>, Caldera<sup>6</sup> y Red Hat<sup>7</sup>.

En particular, para el desarrollo del Sistema de Acopio Permanente se decidió utilizar Red Hat en su versión 7.0, que es la que en el momento se tenía disponible y era en ese tiempo la más actual. La decisión de utilizar Red Hat Linux como sistema operativo no fue tarea

---

<sup>2</sup> <http://www.suse.com>

<sup>3</sup> <http://www.mandrakesoft.com>

<sup>4</sup> <http://www.debian.org>

<sup>5</sup> <http://www.slackware.com>

<sup>6</sup> <http://www.caldera.com>

<sup>7</sup> <http://www.redhat.com>

difícil, ya que el uso y la experiencia con el mismo habían sido bastante satisfactorios desde hace un par de años que se usaba en la DGEDI.

Hablar hoy en día de Red Hat es hablar no sólo de uno de los más clásicos y veteranos sistemas Linux, sino también de una de las mayores, más completas y más fiables distribuciones. No en vano, Red Hat se ha convertido en uno de los principales estándares en cuanto a distribuciones se refiere, y en la que se basan muchas otras. Por si fuera poco, la empresa Red Hat tiene acuerdos importantes de colaboración con compañías que se están integrando de alguna manera al mundo Linux, tales como Oracle, IBM, Dell, HP y Compaq.

La gran diferencia entre Red Hat y muchas otras distribuciones, es que Red Hat ha sido capaz de evolucionar de una forma más rápida y de incluir herramientas gráficas que permiten una instalación, administración y configuración más rápida, cómoda y fácil de un servidor.

En efecto, la instalación de Linux Red Hat es bastante sencilla y automática con su programa de instalación, que puede ocuparse de muchas configuraciones y problemas diferentes, solucionando la mayoría de los mismos. Red Hat es también una de las empresas líderes en el desarrollo de Linux en incorporar las características de seguridad más actuales, además de esforzarse en proporcionar las últimas versiones del software Linux.

Uno de los puntos más fuertes de esta distribución, es la aplicación denominada RPM (Red Hat Package Manager), ya que permite instalar y desinstalar paquetes de software, incluyendo componentes del sistema operativo, de una forma muy transparente. El impacto del avanzado sistema de manejo de paquetes RPM es tal, que de hecho es usado por muchas otra distribuciones de Linux.

Precisamente, el proceso de instalación y configuración del software necesario para el desarrollo del Sistema de Acopio Permanente se describe a continuación haciendo uso de los paquetes (comúnmente llamados rpms) que se incluyen en los discos de instalación.

El equipo en el que se instaló Red Hat Linux 7.0 cuenta con las siguientes características:

- ✓ Procesador Intel Pentium III a 800Mhz
- ✓ 512 MB de memoria RAM
- ✓ 2 discos duros SCSI de 17 GB cada uno

Después de la instalación básica del sistema operativo, lo que se tiene que hacer a continuación es asegurarse de que han sido instalados los paquetes necesarios de Apache, de PHP, y de PostgreSQL.

Los paquetes que deben estar instalados son:

Servidor Apache	apache-1.3.12-25.i386.rpm
PHP	php-4.0.1pl2-9.i386.rpm
Módulo de PHP para Apache	mod_php-4.0.1pl2-9.i386.rpm
Servidor PostgreSQL	postgresql-7.0.2-17.i386.rpm postgresql-server-7.0.2-17.i386.rpm
Extensión de PostgreSQL para PHP	php-pgsql-4.0.1pl2-9.i386.rpm
Módulo de autenticación para Apache basada en PostgreSQL	mod_auth_pgsq-0.8-3.i386.rpm

Tabla 4.1 RPMs que se requiere que sean instalados.

La instalación se lleva a cabo utilizando el comando rpm:

```
[root@hades]# rpm -ivh apache-1.3.12-25.i386.rpm
[root@hades]# rpm -ivh php-4.0.1pl2-9.i386.rpm
[root@hades]# rpm -ivh mod_php-4.0.1pl2-9.i386.rpm
[root@hades]# rpm -ivh postgresql-7.0.2-17.i386.rpm
[root@hades]# rpm -ivh postgresql-server-7.0.2-17.i386.rpm
[root@hades]# rpm -ivh php-pgsq-4.0.1pl2-9.i386.rpm
[root@hades]# rpm -ivh mod_auth_pgsq-0.8-3.i386.rpm
```



## 4.5 Configuración del servidor web

Los archivos de configuración del servidor web Apache se ubican dentro del directorio `/etc/httpd/conf`. En este directorio debe estar el archivo `httpd.conf`, que es el archivo principal de configuración de Apache.

A continuación se detalla un poco el contenido de el archivo `http.conf`.

```
# Permite que Apache se ejecute de manera independiente, y no desde el
# demonio inetd

ServerType standalone

# Puerto en el que escucha las peticiones. Este es el valor estándar.

Port 80

# Para traducir la dirección IP de los clientes al registrar los accesos
al sitio.

HostnameLookups on

# Usuario bajo el que se ejecuta el servidor

User nobody

# Grupo al que pertenece el usuario que ejecuta el servidor

Group nobody

# Dirección de correo del responsable de la administración del servidor
# web a la que se enviaran mensajes en caso de error.

ServerAdmin admin@sap.estadistica.unam.mx

# Directorio bajo el cual se encuentran los archivos de configuración.

ServerRoot /etc/httpd

# Archivo en el que se almacenan los registros de errores

ErrorLog /var/log/httpd/error_log

# Archivos en el que se almacenan los registros de acceso

TransferLog /etc/httpd/logs/access_log

# Para permitir la existencia de conexiones persistentes (HTTP 1.1), es
# decir, que por una misma conexión se puedan enviar varias peticiones
# HTTP.

KeepAlive On
```

```
# Número máximo de peticiones que se pueden enviar por una misma conexión.  
# En caso de poner 0, el número será ilimitado. Cuantas más peticiones se  
# puedan enviar por una conexión, mejor será el rendimiento de la  
# comunicación.
```

```
MaxKeepAliveRequests 100
```

```
# Número de segundos máximo en el que se esperará por la siguiente  
# petición.
```

```
KeepAliveTimeout 15:
```

```
# Número de servidores que se lanzan inicialmente.
```

```
StartServers 8
```

```
# Número máximo de clientes que pueden conectarse al servidor de web.
```

```
MaxClients 150
```

```
# Nombre del servidor web. Este es el nombre que se debe de escribir en la  
# barra de direcciones del navegador cuando alguien quiera acceder al  
# sistema. (http://sap.estadistica.unam.mx/)
```

```
ServerName sap.estadistica.unam.mx
```

```
# Directorio principal bajo el que se guardan los archivos accesibles  
# por el servidor. En este directorio se guardan todos los archivos HTML,  
# las imágenes, y los programas escritos en PHP que van a ser  
# usados por el sistema.
```

```
DocumentRoot /var/www/html/sap
```

```
# Se carga el módulo de autenticación basado en PostgreSQL. Este módulo  
# se incluye en el paquete mod_auth_pgsq-0.8-3
```

```
LoadModule auth_pgsq_module /usr/lib/apache/mod_auth_pgsq.so  
AddModule mod_auth_pgsq.c
```

```
# Se carga el módulo de PHP.  
# Este módulo se incluye en el paquete mod_php-4.0.1pl2, y  
# es necesario para que Apache tenga soporte para PHP, y pueda  
# ejecutar los scripts escritos en PHP.
```

```
LoadModule php4_module modules/libphp4.so  
AddModule mod_php4.c
```

```
# Se asocian los archivos con las extensiones php y phtml con PHP. Los  
# programas escritos en PHP deben de tener alguna de estas extensiones  
# para que puedan ser ejecutados de manera correcta.
```

```
<IfModule mod_php4.c>  
    AddType application/x-httpd-php .php .phtml  
</IfModule>
```

```
# Se protege el directorio en el que se almacenan todos los archivos
# usados en el sistema. Sólo los usuarios autorizados podrán acceder
# a estos archivos, y por lo tanto, hacer uso del sistema.
#
# Los nombres de usuario y contraseñas que se proporcionan a los usuarios
# son almacenados en una tabla de una base de datos PostgreSQL. En este
# caso resulta muy útil que esa tabla esté dentro de la base de datos
# principal del sistema.
#
#      Base de datos: sap
#      Tabla: usuarios
#
# +-----+-----+
# | Campo          | Tipo  |
# +-----+-----+
# | login          | text  |
# | password       | text  |
# | grupo          | text  |
# +-----+-----+
#
# Dado que el sistema está pensado para ser utilizado únicamente por las
# entidades académicas de la UNAM, se está limitando además el acceso al
# sistema para usuarios dentro de la RedUNAM (subredes 132.248.x.x)
```

```
<Directory /var/www/html/sap>
  AuthType basic
  AuthName "Sistema de Acopio Permanente"
  Auth_PG_host localhost
  Auth_PG_port 5432
  Auth_PG_user postgres
  Auth_PG_database sap
  Auth_PG_pwd_table usuarios
  Auth_PG_grp_table usuarios
  Auth_PG_uid_field login
  Auth_PG_pwd_field password
  Auth_PG_gid_field grupo
  Auth_PG_pwd_ignore_case on
  Auth_PG_encrypted on
  require group sap
  Allow from 132.248
</Directory>
```

Después de examinar un poco a detalle el archivo de configuración de Apache, lo que se debe de resaltar es:

- ✓ Todos los archivos y directorios que vayan a ser usados en el sistema deben de estar bajo el directorio /var/www/html/sap .
- ✓ Todos los programas escritos en PHP deben de tener la extensión php.

Por último, para levantar el servidor web, se ejecuta el siguiente comando:

```
[root@hades]# /etc/rc.d/init.d/httpd start
```

## 4.6 Creación de la base de datos

Después de haber instalado PostgreSQL, se puede levantar de la siguiente manera:

```
[root@hades]# /etc/rc.d/init.d/postgresql start
```

En la instalación del paquete `postgresql-7.0.2-17` se incluyen varios programas que ayudan a trabajar con las bases de datos en PostgreSQL, algunos de los más importantes son:

<code>/usr/bin/createdb</code>	Para crear una nueva base de datos. Sólo los usuarios de PostgreSQL pueden crear bases de datos.
<code>/usr/bin/createuser</code>	Para crear un usuario de PostgreSQL.
<code>/usr/bin/dropdb</code>	Para eliminar una base de datos existente. Sólo los usuarios de PostgreSQL pueden eliminar bases de datos.
<code>/usr/bin/dropuser</code>	Para eliminar un usuario de PostgreSQL.
<code>/usr/bin/pg_dump</code>	Para crear respaldos de una base de datos.
<code>/usr/bin/psql</code>	Es un cliente en modo texto que permite interactuar con un servidor PostgreSQL. A través de este cliente se puede mandar prácticamente cualquier consulta SQL al servidor.

Tabla 4.2 Programas útiles para trabajar con PostgreSQL.

La seguridad de las base de datos en PostgreSQL está enfocada en distintos niveles:

- ✓ Protección de los archivos de bases de datos. Todos los archivos que comprenden una base de datos son protegidos y sólo pueden ser leídos por el usuario `postgres`.
- ✓ Las conexiones de los clientes a un servidor PostgreSQL son, por default, permitidos sólo vía sockets Unix, no vía sockets TCP/IP.
- ✓ Las conexiones de los clientes pueden ser restringidas por dirección IP y/o por nombre.
- ✓ A cada usuario en PostgreSQL le es asignado un nombre de usuario y opcionalmente una contraseña. Por default los usuarios no tienen acceso de escritura a las bases de datos que ellos no han creado. Este nombre de usuario es usado también para restringir el acceso a las tablas de una base de datos.

La autenticación es el proceso mediante el cual el servidor PostgreSQL se asegura de que el usuario que está solicitando acceso a una base de datos es en efecto quien dice ser. Únicamente los usuarios que sean registrados como usuarios de PostgreSQL (comando `createuser`) tienen permitido conectarse y ejecutar operaciones sobre los objetos de una base de datos.

Los usuarios registrados dentro de un servidor PostgreSQL están lógicamente separados de los usuarios del sistema operativo en el cual se encuentra ejecutándose el servidor. EL comando `createuser` sólo afecta a los usuarios con respecto a PostgreSQL; éste no tiene efecto sobre los privilegios del usuario con respecto al sistema operativo.

Si los usuarios de un servidor PostgreSQL en particular también tienen cuentas de usuario en el sistema operativo, tiene mucho sentido asignar los mismos identificadores de usuario tanto a los usuarios del sistema operativo, como a los usuarios de PostgreSQL. Sin embargo, un servidor que acepta conexiones remotas puede tener muchos usuarios que no tengan una cuenta local, y en estos casos no debería de haber ninguna relación entre los usuarios de PostgreSQL y los usuarios del sistema.

Inicialmente, después de la instalación, el único usuario válido para PostgreSQL es el usuario `postgres`, por lo que cualquier operación que se tenga que hacer, se tendrá que hacer en principio como usuario `postgres`. Este en particular, es un usuario tanto de PostgreSQL, como del sistema.

Cuando alguna aplicación cliente desea conectarse a una base de datos, ésta debe de especificar algún nombre de usuario registrado dentro de PostgreSQL para que se pueda establecer la conexión de manera satisfactoria. Dentro del ambiente de PostgreSQL el nombre de usuario activo determina qué tipo de privilegios se otorgan sobre los objetos de una base de datos. Por lo tanto es muy importante restringir los nombres de usuario que pueden usarse en una aplicación cliente.

PostgreSQL le permite a un usuario crear cualquier número de bases de datos en un servidor dado siempre y cuando dicho usuario tenga los privilegios para hacerlo. El usuario que crea la base de datos se convierte automáticamente en el dueño y administrador de la misma. Por default, sólo el dueño de la base de datos puede hacer cualquier operación sobre ella. Si se desea que otros usuarios puedan acceder y manipular la base de datos, se deben de otorgar ciertos privilegios. Actualmente, hay cuatro diferentes tipos de privilegios: `select` (leer), `insert` (agregar), y `update/delete` (escribir). Para asignar estos privilegios se utiliza el comando `GRANT` de `sql`.

Antes de crear la cuenta del usuario de PostgreSQL que administrará la base de datos para el Sistema de Acopio Permanente, se va a dar de alta a este usuario también como usuario del sistema.

```
[root@hades]# adduser sap_usr
[root@hades]# passwd sap_usr
Changing password for user sap_usr
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
[root@hades]#
```

Ahora se dará de alta al usuario *sap\_usr* como un usuario de PostgreSQL.

```
[root@hades]# su - postgres
bash-2.04$
bash-2.04$ createuser sap_usr -P
Enter password for user "sap_usr":
Enter it again:
Shall the new user be allowed to create databases? (y/n) y
Shall the new user be allowed to create more new users? (y/n) y
CREATE USER
bash-2.04$ exit
logout
[root@hades]#
```

Entonces el usuario *sap\_usr* puede ya crear la base de datos con el comando `createdb`.

```
[sap_usr@hades]$ createdb sap
CREATE DATABASE
[sap_usr@hades]$
```

Ya que la base de datos está lista, lo que sigue a continuación es crear las tablas. La información de cada una de las tablas es mapeada de los requerimientos de información que fueron reflejados en los diagramas entidad-relación presentados en el capítulo 3.

Para el llenado de los datos en las tablas se está utilizando la siguiente notación:

- PK - Es una llave primaria.
- NN - Es un campo no nulo.
- FK - Es una llave foránea.
- U - Es un campo único.

Tabla: dependencias

CAMPO	TIPO DE LLAVE	NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto
tipo			Texto

Tabla: usuarios

CAMPO	TIPO DE LLAVE	NULO/UNICO	TIPO DE DATO
login	PK	NN, U	Texto
password		NN	Texto
grupo		NN	Texto
dependencia	FK	NN	Entero

- ✓ Tablas relacionadas al módulo Proyectos de investigación. Datos obtenidos del diagrama E-R presentado en la figura 3.3.

Tabla: proyectos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
lineainv			Texto
reponsable		NN	Texto
rfcresp			Texto
numtrab			Texto
camposciencia	FK2	NN	Entero
colaboradores			Texto
tipoact		NN	Texto
status		NN	Texto
terminado		NN	Booleano
alumnosbach			Entero
alumnoslic			Entero
alumnospost			Entero
comentarios			Texto

Tabla: fuentesfinanc

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: camposciencia

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: proy\_fuentefinanc

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
proyecto	PK, FK1	NN	Entero
fuentefinanc	PK, FK2	NN	Entero

- Tablas relacionadas al módulo Productos editoriales. Datos obtenidos del diagrama E-R presentado en la figura 3.4

Tabla: **prodcatalogos**

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
numero			Texto
volumen			Texto
issn			Texto

Tabla: **prodlibros**

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
edicion			Texto
coleccion			Texto
isbn			Texto



Tabla: prodmatapoyo

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
edicion			Texto
coleccion			Texto
isbn			Texto
receptor	FK2	NN	Entero
tipomat			Texto

Tabla: prodsoftware

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
version			Texto
lenguaje			Texto
tamano			Texto

Tabla: prodinformes

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
totalpag			Entero

Tabla: prodcarteles

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
tiraje			Texto
evento			Texto

Tabla: prodcartograficos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
escala1			Texto
escala2			Texto

Tabla: prodcuadernos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
edicion			Texto
totalpag			Texto
tiraje			Texto
coleccion			Texto

Tabla: prodmataudiovis

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
típomaterial	FK2	NN	Entero

Tabla: prodfolietos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
numero			Texto

Tabla: prodboletines

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
numero			Texto
volumen			Texto
fechatiraje			Fecha
periodicidad	FK2	NN	Entero

Tabla: prodmemorias

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
título		NN	Texto
autor			Texto
coautores			Texto
año			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
numero			Texto
volumen			Texto

Tabla: prodrevistas

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
título		NN	Texto
autor			Texto
coautores			Texto
año			Texto
editorial			Texto
ciudad			Texto
totalpag			Entero
tiraje			Texto
numero			Texto
volumen			Texto
arbitrada			Booleano
issn			Texto
periodicidad	FK2	NN	Entero

Tabla: receptores

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: tiposmaterial.

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: periodicidades

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

- Y Tablas relacionadas al módulo Publicaciones del personal. Datos obtenidos del diagrama E-R presentado en la figura 3.5.

Tabla: pubarticulosmem

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
memoria		NN	Texto
paginaini			Entero
paginafin			Entero
volumen			Texto
numero			Texto

Tabla: pubinformes

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
totalpag			Entero

Tabla: pubarticulosra

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
revista		NN	Texto
paginaini			Entero
paginafin			Entero
volumen			Texto
numero			Texto

Tabla: pubarticulosrna

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
revista			Texto
paginaini			Entero
paginafin			Entero
volumen			Texto
numero			Texto

Tabla: pubreportes

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
totalpag			Entero

Tabla: pubprologos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
libro		NN	Texto
paginaini			Entero
paginafin			Entero
editorial			Texto
coleccion			Texto
edicion			Texto
ciudad			Texto
tiraje			Texto
isbn			Texto

Tabla: pubcapituloslib

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
libro		NN	Texto
paginaini			Entero
paginafin			Entero
editorial			Texto
coleccion			Texto
edicion			Texto
ciudad			Texto
tiraje			Texto
isbn			Texto
editor			Texto

Tabla: pubintroducciones

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
publicacion		NN	Texto
paginaini			Entero
paginafin			Entero
editorial			Texto
coleccion			Texto
edicion			Texto
ciudad			Texto
tiraje			Texto
isbn			Texto

Tabla: publicos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
editorial			Texto
tiraje			Texto
totalpag			Entero
coleccion			Texto
edicion			Texto
ciudad			Texto
isbn			Texto



Tabla: pubmatapoyo

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
título		NN	Texto
autor			Texto
coautores			Texto
año			Texto
editorial			Texto
tiraje			Texto
totalpag			Entero
coleccion			Texto
edicion			Texto
isbn			Texto

Tabla: pubcatalogos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
título		NN	Texto
autor			Texto
coautores			Texto
año			Texto
editorial			Texto
tiraje			Texto
totalpag			Entero
ciudad			Texto
volumen			Texto
numero			Texto
issn			Texto

Tabla: pubarticulosperiod

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
autor			Texto
coautores			Texto
anio			Texto
periodico			Texto
paginaini			Entero
paginafin			Entero
epoca			Texto
numero			Texto
fecha			Fecha

- Y Tablas relacionadas al módulo Personal académico. Datos obtenidos del diagrama E-R presentado en la figura 3.7

Tabla: academicos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
rfc			Texto
tesislic			Entero
tesismae			Entero
tesisdoc			Entero
serviciosoc			Entero
estudio	FK2	NN	Entero
lugarest	FK3	NN	Entero
nivel	FK4	NN	Entero

Tabla: acad\_fuentefinanc

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
academico	PK, FK1	NN	Entero
fuentefinanc	PK, FK2	NN	Entero

Tabla: categorías

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: acad\_categoria

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
academico	PK, FK1	NN	Entero
categoria	PK, FK2	NN	Entero

Tabla: niveles

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: estudios

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: lugares

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

- Tablas relacionadas al módulo Actividades académicas. Datos obtenidos del diagrama E-R presentado en la figura 3.8

Tabla: acadseminarios

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadcongresos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadconferencias

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadsimposios

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadtalleres

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
sesiones			Entero
ponente			Texto

Tabla: acadcursos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
sesiones			Entero
ponente			Texto

Tabla: acadforos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadencuentros

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadcoloquios

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: acadconcursos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

- Tablas relacionadas al módulo Actividades de difusión. Datos obtenidos del diagrama E-R presentado en la figura 3.9

Tabla: difconciertos

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
funciones			Entero

Tabla: difmuestras

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
funciones			Entero

Tabla: difobrasd

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
funciones			Entero

Tabla: difobrasf

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
funciones			Entero

Tabla: difobrast

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha
funciones			Entero

Tabla: difvisitag

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha



Tabla: difexposiciones

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: differias

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: diffestivales

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: difjornadas

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: difmesasr

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: difperform

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fechaini			Fecha
fechafin			Fecha

Tabla: diflecturas

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fecha			Fecha
tipoelectura	FK2		Entero

Tabla: difprespublic

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
sede			Texto
recinto			Texto
asistentes			Entero
fecha			Fecha
tipopresentacion	FK2		Entero

Tabla: tiposlectura

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: tipospresentacion

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

- Tablas relacionadas al módulo Actividades de comunicación. Datos obtenidos del diagrama E-R presentado en la figura 3.10

Tabla: comcapsulas

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
duracion			Real
emisiones			Entero
medio			Texto
fechaini			Fecha
fechafin			Fecha
coproduccion			Booleano

Tabla: comprogramas

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
duracion			Real
emisiones			Entero
medio			Texto
fechaini			Fecha
fechafin			Fecha
coproduccion			Booleano

Tabla: compromocionales

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
titulo		NN	Texto
duracion			Real
emisiones			Entero
medio			Texto
fechaini			Fecha
fechafin			Fecha
coproduccion			Booleano

- Tablas relacionadas al módulo Actividades de educación continua. Datos obtenidos del diagrama E-R presentado en la figura 3.11

Tabla: acteducontinua

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
modulos			Entero
asistentes			Entero
constancias			Entero
ponentes			Entero
becados			Entero
sede			Texto
duracion			Texto
diplomado			Booleano
modalidad	FK2	NN	Entero
tipoactividad	FK3	NN	Entero
fechaini			Fecha
fechafin			Fecha
comentarios			Texto

Tabla: modalidades

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

Tabla: tiposactividad

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
nombre		NN	Texto

- Y Tablas relacionadas al módulo Premios y distinciones recibidos. Datos obtenidos del diagrama E-R presentado en la figura 3.12

Tabla: premiosrec

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
academico			Texto
rfcacad			Texto
institucion			Texto
fecha			Fecha

- Tablas relacionadas al módulo Premios y distinciones otorgados. Datos obtenidos del diagrama E-R presentado en la figura 3.13

Tabla: premiosotor

CAMPO	TIPO DE LLAVE	NO-NULO/UNICO	TIPO DE DATO
Id	PK	NN, U	Entero
dependencia	FK1	NN	Entero
nombre		NN	Texto
academico			Texto
rfcacad			Texto
institucion			Texto
fecha			Fecha

Con los datos de cada una de las tablas presentadas anteriormente se genera un archivo en el que se escriben las sentencias sql necesarias para crear las tablas en la base de datos.

El contenido completo del archivo *bd.sap.sql* utilizado para crear las tablas de la base de datos se muestra en el Apéndice A.

Una vez listo el archivo, se ejecuta el comando psql con la opción -f para indicarle a PostgreSQL que ejecute todas las sentencias SQL que se encuentran en el archivo *bd.sap.sql*

```
[sap_usr@hades]$ psql sap -f bd.sap.sql
[sap_usr@hades]$
```

Con esto queda lista la base de datos con todas las tablas y catálogos necesarios.

## 4.7 Programación del sistema

Una vez que ha quedado lista la base de datos, y que además se ha hecho ya el diseño de la interfaz del usuario, el paso siguiente es la creación de todos los archivos y programas encargados de hacer funcionar el sistema.

De acuerdo a la configuración del servidor web, todos los archivos que vayan a ser creados deben de estar bajo el directorio `/var/www/html/sap`, ya que éste se ha configurado para que sea un directorio protegido, y al cual únicamente tendrán acceso los usuarios válidos del sistema. Ver figura 4.3

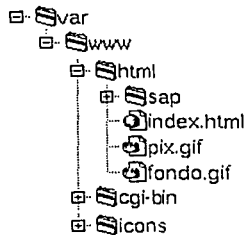


Figura 4.3 Estructura principal del directorio sap.

Como se observa en la figura anterior, se ha creado un archivo `index.html`, que es el que muestra la página de entrada al sistema, misma que se observa en la figura 4.4.

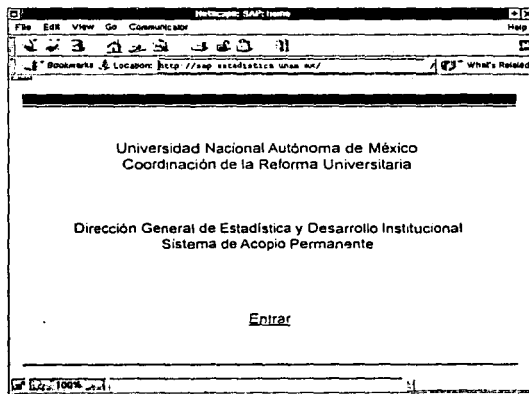


Figura 4.4 Entrada al sistema.

El código del archivo *index.html* se muestra a continuación:

```
<html>
<head>
<title>SAP: home</title>
</head>

<body link="#006699" background="fondo.gif" text="#020167">
<table cellpadding="0" cellspacing="0" width="97%" align="center">
<tr>
  <td bgcolor="#E6E6E6">
    
  </td>
</tr>
<tr>
  <td bgcolor="#006699">
    
  </td>
</tr>
<tr bgcolor="#ffffff">
  <td>
    <font face="Tahoma, helvetica, arial" size="+2">
    <br><br>
    <div align=center>
      Universidad Nacional Autónoma de México
    <br>
      Coordinación de la Reforma Universitaria
    <br>
    <br>
    <br>
      Dirección General de Estadística y Desarrollo Institucional
    <br>
      Sistema de Acopio Permanente
    <br>
    <br>
    <br>
    <br>
    <br>
    <b><a href="sap/index.php">Entrar</a></b>
    <br>
    <br>
    <br>
    <br>
  </td>
</tr>
<tr>
  <td bgcolor="#006699">
    
  </td>
</tr>
</table>

</body>
</html>
```



Cuando se selecciona la opción "Entrar", se muestra una ventana en la que los usuarios deben de proporcionar su login y su password para que puedan acceder al sistema. Ver figura 4.5

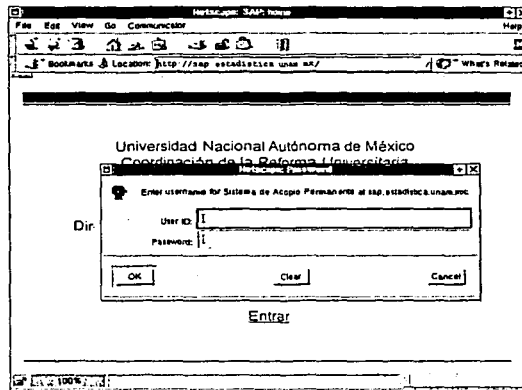


Figura 4.5 Validación de usuarios.

En el caso de que los datos proporcionados por el usuario sean válidos, se muestra una nueva ventana en la que se presentan los principales módulos en los cuales el usuario puede realizar el proceso de acopio de información. Ver figura 4.6

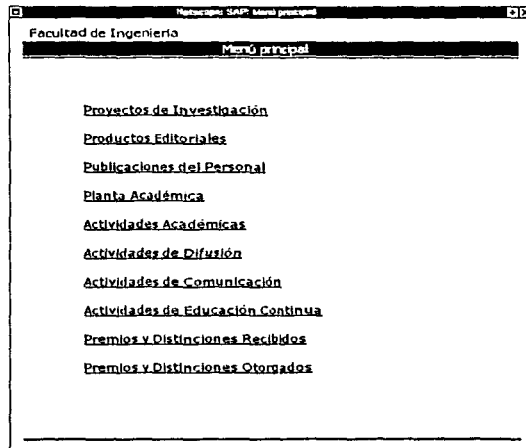


Figura 4.6 Menú principal.

El primer programa al que acceden los usuarios, es el del archivo *index.php*. La funcionalidad de este programa es básicamente abrir una nueva ventana de navegación en la que se omiten la barra de menú, la barra de herramientas y la barra de direcciones del navegador, y se muestre exclusivamente información relacionada con el sistema, información tal como el nombre de la dependencia a la que pertenece el usuario, y el módulo en el que se encuentra trabajando.

Es decir, se sustituye una barra de navegación como la mostrada en la figura 4.7

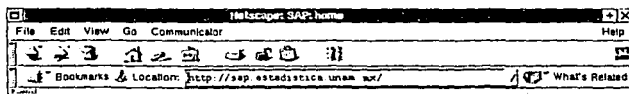


Figura 4.7 Barra de navegación original.

por una más sencilla y útil, como la mostrada en la figura 4.8



Figura 4.8 Barra de navegación simplificada.

Esta idea forma parte del diseño de la interfaz del sistema, y tiene como principal objetivo simplificar el número de opciones que el usuario tiene presentes cuando está usando el sistema, además de forzarlo a utilizar la navegación que se ha planteado para ser usada dentro del sistema, o sea, el usuario no podrá usar las opciones típicas de un navegador web, "hacia delante", "hacia atrás", "recargar página", etc.

El código del archivo *index.php* es el siguiente:

```
<html>
  <head>
    <script language="javascript1.2" src="./include/header.js">
    </script>
    <title>SAP: Inicio</title>
  </head>
  <body onLoad="abreVentana('menu.php', 'menu', 750, 510);
  location.href='../index.html'">
  </body>
</html>
```

Como se mencionó anteriormente, este programa abre una nueva ventana, y en ella se presenta en principio la información sobre los módulos del sistema, información que se guarda en el archivo *menu.php*.

El programa *menu.php* se muestra a continuación:

```
<?
$encabezado = "Menú principal";
$tit_pagina = "SAP: Menú principal";
include("../include/pg.inc");
include("../include/header.inc");
?>

<table align="center" width="70%" border="0">
<tr>
<td>
<font face="Tahoma, helvetica, arial" size="+1">
<br><br>
<a href="proyectos/index.php">Proyectos de Investigación</a>
<br>
<br>
<a href="proeditoriales/index.php">Productos Editoriales</a>
<br>
<br>
<a href="pubpersonal/index.php">Publicaciones del Personal</a>
<br>
<br>
<a href="personal/index.php">Personal Académico</a>
<br>
<br>
<a href="actacademicas/index.php">Actividades Académicas</a>
<br>
<br>
<a href="actdifusion/index.php">Actividades de Difusión</a>
<br>
<br>
<a href="actcomunicacion/index.php">Actividades de Comunicación</a>
<br>
<br>
<a href="acteducontinua/index.php">Actividades de Educación Continua</a>
<br>
<br>
<br>
<b><a href="premiosrec/index.php">Premios y Distinciones Recibidos</a>
<br>
<br>
<b><a href="premiosotor/index.php">Premios y Distinciones Otorgados</a>
<br>
<br>
<br>
</td>
</tr>
</table>

<? include ("../include/footer.inc"); ?>
```

Los archivos bajo el directorio *sap* se han organizado en otros subdirectorios, uno para cada módulo del sistema. Ver figura 4.9

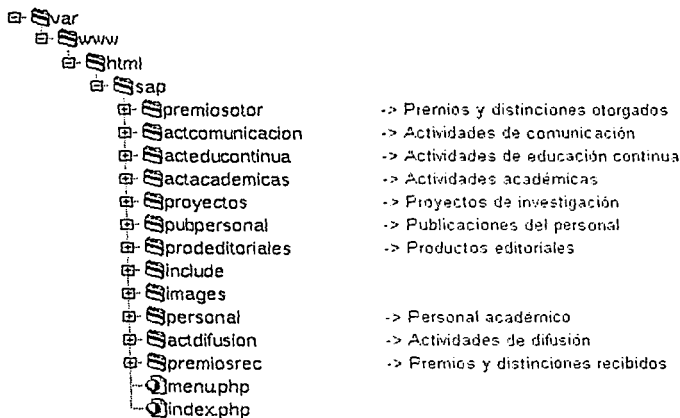


Figura 4.9 Estructura principal de directorios.

Adicionalmente se han creado dos subdirectorios, uno llamado *include*, y otro llamado *images*, en ambos directorios se guardan archivos que son comunes a todos los módulos, es decir, archivos que son útiles para todos los módulos del sistema, la diferencia es que en el segundo de ellos se guardan exclusivamente archivos gráficos, imágenes.

Para lograr mantener una uniformidad en el estilo, el diseño, y la presentación de cada una de la pantallas que serán presentadas al usuario, se ha planteado un esquema de creación de páginas, en el que todos los archivos deben de tener el mismo encabezado (*header.inc*), y el mismo pie (*footer.inc*), cada uno con su respectivo contenido. Ver figura 4.10

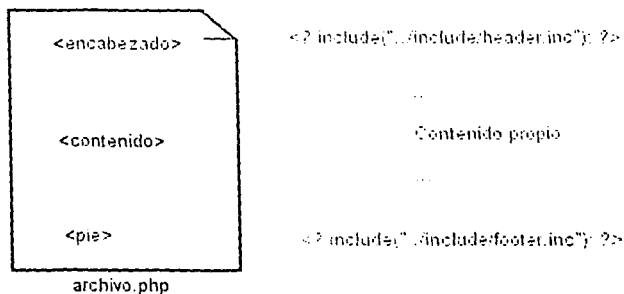


Figura 4.10 Esquema de creación de páginas.

Ambos archivos, *header.inc* y *footer.inc* se encuentran bajo el directorio *include*. Ver figura 4.11

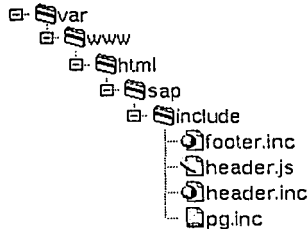


Figura 4.11 Contenido del directorio include.

El contenido del archivo *header.inc* se presenta a continuación:

```

<html>
<head>
<script language="javascript" src="/sap/include/header.js">
</script>
<title><?php print "$tit_pagina"; ?></title>
</head>

<body link="#006699" background="/sap/images/fondo.gif" text="#020167">
<br>
<table cellpadding="0" cellspacing="0" width="97%" align="center">
<tr>
<td align="left">
<font face="Tahoma, helvetica, arial" color="#020167">
<?php print "$NOMBRE_DEPENDENCIA"; ?>
</font>
</td>
</tr>
<tr>
<td bgcolor="#006699" align="center">
<font face="Tahoma, helvetica, arial" size="+1" color="#ffffff">
<?php print "$encabezado"; ?>
</font>
</td>
</tr>
<tr>
<td bgcolor="#E6E6E6">

</td>
</tr>
<tr>
<td bgcolor="#006699">

</td>
</tr>
<tr bgcolor="#ffffff">
<td align="center">

```

El contenido del archivo *footer.inc* se muestra a continuación:

```

    </td>
</tr>
<tr>
  <td bgcolor="#006699">
    
  </td>
</tr>
</table>
</body>
</html>

```

Adicionalmente, la mayoría de los programas realizados necesitan establecer una conexión a la base de datos, y para ello se han definido en el archivo *pg.inc* los parámetros necesarios.

El contenido del archivo *pg.inc* se muestra a continuación:

```

<?
$CONN = pg_Connect("dbname=sap user=sap_usr password=sap_pwd");
$query = "SELECT dependencia FROM usuarios WHERE login = '$REMOTE_USER'";
$res = pg_Exec($CONN,$query);
$DEP = pg_Result($res, 0, "dependencia");
$query = "SELECT nombre FROM dependencias WHERE id=$DEP";
$rs=pg_exec($CONN,$query);
$NOMBRE_DEPENDENCIA = pg_result($rs,0, "nombre");

/* Las variables $CONN, $DEP y $NOMBRE_DEPENDENCIA quedan disponibles para
todos los programas que incorporen este archivo */
?>

```

Para ilustrar el funcionamiento general del proceso de altas, bajas y cambios de registros en el Sistema de Acopio Permanente, que en todos los módulos es muy similar, y para mostrar el uso de los archivos *header.inc*, *footer.inc* y *pg.inc* se presenta a continuación el código de los programas en el módulo de Actividades de Educación Continua. Ver figura 4.12

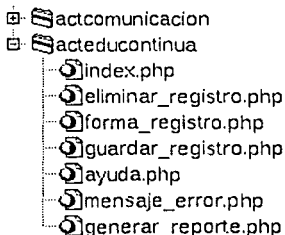


Figura 4.12 Archivos del módulo de Actividades de Educación Continua.

Primer paso, se presentan los principales módulos. Ver figura 4.13

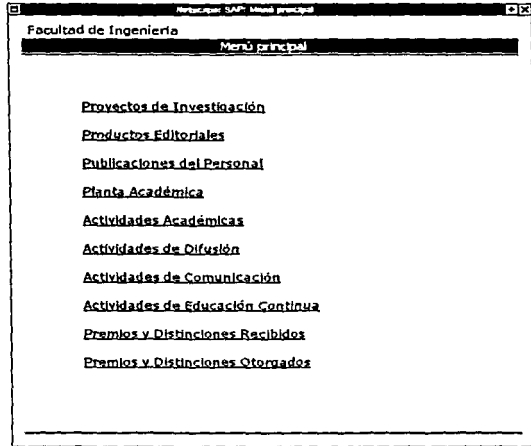


Figura 4.13 Principales módulos del sistema.

Al seleccionar un módulo, se presenta una lista de los registros que se tienen actualmente capturados. En este caso, la lista de registros corresponde al módulo de Actividades de Educación Continua. Esta lista es generada por el archivo *acteducontinua/index.php*.

<?

```
/* Se incorporan los archivos header.inc y pg.inc */

include ("../include/pg.inc");
$encabezado = "Actividades de Educación Continua";
$tit_pagina = "SAP: Actividades de Educación Continua";
include ("../include/header.inc");

$res = pg_Exec($CONN, "SELECT id,nombre FROM acteducontinua WHERE
dependencia = 'SDEP' ORDER BY nombre");
$total_regs = pg_numrows($res);
if($total_regs > 0){
    if($total_regs < 300)
        $ligas_por_pag = 10;
    else
        $ligas_por_pag = 15;
    if($total_regs*$ligas_por_pag == 0)
        $ligas = $total_regs / $ligas_por_pag;
    else
    {
        $ligas = (int)($total_regs/$ligas_por_pag);
        $ligas ++;
    }
}
```

```

if($pagina){
  $lim_inf = ($pagina-1)*$ligas_por_pag;
  if($ligas == $pagina) $lim_sup = $total_regs;
  else $lim_sup = $pagina*$ligas_por_pag;
}
else{
  if($ligas_por_pag < $total_regs ){
    $pagina=1;
    $lim_inf=0;
    $lim_sup=$ligas_por_pag;
  }
  else if($total_regs==0)
  {
    $pagina=1;
    $lim_inf=0;
    $lim_sup=0;
  }
  else{
    $pagina=1;
    $lim_inf=0;
    $lim_sup=$total_regs;
    $ligas=1;
  }
}

if($lim_inf != $lim_sup)
{
  print "
  <br><br>
  <table border='0' align='center' width='95%' cellpadding='10'>
  <tr>
  <td align='left' valign='top'>
  <a href=\"javascript:
  abreVentana('ayuda_educontinua.php', 'ayudaEducontinua', 700, 450)\
  onMouseOver=\"ayuda.src='/sap/images/ayuda2.gif'\
  onMouseOut=\"ayuda.src='/sap/images/ayuda.gif'\
  <img border=0 src=\"/sap/images/ayuda.gif\" alt=\"AYUDA\"
  name=\"ayuda\"></a>
  </td>
  <td align='right'>
  <font face='Tahoma, helvetica, arial' size='+1'>Total registros:
  <b>$total_regs </b></font></font><br><br></td></tr>";
  for ($i=$lim_inf; $i<$lim_sup; $i++){
    $id= pg_Result($res,$i,'id');
    $titulo = pg_Result($res,$i,'nombre');

    if(strlen($titulo) >= 85) $titulo = substr($titulo,0,85)."...";

    $num = $i + 1;
    print "
    <tr>
    <td width='5%' align='right' valign='top'>
    <font face='Tahoma, helvetica, arial' size='+1'>$num.</font>
    </td>
    <td valign='top'>
    <font face='Tahoma, helvetica, arial' size='+1'>

```



```

        <a href='forma_registro.php?id=$id&pagina=$pagina'>$titulo</a>
      </font>
    </td>
  </tr>";
}

print "
<tr>
  <td align='center' colspan='2'>
    <font face='Tahoma, helvetica, arial'>
      <br><br>";
    for ($j=1;$j<=$ligas;$j++){
      if ($j==$pagina)print " <font color='#006699'><b>$j</b></font>";
      else print " &nbsp;  <a href='index.php?pagina=$j'>$j</a>";
    }
    print "
  </td>
</tr>
</table>";
}
}
else{
  print "
  <br><br>
  <font face='Tahoma, helvetica, arial'>
  <div align='center'>No se encontró ningún registro.</div>";
}
?>

<table align='center' width='70%' border='0'>
<tr>
<td align='center'>
<font face='Tahoma, helvetica, arial' size='-1'>
<br><br>
<form method='post' name='forma'>
<input type='hidden' name='pagina' value='<? print $pagina; ?>'>
<b>
<input type='button' value="Nuevo registro"
onclick="javascript:document.forma.action='forma_registro.php';
document.forma.submit()" ">
&nbsp;  
<input type='button' value="Generar Reporte "
onclick="javascript:document.forma.action='generar_reporte.php';
document.forma.submit()" ">
&nbsp;  
<input type='button' value=' Menú principal '
onClick="location.href='../menu.php'">
<br>
</b>
</td>
</tr>
</table>
<br>

/* Se incorpora el archivo footer.inc */
<? include ("../include/footer.inc"); ?>

```

La página resultante es muy similar a la presentada en la figura 4.14

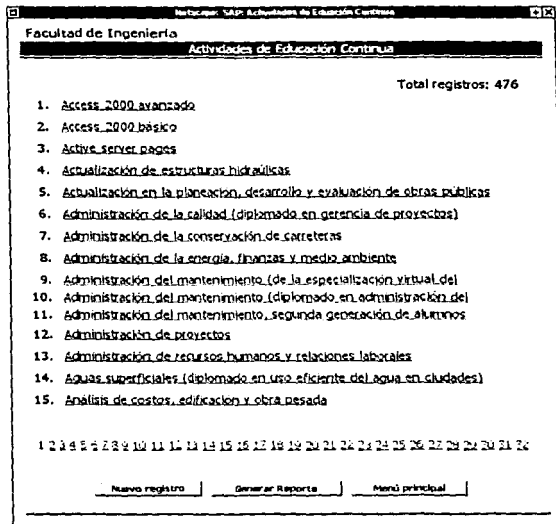


Figura 4.14 Lista de registros en Actividades de Educación Continua.

Cuando se selecciona algún registro de esta lista, se pasa al programa *acteducontinua/forma\_registro.php* que se encarga de mostrar todos los datos que se tengan de dicho registro, además de dar la opción de actualizarlo o eliminarlo. En el caso de seleccionar la opción "Nuevo registro", igualmente se pasa al programa *acteducontinua/forma\_registro.php*, pero en este caso se muestra una forma en blanco para que el usuario capture los datos del nuevo registro.

El código del archivo *acteducontinua/forma\_registro.php* se muestra a continuación.

```

<?
/* Se incorporan los archivos header.inc y pg.inc */
include ("../include/pg.inc");
$tit_pagina = "SAP: Actividades de Educación Continua";
$encabezado = "Actividades de Educación Continua";
include ("../include/header.inc");

function imprimeSelect($nombre,$sarr_mod,$selected){
    print "<select name=$nombre>";
    while( list($index,$valor) = each($sarr_mod) ){
        if($index == $selected) print "<option value='$index' selected>";
        else print "<option value='$index'>";
    }
    print " $valor </option>";
}
print "</select> \n";
}

```

```
$res = pg_Exec($CONN, "SELECT * FROM modalidades");
$filas = pg_NumRows($res);

for($i=0; $i < $filas; $i++)
{
    $sid_mod = pg_Result($res,$i, "id");
    $nombre_mod = pg_Result($res,$i, "nombre");

    $modalidades[$i] = $nombre_mod;
}

$res = pg_Exec($CONN, "SELECT * FROM tiposactividad");
$filas = pg_NumRows($res);

for($i=0; $i < $filas; $i++)
{
    $sid_act = pg_Result($res,$i, "id");
    $nombre_act = pg_Result($res,$i, "nombre");

    $tipos_act[$i] = $nombre_act;
}

if($sid){
$res = pg_Exec($CONN, " SELECT * FROM acteducontinua WHERE id=$sid; ");

$sid = pg_Result($res,0,'id');
$dependencia = pg_Result($res,0,'dependencia');
$nombre = pg_Result($res,0,'nombre');
$modalidad = pg_Result($res,0,'modalidad');
$tipo_act = pg_Result($res,0,'tipo_act');

$num_ponentes= pg_Result($res,0,'ponentes');
    if($num_ponentes==0) unset($ponentes);
$num_modulos= pg_Result($res,0,'modulos');
    if($num_modulos==0) unset($modulos);
$num_asistentes= pg_Result($res,0,'asistentes');
    if($num_asistentes==0) unset($asistentes);
$num_constancias= pg_Result($res,0,'constancias');
    if($num_constancias==0) unset($constancias);
$num_becados= pg_Result($res,0,'becados');
    if($num_becados==0) unset($becados);

$sede = pg_Result($res,0,'sede');
$duracion= pg_Result($res,0,'duracion');

$fecha_inicio=explode("/",pg_result($res,0,fechalni));
$fecha_termino=explode("/",pg_result($res,0,fechafin));

$diplomado = pg_Result($res,0,'diplomado');

$comentarios = pg_Result($res,0,'comentarios');

$sarr_comentarios = split ("\\", $comentarios);
```

```

if(sizeof($arr_comentarios) > 0 ){
    $comentarios='';
    for($i=0;$i < sizeof($arr_comentarios); $i++){
        $comentarios .= "\n".$arr_comentarios[$i];
    }
}

$nuevo_registro = FALSE;
}
else{
    $res = pg_Exec($CONN, "select nextval('seq_acteducontinua') as id");
    $id = pg_Result($res,0,'id');
    $nuevo_registro = TRUE;
}

print "

<br>

<form action='guardar_registro.php' method=get name=forma>
<table border=0 align=center width=90% cellpadding=5 cellspacing=10>

<input type=hidden name=id value='$id'>
<input type=hidden name=pagina value='$pagina'>
<input type=hidden name=nuevo_registro value=$nuevo_registro>

<tr><td align=right valign=top>
<font face='Tahoma,Helvetica,Arial'>Nombre:</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=35 name=nombre value='$nombre' >
</td></tr>

<tr><td align=right valign=top>
<font face='Tahoma,Helvetica,Arial'>Modalidad:</td>
<td>
<table border=0 cellpadding=0 cellspacing=0>
<tr>
<td valign=top>
<font face='Tahoma,Helvetica,Arial' size=-1> ";
    imprimeSelect('modalidad',$modalidades,"$modalidad");
print "
</td>
<td>
<table border=0>
<tr align=right>
<td>
<font face='Tahoma,Helvetica,Arial'>
No. ponentes:
</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=2 name=num_ponentes value='$num_ponentes' >
</td>
</tr>

```

```

<tr align=right>
<td>
<font face='Tahoma,Helvetica,Arial'>
No. módulos:
</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=2 name=num_modulos value='$num_modulos' >
</td>
</tr>
<tr align=right>
<td>
<font face='Tahoma,Helvetica,Arial'>
No. asistentes:
</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=2 name=num_asistentes value='$num_asistentes' >
</td>
</tr>
<tr>
<td align=right>
<font face='Tahoma,Helvetica,Arial'>
No. constancias ó diplomas:
</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=2 name=num_constancias value='$num_constancias' >
</td>
</tr>
<tr align=right>
<td>
<font face='Tahoma,Helvetica,Arial'>
No. becados:
</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=2 name=num_becados value='$num_becados' >
</td>
</tr>
</table>
</td></tr></table>
</td></tr>

<tr><td align=right valign=top>
<font face='Tahoma,Helvetica,Arial'>Sede:</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=35 name=sede value='$sede' >
</td></tr>

<tr><td align=right valign=top>
<font face='Tahoma,Helvetica,Arial'>Duraci&oacute;n:</td>
<td>
<font face='Tahoma,Helvetica,Arial'>
<input type=text size=20 name=duracion value='$duracion' >
</td></tr>";

```

```

print "
<tr>
<td align=right valign=top>
<font face='Tahoma,Arial,Helvetica'>Fecha de Inicio:</td>
<td>
<font face='Tahoma,Arial,Helvetica'>
<input type=text name=fecha_iniciodia maxlength=2 value='$fecha_inicio[0]'
size=2> &nbsp;&nbsp;&nbsp;
<select name=fecha_iniciomes> ";
$meses=array(0 => '----',1 => 'Enero',2 => 'Febrero',
3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio',
7 => 'Julio', 8 => 'Agosto', 9 => 'Septiembre',
10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre');

for($i=0;$i<=12;$i++){
if($fecha_inicio[1] == $i)
print "<option value=$i selected>$meses[$i] " ;
else
print "<option value=$i>$meses[$i] " ;
}
print "</select>
Año
<input type=text size=4 name=fecha_inicioanio value='$fecha_inicio[2]'>
</td>
</tr>

<tr>
<td align=right valign=top>
<font face='Tahoma,Arial,Helvetica'>Fecha de Término:</td>
<td>
<font face='Tahoma,Arial,Helvetica'>
<input type=text name=fecha_terminodia maxlength=2
value=\"\$fecha_termino[0]\" size=2> &nbsp;&nbsp;&nbsp;
<select name=fecha_terminomes>");
$meses=array(0 => '----',1 => 'Enero',2 => 'Febrero',
3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio',
7 => 'Julio', 8 => 'Agosto', 9 => 'Septiembre',
10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre');
for($i=0;$i<=12;$i++){
if($fecha_termino[1] == $i)
print("<option value=$i selected>$meses[$i]");
else
print("<option value=$i>$meses[$i]");
}
print "</select> Año
<input type=text size=4 name=fecha_terminoanio value='$fecha_termino[2]'>
</td>
</tr>

<tr><td align=right valign=top>
<font face='Tahoma,Helvetica,Arial'>
Tipo:
</td>
<td>

```



```

</b>
</div>
</font>
</td></tr>

</table>
</form>
";

/* Se incorpora del archivo footer.inc */
include ("../include/Footer.inc") ?>

```

La página que se muestra cuando un registro es seleccionado es muy similar a la presentada en la figura 4.15.

Nombre:

Modalidad:

No. ponentes:

No. módulos:

No. asistentes:

No. constancias o diplomas:

No. becados:

Sede:

Duración:

Fecha de Inicio:   Año

Fecha de Término:   Año

Tipo:

Diplomado:

Comentarios:

Figura 4.15 Datos de un registro en Actividades de Educación Continua.

En este caso se tiene la opción de modificar el registro y posteriormente guardar los cambios, en cuyo caso se pasa al programa `acteducontinua/guardar_registro.php`. Si en lugar de ello se decide eliminar el registro, se pasa al programa `acteducontinua/eliminar_registro.php`. Si se elige la opción "Cancelar", entonces se regresa a la página de la lista de registros.

Como se mencionó anteriormente, el programa `acteducontinua/guardar_registro.php` tiene dos funcionalidades básicas, la primera es actualizar los datos de un registro, y la otra, ingresar los datos de un nuevo registro. Dentro del programa se tiene el código que determina qué tipo de operación es la que se debe de realizar.



La página que se muestra cuando se selecciona la opción "Nuevo registro" es muy similar a la presentada en la figura 4.16.

Facultad de Ingeniería  
Actividades de Educación Continua

Nombre: \_\_\_\_\_

Modalidad:  No. ponentes:   
 No. módulos:   
 No. asistentes:   
 No. constancias o diplomas:   
 No. becados:

Sede: \_\_\_\_\_

Duración: \_\_\_\_\_

Fecha de Inicio:  Año

Fecha de Término:  Año

Tipo:  Diplomado: Si  No

Comentarios:

Figura 4.16 Forma de captura de un nuevo registro.

En este caso se tiene la opción de guardar los datos para el nuevo registro, en cuyo caso se pasa al programa *acteducontinua/guardar\_registro.php*. Si se elige la opción "Cancelar", entonces se regresa a la página de la lista de registros.

El código del archivo *acteducontinua/guardar\_registro.php* se muestra a continuación.

```
<?
/* Incorporación del archivo pg.inc */

include ("../include/pg.inc") ;

if (!$num_modulos) $num_modulos=0;
if (!$num_asistentes) $num_asistentes=0;
if (!$num_constancias) $num_constancias=0;
if (!$num_ponentes) $num_ponentes=0;
if (!$num_becados) $num_becados=0;

$nombre = ereg_replace ("\\.\\\"", "", $nombre);
$sede = ereg_replace ("\\.\\\"", "", $sede);
$duracion = ereg_replace ("\\.\\\"", "", $duracion);

$comentarios = ereg_replace ("\\.\\\"", "", $comentarios);
$comentarios = preg_replace ("/.(\n)+/", "|", $comentarios);
```

```

        if (($fecha_iniciodia) && ($fecha_inicioanio))
        {$fechainicio="$fecha_iniciodia/$fecha_iniciomes/$fecha_inicioanio";}
        else { $fechainicio=""; }

        if (($fecha_terminodia) && ($fecha_terminoanio))
        {$fechatermino="$fecha_terminodia/$fecha_terminomes/$fecha_terminoanio";}
        else { $fechatermino=""; }

if (!$nuevo_registro) {
    $query = "update acteducontinua set
    nombre='$nombre',modalidad='$modalidad',";
    $query.= "ponentes=$num_ponentes,modulos=$num_modulos,
    asistentes=$num_asistentes,";
    $query.= "constancias=$num_constancias,becados=$num_becados,";
    $query.= "sede='$sede',duracion='$duracion',";
    $query.= "fechaini='$fechainicio', fechafin='$fechatermino',";
    $query.= "tipo_act='$tipo_act', diplomado='$diplomado',
    comentarios='$comentarios'";
    $query.=" where id=$id";
}
else{

    $query = "insert into acteducontinua
    values($id,'$DEF','$nombre','$modalidad',$num_modulos,";
    $query.= "$num_asistentes, $num_constancias,
    $sede','$duracion','$fechainicio',";
    $query.= "'$fechatermino','$diplomado','$comentarios', ";
    $query.= "$num_ponentes, $num_becados,'$tipo_act') ";
}

$res = @pg_Exec($CONN, "$query");

$msg = "A ocurrido un error en el archivo ".__FILE__.";
$msg = "en la línea ".__LINE__;

$msg = "$msg <br><br>". pg_ErrorMessage($CONN);
$msg = urlencode($msg);

if($res){
    print header("location: index.php?pagina=$pagina");
}
else
    print header("location: mensaje_error.php?error=$msg");

?>

```

Si se pudo efectuar de manera satisfactoria la incorporación del nuevo registro, o en su caso la actualización del registro, se pasa de nuevo a la lista de registros, en caso contrario se muestra un mensaje de error con el programa *acteducontinua/mensaje\_error.php*.

En el caso de que se decida eliminar el registro, entonces el programa que debe de ejecutarse es *acteducontinua/eliminar\_registro.php*, cuyo código se presenta a continuación.

```

<?
/* Se incorpora del archivo pg.inc */ .
include ("../include/pg.inc") ;
if(!$nuevo_registro){
    $res = pg_Exec($CONN, " DELETE FROM acteducontinua WHERE id = $id ");
    if($res)
        header("location: index.php?pagina=$pagina");
    else{
        $mesg = "A ocurrido un error en el archivo ".__FILE__." en la
linea ".__LINE__;
        $mesg = "$mesg <br><br>".pg_ErrorMessage($CONN);
        $mesg = urlencode($mesg);

        header("location: mensaje_error.php?error=$mesg");
    }
}
else{
    header("location: index.php?pagina=$pagina");
}
?>

```

En el caso de que suceda algún error cuando se está haciendo una operación sobre la base de datos, se hace una llamada al programa *acteducontinua/mensaje\_error.php*, cuyo código se muestra a continuación.

```

<?
/* Se incorporan los archivos header.inc y pg.inc */
$encabezado = "";
$tit_pagina = "SAP: Error";
include ("../include/pg.inc");
include ("../include/header.inc");
?>

<font face="Tahoma,Helvetica,Arial" size="+2">
<br><br>
<div align=center><font color=red> Atención </font>
<br><br>
    <?
        print "$error <br><br><br>";
        if($link)
            print "<b><a href=\"\$link\">Regresar</a>";
        else
            print "<b><a href=\"index.php\">Intentar de nuevo</a>";
    ?>
</div>

/* Se incorpora el archivo footer.inc */
<? include ("../include/footer.inc"); ?>

```

## 4.8 Pruebas y mantenimiento

Generalmente se suelen invertir muchos recursos en el mantenimiento de los sistemas, pues éstos deben ajustarse continuamente a los cambios en una institución, o mejorarse para obtener resultados óptimos.

Los costos de mantenimiento excesivos se reflejan directamente en el diseñador del sistema, debido a que aproximadamente 70 por ciento de los errores de software se pueden atribuir a un diseño del sistema inadecuado. Desde una perspectiva del sistema, tiene sentido que la detección y corrección de los errores del diseño tempranamente es menos costosa que el dejar que los errores permanezcan sin aparecer hasta que sea necesario el mantenimiento.

El mantenimiento de un sistema es sin duda un consideración importante, es por ello que se debe de tener el mayor cuidado posible durante todas las etapas de desarrollo del mismo.

### 4.8.1 Pruebas

Durante la etapa de desarrollo del Sistema de Acopio Permanente, frecuentemente se fueron realizando pruebas de distinto tipo para evitar que el sistema tuviera el menor número de problemas. Por ejemplo, se realizaron pruebas para asegurar que los campos requeridos en algún registro fueran correctamente llenados por parte de los usuarios, que los datos que fueran capturados a través de los formularios llegaran correctamente a los programas que los procesaban, y que los datos que tenían que ser mostrados en las pantallas fueron claros e informativos. Este trabajo fue hecho constantemente durante la programación de cada módulo del sistema.

Después de haber terminado la programación de todos los módulos, se realizaron pruebas tratando de simular el funcionamiento real del sistema. Se crearon usuarios de prueba y con éstos se verificó que cada módulo y submódulo del sistema trabajaba de manera correcta al realizar las principales actividades del sistema; la alta de registros, la actualización de registros, la eliminación de registros y la generación de reportes.

Parte del trabajo del analista de sistemas es asegurar que haya canales y procedimientos adecuados en su lugar para permitir la retroalimentación y la posterior respuesta a las necesidades de mantenimiento. Los usuarios y operadores deben ser capaces de comunicar problemas y sugerencias fácilmente a aquellos que darán mantenimiento al sistema.

Cuando se terminaron de realizar todas las pruebas necesarias y se aseguró que el sistema podía funcionar sin ningún inconveniente, éste fue liberado. Se informó a los usuarios el nombre de usuario y la contraseña que les fue asignado, y se les hizo saber que podían hacer ya uso del sistema. Se les informó también que si detectaban algún error o tenían algún comentario sobre el funcionamiento del sistema, nos lo hicieran saber, ya sea mediante una llamada telefónica, ó mediante correo electrónico.

## 4.8.2 Mantenimiento

Después de que el sistema estuvo operando realmente, dio inicio el mantenimiento del mismo. Los tres tipos de mantenimiento que se contemplaron fueron:

✓ **Mantenimiento correctivo.** Independientemente de cuán bien diseñado, desarrollado y probado está un sistema, los errores pueden ocurrir en cualquier momento de manera inevitable. El mantenimiento correctivo se relaciona con la solución o la corrección de errores de programación no detectados anteriormente. ✓

Las primeras impresiones de los usuarios después de hacer uso del sistema por unos días fue en la mayoría de los casos bastante satisfactoria, hubo muy buenos comentarios acerca del mismo, sin embargo hubo también uno que otro comentario referente a pequeños detalles que se detectaron y que sugerían la corrección de determinados módulos del sistema, éstos afortunadamente fueron muy pocos, y en gran medida enfocados a "pulir" el sistema.

Los cambios sugeridos y en su caso necesarios fueron hechos en su momento, posteriormente no se hicieron mayores cambios ni modificaciones.

✓ **Mantenimiento perfecto.** El mantenimiento perfecto trata de mejorar las características del sistema, como hacerlo más rápido, más óptimo, como mejorar opciones de ayuda, etcétera. ✓

En este caso, lo único que se realizó, y que en su momento no estaba contemplado, fue crear algunas opciones de ayuda, pues resultó que había usuarios que era la primera vez que usaban el sistema, y tenían algunas dudas sobre el tipo de información que debían de capturar. En esta ayuda, se describen brevemente a qué se refieren cada uno de los campos de datos en un registro, cuáles son los valores válidos permitidos, y en algunos casos, cuál es el formato adecuado de captura de datos.

✓ **Mantenimiento preventivo.** Este tipo de mantenimiento es uno de los más eficaces en función de los costos, ya que si se realiza de manera oportuna y adecuada, puede evitar serios problemas en el sistema.

El principal trabajo en este tipo de mantenimiento consistió en realizar respaldos continuos de la base de datos. Debido a la importancia de la información que se recaba, es indispensable contar con datos fiables y además respaldados. ✓

Una parte en la que se tuvo especial cuidado, consistió en verificar que los respaldos obtenidos de las bases de datos fueran efectivamente útiles, ya que había sucedido en alguna ocasión que el respaldo de una base de datos no se pudo recuperar.

Como se mencionó anteriormente, el trabajo principal relacionado con el respaldo de la información estuvo enfocado a la base de datos, sin embargo, aunque con menos frecuencia, también se respaldó el código fuente de la aplicación, que incluye todos los archivos y programas necesarios para que el sistema pueda funcionar.

## CONCLUSIONES

## Conclusiones

El desarrollo de un sistema de información se caracteriza porque cuenta con una serie de pasos o etapas que se tienen que seguir de forma sistemática. Es por ello que al elegir la metodología denominada "Método de ciclo de vida del desarrollo de sistemas" se pudo cumplir con los objetivos planteados de forma efectiva, ya que en esta metodología las etapas están ya bien definidas y cada una de ellas cumplió un papel trascendente durante el desarrollo del sistema.

La etapa de planificación consistió básicamente en la asignación de las tareas apropiadas a cada uno de los miembros del equipo de desarrollo. Se designó a las personas encargadas de realizar el análisis y diseño de la base de datos, a las personas encargadas de crear la interfaz gráfica para el usuario y a las personas encargadas de la programación del sistema, entre otras tareas. Esta etapa permitió estudiar también cuáles eran los principales factores que influían en el desarrollo del sistema, y se pudo determinar que se podían alcanzar los objetivos deseados de manera oportuna.

En la etapa de análisis el trabajo principal fue estudiar el sistema de acopio de información que se tenía actualmente para conocer y entender su funcionamiento, y establecer entonces dónde era necesario efectuar mejoras. Este estudio fue útil también para conocer e identificar cuáles eran las principales actividades del sistema, y además determinar cuáles eran los datos que el sistema iba a manejar, cómo iban a conformarse, y cómo iban a organizarse.

Dentro de la etapa de diseño se fijaron las especificaciones de procesos, es decir, se identificaron cuáles eran los principales procesos o tareas que debía realizar el sistema con todos y cada uno de los datos que tenía a la entrada y a la salida. Dado que en esta etapa eran ya conocidos todos los datos que se iban a almacenar y manipular, y se había especificado también cómo estaban relacionados, se preparó el diagrama entidad-relación para posteriormente tomarlo como base y crear la base de datos. Del mismo modo, se realizó el diseño de la interfaz del usuario tomando en consideración la necesidad de mejorarla en todos sentidos. Esta interfaz fue creada pensando en hacerla más consistente, más simple, y más fácil de usar.

La creación de la base de datos y la programación del sistema se realizaron dentro de la etapa de construcción del sistema. La instalación y configuración del software necesario para el desarrollo del sistema no fue demasiado complicada, por lo que gran parte del trabajo en esta etapa, y en general en todo el proyecto, fue la elaboración de todos los programas necesarios para lograr que el funcionamiento del sistema fuera el apropiado y que cumpliera con todos los requerimientos que fueron establecidos.

Finalmente, la etapa de pruebas y mantenimiento sirvió para verificar la correcta labor del sistema y la detección de los posibles fallos que pudiera tener. Se hizo la corrección de algunos problemas que fueron encontrados, la mayoría de ellos consistentes en pequeños detalles que fueron desapercibidos durante de la programación del sistema. Una vez liberado el sistema, la labor principal se centró en la realización continua de respaldos de la base de datos, dada la importancia de mantener seguros, íntegros y confiables los datos manejados por el sistema.

Durante el tiempo que lleva funcionando la nueva versión del Sistema de Acopio Permanente, se pudo observar que las herramientas de software que se escogieron para el desarrollo, incluyendo el sistema operativo Red Hat Linux, el servidor de web Apache, el manejador de bases de datos PostgreSQL, y el lenguaje de programación PHP, fueron una adecuada elección. Todos ellos representan proyectos de software libre realmente competitivos, que proporcionan alternativas bastante viables y eficientes para el desarrollo de sistemas.

La nueva versión del Sistema de Acopio Permanente lleva ya un par de años en funcionamiento y hasta el momento no se han tenido ningún tipo de problemas puesto que el sistema está disponible en todo momento. Cabe mencionar, que por el contrario, se han recibido críticas bastante buenas por parte de los usuarios.

Un punto sobresaliente para el éxito que se ha tenido con el sistema es sin duda la utilización de software bastante fiable y flexible que permite desarrollar aplicaciones bastante robustas, y con un grado de desempeño y confiabilidad muy bueno.

Se puede afirmar sin duda que es indispensable elegir una metodología adecuada para el desarrollo de sistemas, pero también es cierto que la elección del software utilizado en la construcción del sistema que cumpla con los requisitos de confiabilidad, seguridad, robustez y flexibilidad, puede garantizar en un porcentaje muy alto lograr obtener un sistema que resulte exitoso.



## APÉNDICES

**Código de la base de datos**

A continuación se listan todas las sentencias SQL necesarias para la creación de las tablas de la base de datos que se incluyeron en el archivo llamado *bd.sap.sql*.

-- Tablas comunes a todos los módulos

```
CREATE TABLE dependencias(
  id INT PRIMARY KEY NOT NULL,
  nombre TEXT NOT NULL,
  tipo INT);

INSERT INTO dependencias
  VALUES(21101,'Coordinacion y Consejo Tecnico de Humanidades',5);
INSERT INTO dependencias
  VALUES(21102,'Centro de Estudios sobre la Universidad',5);
INSERT INTO dependencias
  VALUES(21103,'Centro de Inv Interdisciplinarias en Ciencias y Hum',5);
INSERT INTO dependencias
  VALUES(21104,'Centro Universitario de Inv Bibliotecológicas',5);
INSERT INTO dependencias
  VALUES(21105,'Centro Coord y Dif de los Estudios Latinoamericanos',5);
INSERT INTO dependencias
  VALUES(21106,'Centro Regional Inv Multidisciplinarias de la UNAM',5);
INSERT INTO dependencias
  VALUES(21107,'Centro de Investigaciones sobre América del Norte',5);
INSERT INTO dependencias
  VALUES(212,'Instituto de Investigaciones Bibliográficas',5);
INSERT INTO dependencias
  VALUES(213,'Instituto de Investigaciones Económicas',5);
INSERT INTO dependencias
  VALUES(214,'Instituto de Investigaciones Estéticas',5);
INSERT INTO dependencias
  VALUES(215,'Instituto de Investigaciones Filosóficas',5);
INSERT INTO dependencias
  VALUES(216,'Instituto de Investigaciones Históricas',5);
INSERT INTO dependencias
  VALUES(217,'Instituto de Investigaciones Jurídicas',5);
INSERT INTO dependencias
  VALUES(218,'Instituto de Investigaciones Sociales',5);
INSERT INTO dependencias
  VALUES(219,'Instituto de Investigaciones Filológicas',5);
INSERT INTO dependencias
  VALUES(221,'Instituto de Investigaciones Antropológicas',5);
INSERT INTO dependencias
  VALUES(31101,'Consejo Técnico y Coord de Investigación Científica',5);
INSERT INTO dependencias
  VALUES(31102,'Centro de Ciencias de la Atmósfera',5);
INSERT INTO dependencias
  VALUES(31103,'Centro de Investigación sobre Fijación de Nitrógeno',5);
INSERT INTO dependencias
  VALUES(31104,'Centro de Neurobiología en Querétaro',5);
INSERT INTO dependencias
  VALUES(31105,'Centro de Instrumentos',5);
INSERT INTO dependencias
  VALUES(31106,'Centro de Ciencias Mat Condensada en Ensenada B. C.',5);
INSERT INTO dependencias
  VALUES(31107,'Centro de Ciencias Físicas',0);
```

```
INSERT INTO dependencias
  VALUES (31108, 'Centro de Investigación en Energía', 5);
INSERT INTO dependencias
  VALUES (312, 'Instituto de Astronomía', 5);
INSERT INTO dependencias
  VALUES (313, 'Instituto de Biología', 5);
INSERT INTO dependencias
  VALUES (314, 'Instituto de Física', 5);
INSERT INTO dependencias
  VALUES (315, 'Instituto de Geofísica', 5);
INSERT INTO dependencias
  VALUES (316, 'Instituto de Geografía', 5);
INSERT INTO dependencias
  VALUES (317, 'Instituto de Geología', 5);
INSERT INTO dependencias
  VALUES (318, 'Instituto de Investigaciones Biomédicas', 5);
INSERT INTO dependencias
  VALUES (319, 'Instituto de Matemáticas', 5);
INSERT INTO dependencias
  VALUES (321, 'Instituto de Química', 5);
INSERT INTO dependencias
  VALUES (322, 'Instituto de Inv en Matemáticas Aplicadas y Sistemas', 5);
INSERT INTO dependencias
  VALUES (323, 'Instituto de Ingeniería', 5);
INSERT INTO dependencias
  VALUES (324, 'Instituto de Investigaciones en Materiales', 5);
INSERT INTO dependencias
  VALUES (325, 'Instituto de Ciencias del Mar y Limnología', 5);
INSERT INTO dependencias
  VALUES (326, 'Instituto de Fisiología Celular', 5);
INSERT INTO dependencias
  VALUES (327, 'Instituto de Ciencias Nucleares', 5);
INSERT INTO dependencias
  VALUES (328, 'Instituto de Biotecnología', 5);
INSERT INTO dependencias
  VALUES (329, 'Instituto de Ecología', 5);
INSERT INTO dependencias
  VALUES (411, 'Facultad de Ciencias', 2);
INSERT INTO dependencias
  VALUES (412, 'Facultad de Ciencias Políticas y Sociales', 2);
INSERT INTO dependencias
  VALUES (413, 'Facultad de Contaduría y Administración', 2);
INSERT INTO dependencias
  VALUES (414, 'Facultad de Derecho', 2);
INSERT INTO dependencias
  VALUES (415, 'Facultad de Filosofía y Letras', 2);
INSERT INTO dependencias
  VALUES (416, 'Facultad de Ingeniería', 2);
INSERT INTO dependencias
  VALUES (417, 'Facultad de Medicina', 2);
INSERT INTO dependencias
  VALUES (418, 'Facultad de Química', 2);
INSERT INTO dependencias
  VALUES (419, 'Facultad de Medicina Veterinaria y Zootecnia', 2);
INSERT INTO dependencias
  VALUES (421, 'Facultad de Odontología', 2);
INSERT INTO dependencias
  VALUES (422, 'Facultad de Psicología', 2);
```

```
INSERT INTO dependencias
  VALUES(423,'Facultad de Economía',2);
INSERT INTO dependencias
  VALUES(424,'Facultad de Arquitectura',2);
INSERT INTO dependencias
  VALUES(432,'Escuela Nacional de Artes Plásticas',2);
INSERT INTO dependencias
  VALUES(434,'Escuela Nacional de Enfermería y Obstetricia',4);
INSERT INTO dependencias
  VALUES(435,'Escuela Nacional de Música',4);
INSERT INTO dependencias
  VALUES(436,'Escuela Nacional de Trabajo Social',2);
INSERT INTO dependencias
  VALUES(441,'Facultad de Estudios Superiores Cuautitlán',2);
INSERT INTO dependencias
  VALUES(442,'Escuela Nacional de Estudios Profesionales Acatlán',2);
INSERT INTO dependencias
  VALUES(443,'Escuela Nacional de Estudios Profesionales Iztacala',3);
INSERT INTO dependencias
  VALUES(444,'Escuela Nacional de Estudios Profesionales Aragón',2);
INSERT INTO dependencias
  VALUES(445,'Facultad de Estudios Superiores Zaragoza',3);
INSERT INTO dependencias
  VALUES(452,'Escuela Nacional Preparatoria',1);
INSERT INTO dependencias
  VALUES(45201,'ENP: Gabino Barreda',0);
INSERT INTO dependencias
  VALUES(45202,'ENP: Erasmo Castellanos Quinto',0);
INSERT INTO dependencias
  VALUES(45203,'ENP: Justo Sierra',0);
INSERT INTO dependencias
  VALUES(45204,'ENP: Vidal Castañeda y Nájera',0);
INSERT INTO dependencias
  VALUES(45205,'ENP: José Vasconcelos',0);
INSERT INTO dependencias
  VALUES(45206,'ENP: Antonio Caso',0);
INSERT INTO dependencias
  VALUES(45207,'ENP: Ezequiel A. Chávez',0);
INSERT INTO dependencias
  VALUES(45208,'ENP: Miguel E. Schulz',0);
INSERT INTO dependencias
  VALUES(45209,'ENP: Pedro de Alba',0);
INSERT INTO dependencias
  VALUES(472,'Colegio de Ciencias y Humanidades',1);
INSERT INTO dependencias
  VALUES(522,'Dirección General de Servicios de Cómputo Académico',0);
INSERT INTO dependencias
  VALUES(523,'Dirección General de Asuntos del Personal Académico',0);
INSERT INTO dependencias
  VALUES(611,'Coordinación de Difusión Cultural',6);
INSERT INTO dependencias
  VALUES(612,'Centro de Enseñanza de Lenguas Extranjeras',7);
INSERT INTO dependencias
  VALUES(613,'Centro de Enseñanza para Extranjeros',7);
INSERT INTO dependencias
  VALUES(614,'Centro Universitario de Estudios Cinematográficos',7);
INSERT INTO dependencias
  VALUES(615,'Centro Universitario de Teatro',7);
```

```
INSERT INTO dependencias
VALUES(616,'Casa del Lago',6);
INSERT INTO dependencias
VALUES(617,'Museo Universitario del Chopo',6);
INSERT INTO dependencias
VALUES(621,'Dirección General de Artes Plásticas',5);
INSERT INTO dependencias
VALUES(622,'Dirección General de Actividades Musicales',6);
INSERT INTO dependencias
VALUES(623,'Dirección General de Actividades Cinematográficas',6);
INSERT INTO dependencias
VALUES(624,'Dirección de Literatura',6);
INSERT INTO dependencias
VALUES(625,'Dirección de Teatro y Danza',6);
INSERT INTO dependencias
VALUES(63102,'Programa Universitario de Alimentos',0);
INSERT INTO dependencias
VALUES(63103,'Programa Universitario de Energía',0);
INSERT INTO dependencias
VALUES(63104,'Programa Universitario de Investigaciones en Salud',0);
INSERT INTO dependencias
VALUES(63105,'Programa Universitario de Estudios sobre la Ciudad',0);
INSERT INTO dependencias
VALUES(63106,'Programa Universitario de Medio Ambiente',0);
INSERT INTO dependencias
VALUES(641,'Dirección General de Divulgación de la Ciencia',);
INSERT INTO dependencias
VALUES(653,'Dirección General de Televisión Universitaria',6);
INSERT INTO dependencias
VALUES(656,'Dirección General de Radio UNAM',6);
```

```
GRANT SELECT ON dependencias TO nobody;
```

```
CREATE TABLE usuarios(
login text PRIMARY KEY NOT NULL,
password text NOT NULL,
grupo text NOT NULL,
dependencia int REFERENCES dependencias
ON UPDATE CASCADE ON DELETE CASCADE);
```

```
GRANT SELECT ON usuarios TO nobody;
```

```
-- Tablas para el módulo de Proyectos de investigación
```

```
CREATE TABLE camposciencia(
id INT PRIMARY KEY NOT NULL,
nombre TEXT NOT NULL);

INSERT INTO camposciencia VALUES(0,'No definido');
INSERT INTO camposciencia VALUES(1,'Ciencias Naturales y Exactas');
INSERT INTO camposciencia VALUES(2,'Ingeniería y Tecnología');
INSERT INTO camposciencia VALUES(3,'Ciencias Agrícolas');
INSERT INTO camposciencia VALUES(4,'Ciencias Médicas');
INSERT INTO camposciencia VALUES(5,'Ciencias Sociales');
INSERT INTO camposciencia VALUES(6,'Humanidades');
```

```
GRANT SELECT ON camposciencia TO nobody;
```

```
CREATE TABLE proyectos (  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_proyectos'),  
  dependencia INT REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  nombre TEXT NOT NULL,  
  lineainv TEXT,  
  responsable TEXT NOT NULL,  
  rfcresp TEXT,  
  numtrab TEXT,  
  camposciencia INT NOT NULL DEFAULT 0 REFERENCES camposciencia  
  ON UPDATE CASCADE ON DELETE SET DEFAULT,  
  colaboradores TEXT,  
  tipoact TEXT NOT NULL DEFAULT 'noindefinido',  
  status TEXT NOT NULL DEFAULT 'continuacion',  
  terminado BOOLEAN NOT NULL DEFAULT 'F',  
  alumnosbach INT DEFAULT 0,  
  alumnoslic INT DEFAULT 0,  
  alumnospost INT DEFAULT 0,  
  comentarios TEXT);
```

```
CREATE SEQUENCE seq_proyectos;
```

```
GRANT ALL ON proyectos TO nobody;  
GRANT ALL ON seq_proyectos TO nobody;
```

```
CREATE TABLE fuentesfinanc (  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO fuentesfinanc  
  VALUES(0,'No definido');  
INSERT INTO fuentesfinanc  
  VALUES(1,'Presupuesto directo(UNAM)');  
INSERT INTO fuentesfinanc  
  VALUES(2,'Programas institucionales (DGAPA,DGEP,PAPIIT,PAPIME)');  
INSERT INTO fuentesfinanc  
  VALUES(3,'Intercambio académico');  
INSERT INTO fuentesfinanc  
  VALUES(4,'Fondos propios');  
INSERT INTO fuentesfinanc  
  VALUES(5,'Gobierno');  
INSERT INTO fuentesfinanc  
  VALUES(6,'Privado');  
INSERT INTO fuentesfinanc  
  VALUES(7,'Educación superior');  
INSERT INTO fuentesfinanc  
  VALUES(8,'Instituciones privadas no lucrativas');  
INSERT INTO fuentesfinanc  
  VALUES(9,'Exterior');  
INSERT INTO fuentesfinanc  
  VALUES(99,'Sin financiamiento');
```

```
GRANT SELECT ON fuentesfinanc TO nobody;
```

```
CREATE TABLE proy_fuentefinanc(  
  proyecto INT NOT NULL REFERENCES proyectos  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  fuentefinanc INT NOT NULL REFERENCES fuentesfinanc  
  ON UPDATE CASCADE ON DELETE CASCADE);
```

```
GRANT ALL on proy_fuentefinanc TO nobody;
```

```
-- Tablas para el módulo de Productos editoriales
```

```
CREATE TABLE tiposmaterial(  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO tiposmaterial VALUES(0,'No definido');  
INSERT INTO tiposmaterial VALUES(1,'CD');  
INSERT INTO tiposmaterial VALUES(2,'CD ROM');  
INSERT INTO tiposmaterial VALUES(3,'Videocasete');  
INSERT INTO tiposmaterial VALUES(4,'Audio casete');  
INSERT INTO tiposmaterial VALUES(5,'Otro');
```

```
GRANT SELECT ON tiposmaterial TO nobody;
```

```
CREATE TABLE receptores(  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO receptores VALUES(0,'No definido');  
INSERT INTO receptores VALUES(1,'Bachillerato');  
INSERT INTO receptores VALUES(2,'Licenciatura');  
INSERT INTO receptores VALUES(3,'Posgrado');  
INSERT INTO receptores VALUES(4,'Técnico');
```

```
GRANT SELECT ON receptores TO nobody;
```

```
CREATE TABLE periodicidades(  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO periodicidades VALUES(0,'No definida');  
INSERT INTO periodicidades VALUES(1,'Anual');  
INSERT INTO periodicidades VALUES(2,'Bimestral');  
INSERT INTO periodicidades VALUES(3,'Cuatrimestral');  
INSERT INTO periodicidades VALUES(4,'Irregular');  
INSERT INTO periodicidades VALUES(5,'Mensual');  
INSERT INTO periodicidades VALUES(6,'Quincenal');  
INSERT INTO periodicidades VALUES(7,'Semanal');  
INSERT INTO periodicidades VALUES(8,'Trimestral');  
INSERT INTO periodicidades VALUES(9,'Otra');
```

```
GRANT SELECT ON periodicidades TO nobody;
```

```
CREATE TABLE prodcatalogos(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodcatalogos'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  ciudad TEXT,  
  totalpag INT DEFAULT 0,  
  tiraje TEXT,  
  numero TEXT,  
  volumen TEXT,  
  issn TEXT);
```

```
CREATE SEQUENCE seq_prodcatalogos;  
GRANT ALL ON prodcatalogos TO nobody;  
GRANT ALL ON seq_prodcatalogos TO nobody;
```

```
CREATE TABLE prodlibros(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodlibros'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  ciudad TEXT,  
  totalpag INT DEFAULT 0,  
  tiraje TEXT,  
  edicion TEXT,  
  coleccion TEXT,  
  isbn TEXT);
```

```
CREATE SEQUENCE seq_prodlibros;  
GRANT ALL ON prodlibros TO nobody;  
GRANT ALL ON seq_prodlibros TO nobody;
```

```
CREATE TABLE prodmatapoyo(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodmatapoyo'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  ciudad TEXT,  
  totalpag INT DEFAULT 0,  
  tiraje TEXT,  
  edicion TEXT,  
  coleccion TEXT,  
  isbn TEXT,  
  receptor INT NOT NULL DEFAULT 0 REFERENCES receptores  
  ON UPDATE CASCADE ON DELETE SET DEFAULT,  
  tipomat TEXT);
```



```
CREATE SEQUENCE seq_prodmatapoyo;

GRANT ALL ON prodmatapoyo TO nobody;
GRANT ALL ON seq_prodmatapoyo TO nobody;

CREATE TABLE prodsoftware(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodsoftware'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  titulo TEXT NOT NULL,
  autor TEXT,
  coautores TEXT,
  anio TEXT,
  version TEXT,
  lenguaje TEXT,
  tamanio TEXT);

CREATE SEQUENCE seq_prodsoftware;

GRANT ALL ON prodsoftware TO nobody;
GRANT ALL ON seq_prodsoftware TO nobody;

CREATE TABLE prodinformes(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodinformes'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  titulo TEXT NOT NULL,
  autor TEXT,
  coautores TEXT,
  anio TEXT,
  totalpag INT);

CREATE SEQUENCE seq_prodinformes;
GRANT ALL ON prodinformes TO nobody;
GRANT ALL ON seq_prodinformes TO nobody;

CREATE TABLE prodcarteles(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodcarteles'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  titulo TEXT NOT NULL,
  autor TEXT,
  coautores TEXT,
  anio TEXT,
  tiraje INT,
  evento TEXT);

CREATE SEQUENCE seq_prodcarteles;

GRANT ALL ON prodcarteles TO nobody;
GRANT ALL ON seq_prodcarteles TO nobody;
```

```
CREATE TABLE prodcartograficos(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodcartograficos'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  escalal TEXT,  
  escala2 TEXT);
```

```
CREATE SEQUENCE seq_prodcartograficos;
```

```
GRANT ALL ON prodcartograficos TO nobody;  
GRANT ALL ON seq_prodcartograficos TO nobody;
```

```
CREATE TABLE prodcuadernos(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodcuadernos'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  edicion TEXT,  
  totalpag TEXT,  
  tiraje TEXT,  
  coleccion TEXT);
```

```
CREATE SEQUENCE seq_prodcuadernos;
```

```
GRANT ALL ON prodcuadernos TO nobody;  
GRANT ALL ON seq_prodcuadernos TO nobody;
```

```
CREATE TABLE prodmataudiovis(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodmataudiovis'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  tipomaterial INT NOT NULL DEFAULT 0 REFERENCES tiposmaterial  
  ON UPDATE CASCADE ON DELETE SET DEFAULT);
```

```
CREATE SEQUENCE seq_prodmataudiovis;
```

```
GRANT ALL ON prodmataudiovis TO nobody;  
GRANT ALL ON seq_prodmataudiovis TO nobody;
```

```
CREATE TABLE prodfolletos(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodfolletos'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,
```

```
autor TEXT,  
coautores TEXT,  
anio TEXT,  
editorial TEXT,  
ciudad TEXT,  
totalpag INT DEFAULT 0,  
tiraje TEXT,  
numero TEXT);
```

```
CREATE SEQUENCE seq_prodfolletos;
```

```
GRANT ALL ON prodfolletos TO nobody;  
GRANT ALL ON seq_prodfolletos TO nobody;
```

```
CREATE TABLE prodboletines(  
id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodboletines'),  
dependencia INT NOT NULL REFERENCES dependencias  
ON UPDATE CASCADE ON DELETE CASCADE,  
titulo TEXT NOT NULL,  
autor TEXT,  
coautores TEXT,  
anio TEXT,  
editorial TEXT,  
ciudad TEXT,  
totalpag INT DEFAULT 0,  
tiraje TEXT,  
numero TEXT,  
volumen TEXT,  
fechatiraje TEXT,  
periodicidad INT NOT NULL DEFAULT 0 REFERENCES periodicidades  
ON UPDATE CASCADE ON DELETE SET DEFAULT);
```

```
CREATE SEQUENCE seq_prodboletines;
```

```
GRANT ALL ON prodboletines TO nobody;  
GRANT ALL ON seq_prodboletines TO nobody;
```

```
CREATE TABLE prodmemorias(  
id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodmemorias'),  
dependencia INT NOT NULL REFERENCES dependencias  
ON UPDATE CASCADE ON DELETE CASCADE,  
titulo TEXT NOT NULL,  
autor TEXT,  
coautores TEXT,  
anio TEXT,  
editorial TEXT,  
ciudad TEXT,  
totalpag INT DEFAULT 0,  
tiraje TEXT,  
numero TEXT,  
volumen TEXT);
```

```
CREATE SEQUENCE seq_prodmemorias;
```

```
GRANT ALL ON prodmemorias TO nobody;
```

```
GRANT ALL ON seq_prodmemorias TO nobody;
```

```
CREATE TABLE prodrevistas(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_prodrevistas'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  ciudad TEXT,  
  totalpag INT DEFAULT 0,  
  tiraje TEXT,  
  numero TEXT,  
  volumen TEXT,  
  arbitrada BOOLEAN,  
  issn TEXT,  
  periodicidad INT NOT NULL DEFAULT 0 REFERENCES periodicidades  
  ON UPDATE CASCADE ON DELETE SET DEFAULT);
```

```
CREATE SEQUENCE seq_prodrevistas;
```

```
GRANT ALL ON prodrevistas TO nobody;  
GRANT ALL ON seq_prodrevistas TO nobody;
```

```
-- Tablas para el módulo de Publicaciones del personal
```

```
CREATE TABLE pubarticulosmem(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubarticulosmem'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  memoria TEXT NOT NULL,  
  paginaini INT DEFAULT 0,  
  paginafin INT DEFAULT 0,  
  volumen TEXT,  
  numero TEXT);
```

```
CREATE SEQUENCE seq_pubarticulosmem;  
GRANT ALL ON pubarticulosmem TO nobody;  
GRANT ALL ON seq_pubarticulosmem TO nobody;
```

```
CREATE TABLE pubinformes(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubinformes'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  totalpag INT);
```

```
CREATE SEQUENCE seq_pubinformes;
```

```
GRANT ALL ON pubinformes TO nobody;  
GRANT ALL ON seq_pubinformes TO nobody;
```

```
CREATE TABLE pubarticulosra(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubarticulosra'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  revista TEXT NOT NULL,  
  paginaini INT DEFAULT 0,  
  paginafin INT DEFAULT 0,  
  volumen TEXT,  
  numero TEXT);
```

```
CREATE SEQUENCE seq_pubarticulosra;
```

```
GRANT ALL ON pubarticulosra TO nobody;  
GRANT ALL ON seq_pubarticulosra TO nobody;
```

```
CREATE TABLE pubarticulosrna(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubarticulosrna'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  revista TEXT NOT NULL,  
  paginaini INT DEFAULT 0,  
  paginafin INT DEFAULT 0,  
  volumen TEXT,  
  numero TEXT);
```

```
CREATE SEQUENCE seq_pubarticulosrna;
```

```
GRANT ALL ON pubarticulosrna TO nobody;  
GRANT ALL ON seq_pubarticulosrna TO nobody;
```

```
CREATE TABLE pubreportes(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubreportes'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  totalpag INT);
```

```
CREATE SEQUENCE seq_pubreportes;
```

```
GRANT ALL ON pubreportes TO nobody;
```

```
GRANT ALL ON seq_pubreportes TO nobody;
```

```
CREATE TABLE pubprologos(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubprologos'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  libro TEXT NOT NULL,  
  paginaini INT DEFAULT 0,  
  paginafin INT DEFAULT 0,  
  editorial TEXT,  
  coleccion TEXT,  
  edicion TEXT,  
  ciudad TEXT,  
  tiraje TEXT,  
  isbn TEXT);
```

```
CREATE SEQUENCE seq_pubprologos;
```

```
GRANT ALL ON pubprologos TO nobody;  
GRANT ALL ON seq_pubprologos TO nobody;
```

```
CREATE TABLE pubcapituloslib(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubcapituloslib'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  libro TEXT NOT NULL,  
  paginaini INT DEFAULT 0,  
  paginafin INT DEFAULT 0,  
  editorial TEXT,  
  coleccion TEXT,  
  edicion TEXT,  
  ciudad TEXT,  
  tiraje TEXT,  
  isbn TEXT,  
  editor TEXT);
```

```
CREATE SEQUENCE seq_pubcapituloslib;
```

```
GRANT ALL ON pubcapituloslib TO nobody;  
GRANT ALL ON seq_pubcapituloslib TO nobody;
```

```
CREATE TABLE pubintroducciones(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubintroducciones'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,
```

```
anio TEXT,  
publicacion TEXT NOT NULL,  
paginaini INT DEFAULT 0,  
paginafin INT DEFAULT 0,  
editorial TEXT,  
coleccion TEXT,  
edicion TEXT,  
ciudad TEXT,  
tiraje TEXT,  
isbn TEXT);
```

```
CREATE SEQUENCE seq_pubintroducciones;
```

```
GRANT ALL ON pubintroducciones TO nobody;  
GRANT ALL ON seq_pubintroducciones TO nobody;
```

```
CREATE TABLE publicibros(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_publicibros'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  tiraje TEXT,  
  totalpag INT,  
  coleccion TEXT,  
  edicion TEXT,  
  ciudad TEXT,  
  isbn TEXT);
```

```
CREATE SEQUENCE seq_publicibros;
```

```
GRANT ALL ON publicibros TO nobody;  
GRANT ALL ON seq_publicibros TO nobody;
```

```
CREATE TABLE pubmatapoyo(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubmatapoyo'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  autor TEXT,  
  coautores TEXT,  
  anio TEXT,  
  editorial TEXT,  
  tiraje TEXT,  
  totalpag INT,  
  coleccion TEXT,  
  edicion TEXT,  
  isbn TEXT);
```

```
CREATE SEQUENCE seq_pubmatapoyo;
```

```
GRANT ALL ON pubmatapoyo TO nobody;  
GRANT ALL ON seq_pubmatapoyo TO nobody;
```

```
CREATE TABLE pubcatalogos(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubcatalogos'),
  dependencia INT NOT NULL REFERENCES dependencias
ON UPDATE CASCADE ON DELETE CASCADE,
  titulo TEXT NOT NULL,
  autor TEXT,
  coautores TEXT,
  anio TEXT,
  editorial TEXT,
  tiraje TEXT,
  totalpag INT,
  ciudad TEXT,
  volumen TEXT,
  numero TEXT,
  issn TEXT);

CREATE SEQUENCE seq_pubcatalogos;

GRANT ALL ON pubcatalogos TO nobody;
GRANT ALL ON seq_pubcatalogos TO nobody;

CREATE TABLE pubarticulosperiod(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_pubarticulosperiod'),
  dependencia INT NOT NULL REFERENCES dependencias
ON UPDATE CASCADE ON DELETE CASCADE,
  titulo TEXT NOT NULL,
  autor TEXT,
  coautores TEXT,
  anio TEXT,
  periodico TEXT,
  paginaini INT,
  paginafin INT,
  epoca TEXT,
  numero TEXT,
  fecha DATE);

CREATE SEQUENCE seq_pubarticulosperiod;

GRANT ALL ON pubarticulosperiod TO nobody;
GRANT ALL ON seq_pubarticulosperiod TO nobody;

-- Tablas para el módulo de Personal académico

CREATE TABLE niveles(
  id INT PRIMARY KEY NOT NULL,
  nombre TEXT NOT NULL);

INSERT INTO niveles VALUES(1,'No especificado');
INSERT INTO niveles VALUES(2,'Licenciatura ');
INSERT INTO niveles VALUES(3,'Especialización ');
INSERT INTO niveles VALUES(4,'Maestría');
INSERT INTO niveles VALUES(5,'Doctorado');
INSERT INTO niveles VALUES(6,'Menos de Licenciatura');
```



```
GRANT SELECT ON niveles TO nobody;
```

```
CREATE TABLE estudios(  
    id INT PRIMARY KEY NOT NULL,  
    nombre TEXT NOT NULL);
```

```
INSERT INTO estudios VALUES(0,'No está realizando estudios');  
INSERT INTO estudios VALUES(1,'Licenciatura');  
INSERT INTO estudios VALUES(2,'Especialización ');  
INSERT INTO estudios VALUES(3,'Maestría');  
INSERT INTO estudios VALUES(4,'Doctorado');  
INSERT INTO estudios VALUES(5,'Post-doctorado');
```

```
GRANT SELECT ON estudios TO nobody;
```

```
CREATE TABLE lugares(  
    id INT PRIMARY KEY NOT NULL,  
    nombre TEXT NOT NULL);
```

```
INSERT INTO lugares VALUES(0,'No especificado');  
INSERT INTO lugares VALUES(1,'Nacional');  
INSERT INTO lugares VALUES(2,'Extranjero');
```

```
GRANT SELECT ON lugares TO nobody;
```

```
CREATE TABLE categorias(  
    id INT PRIMARY KEY NOT NULL,  
    nombre TEXT NOT NULL);
```

```
INSERT INTO categorias  
VALUES(0,'No especificada');  
INSERT INTO categorias  
VALUES(1,'Profesor de Asignatura A');  
INSERT INTO categorias  
VALUES(2,'Profesor de Asignatura B');  
INSERT INTO categorias  
VALUES(3,'Ayudante de Profesor de Asignatura A');  
INSERT INTO categorias  
VALUES(4,'Ayudante de Profesor de Asignatura B');  
INSERT INTO categorias  
VALUES(5,'Profesor de Enseñanza Media Superior A');  
INSERT INTO categorias  
VALUES(6,'Profesor de Enseñanza Media Superior B');  
INSERT INTO categorias  
VALUES(7,'Profesor de Carrera Asociado A Medio Tiempo');  
INSERT INTO categorias  
VALUES(8,'Profesor de Carrera Asociado B Medio Tiempo');  
INSERT INTO categorias  
VALUES(9,'Profesor de Carrera Asociado C Medio Tiempo');  
INSERT INTO categorias  
VALUES(10,'Profesor de Carrera Asociado A Tiempo Completo');  
INSERT INTO categorias  
VALUES(11,'Profesor de Carrera Asociado B Tiempo Completo');  
INSERT INTO categorias  
VALUES(12,'Profesor de Carrera Asociado C Tiempo Completo');  
INSERT INTO categorias  
VALUES(13,'Profesor de Carrera Titular A Medio Tiempo');
```

```
INSERT INTO categorias
VALUES (14, 'Profesor de Carrera Titular B Medio Tiempo');
INSERT INTO categorias
VALUES (15, 'Profesor de Carrera Titular C Medio Tiempo');
INSERT INTO categorias
VALUES (16, 'Profesor de Carrera Titular A Tiempo Completo');
INSERT INTO categorias
VALUES (17, 'Profesor de Carrera Titular B Tiempo Completo');
INSERT INTO categorias
VALUES (18, 'Profesor de Carrera Titular C Tiempo Completo');
INSERT INTO categorias
VALUES (19, 'Ayudante de Profesor A Medio Tiempo');
INSERT INTO categorias
VALUES (20, 'Ayudante de Profesor B Medio Tiempo');
INSERT INTO categorias
VALUES (21, 'Ayudante de Profesor C Medio Tiempo');
INSERT INTO categorias
VALUES (22, 'Ayudante de Profesor A Tiempo Completo');
INSERT INTO categorias
VALUES (23, 'Ayudante de Profesor B Tiempo Completo');
INSERT INTO categorias
VALUES (24, 'Ayudante de Profesor C Tiempo Completo');
INSERT INTO categorias
VALUES (35, 'Técnico Académico Asociado B Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (33, 'Técnico Académico Asociado C Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (36, 'Técnico Académico Asociado C Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (61, 'Técnico Académico Auxiliar A Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (28, 'Técnico Académico Auxiliar A Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (26, 'Técnico Académico Auxiliar B Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (29, 'Técnico Académico Auxiliar B Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (27, 'Técnico Académico Auxiliar C Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (30, 'Técnico Académico Auxiliar C Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (37, 'Técnico Académico Titular A Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (40, 'Técnico Académico Titular A Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (38, 'Técnico Académico Titular B Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (41, 'Técnico Académico Titular B Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (39, 'Técnico Académico Titular C Medio Tiempo Investigación');
INSERT INTO categorias
VALUES (42, 'Técnico Académico Titular C Tiempo Completo Investigación');
INSERT INTO categorias
VALUES (49, 'Investigador de Carrera Titular A Medio Tiempo');
INSERT INTO categorias
VALUES (67, 'Técnico Académico Asociado A Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (70, 'Técnico Académico Asociado A Tiempo Completo Docencia');
```

```
INSERT INTO categorias
VALUES (43, 'Investigador de Carrera Asociado A Medio Tiempo');
INSERT INTO categorias
VALUES (44, 'Investigador de Carrera Asociado B Medio Tiempo');
INSERT INTO categorias
VALUES (45, 'Investigador de Carrera Asociado C Medio Tiempo');
INSERT INTO categorias
VALUES (46, 'Investigador de Carrera Asociado A Tiempo Completo');
INSERT INTO categorias
VALUES (47, 'Investigador de Carrera Asociado B Tiempo Completo');
INSERT INTO categorias
VALUES (48, 'Investigador de Carrera Asociado C Tiempo Completo');
INSERT INTO categorias
VALUES (68, 'Técnico Académico Asociado B Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (50, 'Investigador de Carrera Titular B Medio Tiempo');
INSERT INTO categorias
VALUES (51, 'Investigador de Carrera Titular C Medio Tiempo');
INSERT INTO categorias
VALUES (52, 'Investigador de Carrera Titular A Tiempo Completo');
INSERT INTO categorias
VALUES (53, 'Investigador de Carrera Titular B Tiempo Completo');
INSERT INTO categorias
VALUES (54, 'Investigador de Carrera Titular C Tiempo Completo');
INSERT INTO categorias
VALUES (55, 'Ayudante de Investigador A Medio Tiempo');
INSERT INTO categorias
VALUES (56, 'Ayudante de Investigador B Medio Tiempo');
INSERT INTO categorias
VALUES (57, 'Ayudante de Investigador C Medio Tiempo');
INSERT INTO categorias
VALUES (58, 'Ayudante de Investigador A Tiempo Completo');
INSERT INTO categorias
VALUES (59, 'Ayudante de Investigador B Tiempo Completo');
INSERT INTO categorias
VALUES (60, 'Ayudante de Investigador C Tiempo Completo');
INSERT INTO categorias
VALUES (71, 'Técnico Académico Asociado B Tiempo Completo Docencia');
INSERT INTO categorias
VALUES (69, 'Técnico Académico Asociado C Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (72, 'Técnico Académico Asociado C Tiempo Completo Docencia');
INSERT INTO categorias
VALUES (25, 'Técnico Académico Auxiliar A Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (64, 'Técnico Académico Auxiliar A Tiempo Completo Docencia');
INSERT INTO categorias
VALUES (62, 'Técnico Académico Auxiliar B Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (65, 'Técnico Académico Auxiliar B Tiempo Completo Docencia');
INSERT INTO categorias
VALUES (63, 'Técnico Académico Auxiliar C Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (66, 'Técnico Académico Auxiliar C Tiempo Completo Docencia');
INSERT INTO categorias
VALUES (73, 'Técnico Académico Titular A Medio Tiempo Docencia');
INSERT INTO categorias
VALUES (76, 'Técnico Académico Titular A Tiempo Completo Docencia');
```

```
INSERT INTO categorias
VALUES(74, 'Técnico Académico Titular B Medio Tiempo Docencia');
INSERT INTO categorias
VALUES(77, 'Técnico Académico Titular B Tiempo Completo Docencia');
INSERT INTO categorias
VALUES(75, 'Técnico Académico Titular C Medio Tiempo Docencia');
INSERT INTO categorias
VALUES(78, 'Técnico Académico Titular C Tiempo Completo Docencia');
INSERT INTO categorias
VALUES(31, 'Técnico Académico Asociado A Medio Tiempo Investigación');
INSERT INTO categorias
VALUES(34, 'Técnico Académico Asociado A Tiempo Completo Investigación');
INSERT INTO categorias
VALUES(32, 'Técnico Académico Asociado B Medio Tiempo Investigación');
INSERT INTO categorias
VALUES(79, 'Profesor o Investigador Emérito');
INSERT INTO categorias
VALUES(80, 'Profesor e Investigador visitante');
```

```
GRANT SELECT ON categorias TO nobody;
```

```
CREATE TABLE academicos(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_academicos'),
  dependencia INT REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  rfc TEXT,
  tesislic INT DEFAULT 0,
  tesismae INT DEFAULT 0,
  tesisdoc INT DEFAULT 0,
  serviciosoc INT DEFAULT 0,
  estudio INT NOT NULL DEFAULT 0 REFERENCES estudios
  ON UPDATE CASCADE ON DELETE SET DEFAULT,
  lugarest INT NOT NULL DEFAULT 0 REFERENCES lugares
  ON UPDATE CASCADE ON DELETE SET DEFAULT,
  nivel INT NOT NULL DEFAULT 1 REFERENCES niveles
  ON UPDATE CASCADE ON DELETE SET DEFAULT);
```

```
CREATE SEQUENCE seq_academicos;
```

```
GRANT ALL ON academicos TO nobody;
GRANT ALL ON seq_academicos TO nobody;
```

```
CREATE TABLE acad_fuentefinanc(
  academico INT NOT NULL REFERENCES academicos
  ON UPDATE CASCADE ON DELETE CASCADE,
  fuentefinanc INT NOT NULL REFERENCES fuentesfinanc
  ON UPDATE CASCADE ON DELETE CASCADE);
```

```
GRANT ALL ON acad_fuentefinanc TO nobody;
```

```
CREATE TABLE acad_categoria(
  academico INT NOT NULL REFERENCES academicos
  ON UPDATE CASCADE ON DELETE CASCADE,
  categoria INT NOT NULL REFERENCES categorias
  ON UPDATE CASCADE ON DELETE CASCADE);
```

```
GRANT ALL ON acad_categoria TO nobody;

-- Tablas para el módulo de Actividades académicas

CREATE TABLE acadseminarios(
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadseminarios'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE);

CREATE SEQUENCE seq_acadseminarios;

GRANT ALL ON acadseminarios TO nobody;
GRANT ALL ON seq_acadseminarios TO nobody;

CREATE TABLE acadcongresos(
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadcongresos'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE);

CREATE SEQUENCE seq_acadcongresos;

GRANT ALL ON acadcongresos TO nobody;
GRANT ALL ON seq_acadcongresos TO nobody;

CREATE TABLE acadconferencias(
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadconferencias'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE);

CREATE SEQUENCE seq_acadconferencias;

GRANT ALL ON acadconferencias TO nobody;
GRANT ALL ON seq_acadconferencias TO nobody;

CREATE TABLE acadsimposios(
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadsimposios'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE);
```

```
CREATE SEQUENCE seq_acadsimposios;

GRANT ALL ON acadsimposios TO nobody;
GRANT ALL ON seq_acadsimposios TO nobody;

CREATE TABLE acadforos(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadforos'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_acadforos;
GRANT ALL ON acadforos TO nobody;
GRANT ALL ON seq_acadforos TO nobody;

CREATE TABLE acadencuentros(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadencuentros'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_acadencuentros;

GRANT ALL ON acadencuentros TO nobody;
GRANT ALL ON seq_acadencuentros TO nobody;

CREATE TABLE acacoloquios(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acacoloquios'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_acacoloquios;

GRANT ALL ON acacoloquios TO nobody;
GRANT ALL ON seq_acacoloquios TO nobody;

CREATE TABLE acadconcursos(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadconcursos'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  asistentes INT,
  fechaini DATE,
```

```
    fechafin DATE);

CREATE SEQUENCE seq_acadconcursos;

GRANT ALL ON acadconcursos TO nobody;
GRANT ALL ON seq_acadconcursos TO nobody;

CREATE TABLE acadtalleres(
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadtalleres'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE,
    sesiones INT,
    ponente TEXT);

CREATE SEQUENCE seq_acadtalleres;

GRANT ALL ON acadtalleres TO nobody;
GRANT ALL ON seq_acadtalleres TO nobody;

CREATE TABLE acadcursos (
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acadcursos'),
    dependencia INT NOT NULL REFERENCES dependencias
    ON UPDATE CASCADE ON DELETE CASCADE,
    nombre TEXT NOT NULL,
    sede TEXT,
    asistentes INT,
    fechaini DATE,
    fechafin DATE,
    sesiones INT,
    ponente TEXT);

CREATE SEQUENCE seq_acadcursos;

GRANT ALL ON acadcursos TO nobody;
GRANT ALL ON seq_acadcursos TO nobody;

-- Tablas para el módulo de Actividades de difusión

CREATE TABLE tiposlectura(
    id INT PRIMARY KEY NOT NULL,
    nombre TEXT NOT NULL);

INSERT INTO tiposlectura VALUES(0,'No especificada');
INSERT INTO tiposlectura VALUES(1,'Cuento');
INSERT INTO tiposlectura VALUES(2,'Poesía');
INSERT INTO tiposlectura VALUES(3,'Dramatización');

GRANT SELECT ON tiposlectura TO nobody;
```

```
CREATE TABLE tipospresentacion(  
    id INT PRIMARY KEY NOT NULL,  
    nombre TEXT NOT NULL);  
  
INSERT INTO tipospresentacion VALUES(0,'No especificado');  
INSERT INTO tipospresentacion VALUES(1,'CDROM');  
INSERT INTO tipospresentacion VALUES(2,'Libro');  
INSERT INTO tipospresentacion VALUES(3,'Revista');  
INSERT INTO tipospresentacion VALUES(4,'Video');  
INSERT INTO tipospresentacion VALUES(5,'Catálogo');  
INSERT INTO tipospresentacion VALUES(6,'Escritores');  
INSERT INTO tipospresentacion VALUES(7,'Otro');  
  
GRANT SELECT ON tipospresentacion TO nobody;  
  
CREATE TABLE difconciertos(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difconciertos'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    nombre TEXT NOT NULL,  
    sede TEXT,  
    recinto TEXT,  
    asistentes INT,  
    fechaini DATE,  
    fechafin DATE,  
    funciones INT);  
  
CREATE SEQUENCE seq_difconciertos;  
  
GRANT ALL ON difconciertos TO nobody;  
GRANT ALL ON seq_difconciertos TO nobody;  
  
CREATE TABLE difmuestras(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difmuestras'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    nombre TEXT NOT NULL,  
    sede TEXT,  
    recinto TEXT,  
    asistentes INT,  
    fechaini DATE,  
    fechafin DATE,  
    funciones INT);  
  
CREATE SEQUENCE seq_difmuestras;  
  
GRANT ALL ON difmuestras TO nobody;  
GRANT ALL ON seq_difmuestras TO nobody;  
  
CREATE TABLE difobrasd(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difobrasd'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    nombre TEXT NOT NULL,  
    sede TEXT,
```



```
recinto TEXT,  
asistentes INT,  
fechaini DATE,  
fechafin DATE,  
funciones INT);
```

```
CREATE SEQUENCE seq_difobrasd;
```

```
GRANT ALL ON difobrasd TO nobody;  
GRANT ALL ON seq_difobrasd TO nobody;
```

```
CREATE TABLE difobrasf(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difobrasf'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  nombre TEXT NOT NULL,  
  sede TEXT,  
  recinto TEXT,  
  asistentes INT,  
  fechaini DATE,  
  fechafin DATE,  
  funciones INT);
```

```
CREATE SEQUENCE seq_difobrasf;
```

```
GRANT ALL ON difobrasf TO nobody;  
GRANT ALL ON seq_difobrasf TO nobody;
```

```
CREATE TABLE difobrast(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difobrast'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  nombre TEXT NOT NULL,  
  sede TEXT,  
  recinto TEXT,  
  asistentes INT,  
  fechaini DATE,  
  fechafin DATE,  
  funciones INT);
```

```
CREATE SEQUENCE seq_difobrast;
```

```
GRANT ALL ON difobrast TO nobody;  
GRANT ALL ON seq_difobrast TO nobody;
```

```
CREATE TABLE difvisitasg(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difvisitasg'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  nombre TEXT NOT NULL,  
  sede TEXT,  
  recinto TEXT,  
  asistentes INT,  
  fechaini DATE,  
  fechafin DATE);
```

```
CREATE SEQUENCE seq_difvisitasg;

GRANT ALL ON difvisitasg TO nobody;
GRANT ALL ON seq_difvisitasg TO nobody;

CREATE TABLE difexposiciones (
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difexposiciones'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_difexposiciones;

GRANT ALL ON difexposiciones TO nobody;
GRANT ALL ON seq_difexposiciones TO nobody;

CREATE TABLE differias (
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_differias'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_differias;

GRANT ALL ON differias TO nobody;
GRANT ALL ON seq_differias TO nobody;

CREATE TABLE diffestivales (
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_diffestivales'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_diffestivales;

GRANT ALL ON diffestivales TO nobody;
GRANT ALL ON seq_diffestivales TO nobody;

CREATE TABLE difjornadas (
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difjornadas'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
```

```
sede TEXT,
recinto TEXT,
asistentes INT,
fechaini DATE,
fechafin DATE);

CREATE SEQUENCE seq_difjornadas;

GRANT ALL ON difjornadas TO nobody;
GRANT ALL ON seq_difjornadas TO nobody;

CREATE TABLE difmesasr(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difmesasr'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);
CREATE SEQUENCE seq_difmesasr;

GRANT ALL ON difmesasr TO nobody;
GRANT ALL ON seq_difmesasr TO nobody;

CREATE TABLE difperform(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difperform'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fechaini DATE,
  fechafin DATE);

CREATE SEQUENCE seq_difperform;

GRANT ALL ON difperform TO nobody;
GRANT ALL ON seq_difperform TO nobody;

CREATE TABLE diflecturas(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_diflecturas'),
  dependencia INT NOT NULL REFERENCES dependencias
  ON UPDATE CASCADE ON DELETE CASCADE,
  nombre TEXT NOT NULL,
  sede TEXT,
  recinto TEXT,
  asistentes INT,
  fecha DATE,
  tipolectura INT NOT NULL DEFAULT 0 REFERENCES tiposlectura
  ON UPDATE CASCADE ON DELETE SET DEFAULT );

CREATE SEQUENCE seq_diflecturas;

GRANT ALL ON diflecturas TO nobody;
GRANT ALL ON seq_diflecturas TO nobody;
```

```
CREATE TABLE difprespublic(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_difprespublic'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    nombre TEXT NOT NULL,  
    sede TEXT,  
    recinto TEXT,  
    asistentes INT,  
    fecha DATE,  
    tipopresentacion INT NOT NULL DEFAULT 0 REFERENCES tipospresentacion  
    ON UPDATE CASCADE ON DELETE SET DEFAULT );
```

```
CREATE SEQUENCE seq_difprespublic;
```

```
GRANT ALL ON difprespublic TO nobody;  
GRANT ALL ON seq_difprespublic TO nobody;
```

-- Tablas para el módulo de Actividades de comunicación

```
CREATE TABLE comcapsulas(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_comcapsulas'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    titulo TEXT NOT NULL,  
    duracion REAL,  
    emisiones INT,  
    medio TEXT,  
    fechaini DATE,  
    fechafin DATE,  
    coproduccion BOOLEAN);
```

```
CREATE SEQUENCE seq_comcapsulas;
```

```
GRANT ALL ON comcapsulas TO nobody;  
GRANT ALL ON seq_comcapsulas TO nobody;
```

```
CREATE TABLE comprogramas(  
    id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_comprogramas'),  
    dependencia INT NOT NULL REFERENCES dependencias  
    ON UPDATE CASCADE ON DELETE CASCADE,  
    titulo TEXT NOT NULL,  
    duracion REAL,  
    emisiones INT,  
    medio TEXT,  
    fechaini DATE,  
    fechafin DATE,  
    coproduccion BOOLEAN);
```

```
CREATE SEQUENCE seq_comprogramas;
```

```
GRANT ALL ON comprogramas TO nobody;  
GRANT ALL ON seq_comprogramas TO nobody;
```

```
CREATE TABLE promocionales(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_promocionales'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  titulo TEXT NOT NULL,  
  duracion REAL,  
  emisiones INT,  
  medio TEXT,  
  fechaini DATE,  
  fechafin DATE,  
  coproduccion BOOLEAN);
```

```
CREATE SEQUENCE seq_promocionales;
```

```
GRANT ALL ON promocionales TO nobody;  
GRANT ALL ON seq_promocionales TO nobody;
```

-- Tablas para el módulo de Actividades de educación continua

```
CREATE TABLE modalidades(  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO modalidades VALUES(0,'No especificada');  
INSERT INTO modalidades VALUES(1,'Presencial');  
INSERT INTO modalidades VALUES(2,'Teleconferencia');  
INSERT INTO modalidades VALUES(3,'Videoconferencia');  
INSERT INTO modalidades VALUES(4,'Multimedia');  
INSERT INTO modalidades VALUES(5,'Internet');  
INSERT INTO modalidades VALUES(6,'Otro');
```

```
GRANT SELECT ON modalidades TO nobody;
```

```
CREATE TABLE tiposactividad(  
  id INT PRIMARY KEY NOT NULL,  
  nombre TEXT NOT NULL);
```

```
INSERT INTO tiposactividad VALUES(0,'No especificada');  
INSERT INTO tiposactividad VALUES(1,'Curso');  
INSERT INTO tiposactividad VALUES(2,'Taller');  
INSERT INTO tiposactividad VALUES(3,'Seminario');  
INSERT INTO tiposactividad VALUES(4,'Otra');
```

```
GRANT SELECT ON tiposactividad TO nobody;
```

```
CREATE TABLE acteducontinua(  
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_acteducontinua'),  
  dependencia INT NOT NULL REFERENCES dependencias  
  ON UPDATE CASCADE ON DELETE CASCADE,  
  nombre TEXT NOT NULL,  
  modulos INT,  
  asistentes INT,  
  constancias INT,  
  ponentes INT,
```

```
becados INT,
sede TEXT,
duracion TEXT,
diplomado BOOLEAN,
modalidad INT NOT NULL DEFAULT 0 REFERENCES modalidades
ON UPDATE CASCADE ON DELETE SET DEFAULT,
Tipoactividad INT NOT NULL DEFAULT 0 REFERENCES tiposactividad
ON UPDATE CASCADE ON DELETE SET DEFAULT,
fechaini DATE,
fechafin DATE,
comentarios TEXT);

CREATE SEQUENCE seq_acteducontinua;

GRANT ALL ON acteducontinua TO nobody;
GRANT ALL ON seq_acteducontinua TO nobody;

-- Tablas para el módulo de Premios y distinciones recibidos

CREATE TABLE premiosrec(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_premiosrec'),
  dependencia INT NOT NULL REFERENCES dependencias
ON UPDATE CASCADE ON DELETE CASCADE,
nombre TEXT NOT NULL,
academico TEXT NOT NULL,
rfcacad TEXT,
institucion TEXT,
fecha DATE);

CREATE SEQUENCE seq_premiosrec;

GRANT ALL ON premiosrec TO nobody;
GRANT ALL ON seq_premiosrec TO nobody;

-- Tablas para el módulo de Premios y distinciones otorgados

CREATE TABLE premiosotor(
  id INT PRIMARY KEY NOT NULL DEFAULT nextval('seq_premiosotor'),
  dependencia INT NOT NULL REFERENCES dependencias
ON UPDATE CASCADE ON DELETE CASCADE,
nombre TEXT,
academico TEXT NOT NULL,
rfcacad TEXT,
institucion TEXT,
fecha DATE);

CREATE SEQUENCE seq_premiosotor;

GRANT ALL ON premiosotor TO nobody;
GRANT ALL ON seq_premiosotor TO nobody;
```

## Glosario

### Apache

Es el nombre del servidor web más popular en Internet. El nombre de Apache proviene de las épocas en que Apache era sólo un conjunto de parches para el NCSA HTTPD, y era comúnmente denominado A PATChy sErver.

### Aplicación web

Una aplicación web es cualquier aplicación que utiliza el Protocolo HTTP como principal protocolo de transporte y de intercambio de información.

### API

Siglas de Interfaz de Programación de Aplicaciones ( *Application Programming Interface* ). Es un conjunto estándar de funciones y formatos de datos utilizados comúnmente en el desarrollo de programas para computadora.

### Autenticación

Se llama así al proceso de validación de la conexión de un usuario que determina el permiso de acceso a los recursos de un servidor.

### Base de datos

Conjunto de datos relacionados entre sí. Por datos se entiende hechos conocidos que pueden registrarse y que tienen un significado implícito. La información en una base de datos está típicamente dividida en registros, y éstos, a su vez, en campos.

### CGI

Siglas de interfaz de pasarela común ( *Common Gateway Interface* ). Es un estándar que describe la forma en que los servidores web pueden comunicarse con programas externos. Los programas que utilizan la especificación CGI se suelen llamar programas CGI o scripts CGI. Las aplicaciones CGI permiten acceder a otras fuentes de información a las que no se tiene acceso directo mediante el web, tales como sistemas de bases de datos.

### DBMS

Siglas de sistema administrador de bases de datos ( *Database Management System* ). Es un conjunto de programas empleado para definir, administrar y procesar las bases de datos y sus aplicaciones.

### Distribución

Linux es el núcleo del sistema operativo, el kernel. Linux y su colección de paquetes de software se conocen formalmente con el nombre de distribución. Debido a que Linux es de carácter libre, no hay ninguna organización concreta que asuma la normalización de sistema, lo que ocasiona que existan una gran variedad de distribuciones, sin embargo la mayoría de las distribuciones más populares tienen básicamente las mismas aplicaciones y el mismo kernel.

**Firewall**

Traducido al castellano como cortafuegos, es un dispositivo de seguridad establecido en hardware o software que permite restringir el acceso a una red de computadoras, principalmente de los intrusos de Internet.

**GUI**

Siglas de interfaz gráfica de usuario ( *Graphical User Interface* ). Ésta describe el método de interacción que un programa ofrece al usuario. Una GUI permite controlar una aplicación mediante ratón y teclado, al contrario de una línea de comandos que requiere entrada mediante teclado solamente.

**HTML**

Siglas de lenguaje de marcado de hipertexto ( *Hypertext Markup Language* ). Es el lenguaje utilizado para crear páginas web. Los documentos HTML son archivos de texto que contienen algunas etiquetas que indican a los navegadores como desplegar y formatear el contenido del documento, así como presentar los enlaces hacia otras páginas.

**HTTP**

Siglas de protocolo de transferencia de hipertexto ( *HyperText Transfer Protocol* ). Éste es un protocolo de nivel de aplicación, genérico, orientado a objetos y sin estados que permite el intercambio de información hipermedia entre un navegador y un servidor web.

**Internet**

Red mundial de computadoras que se comunican por un protocolo común de comunicaciones y que facilita servicios de comunicación de datos como conexiones remotas, transferencia de archivos, correo electrónico y grupos de noticias.

**Login**

Es el nombre de usuario para el acceso a una red, a un sistema operativo multiusuario, o a un sistema de información.

**Navegación**

Se refiere a la forma de ir de una pantalla a otra dentro de un sistema.

**Navegador**

Es un programa utilizado para explorar los recursos en Internet, principalmente los relacionados al contenido de páginas web.

**PHP**

Siglas de preprocesador de hipertexto ( *Hypertext Preprocessor* ). PHP es un lenguaje de programación *Open Source* que ofrece una solución simple y completa para la creación de aplicaciones que requieran del manejo de información de manera dinámica e interactiva a través del web. PHP es un lenguaje de script rápido, estable, simple, configurable, independiente de la plataforma, y que se ejecuta del lado del servidor.

**Postgres**

Es el nombre que se suele utilizar para nombrar a PostgreSQL. De las opciones *Open Source*, PostgreSQL es el manejador de bases de datos objeto-relacional más robusto.



**Red Hat**

Es el nombre de una de las empresas dedicadas a la distribución de Linux. La distribución se llama igualmente Red Hat. Hablar hoy en día de Red Hat es hablar no sólo de uno de los más clásicos y veteranos sistemas Linux, sino también de una de las mayores, más completas y más fiables distribuciones.

**Servidor**

En general, se refiere a una computadora que proporciona recursos compartidos a los usuarios de una red. En algunos casos específicos relacionados con Internet, el servidor se refiere al proceso que está en ejecución constante en segundo plano a la espera de peticiones por parte de los clientes. Tal es el caso de los servidores web, los servidores de correo, los servidores de ftp, etc.

**Servidor web**

Un servidor web es un programa que atiende las peticiones enviadas por un navegador, e independientemente de cuál haya sido esta petición, lo que el servidor web le manda como respuesta en la gran mayoría de los casos es un documento HTML.

**Software libre**

Al hablar de software libre, se hace en el sentido más filosófico la palabra libertad. Se habla de la libertad de tener un programa completo, incluyendo su código fuente, la libertad de usarlo, de copiarlo, de modificarlo, de venderlo y de compartirlo.

**SQL**

Siglas de lenguaje de consulta estructurado (*Structured Query Language*). Es un lenguaje que se utiliza para definir la estructura y el procesamiento de una base de datos relacional. Se emplea como un lenguaje de consulta único o puede incorporarse en programas de aplicación.

**WWW**

*World Wide Web*, el web, es uno de los principales servicios de información que se pueden encontrar en Internet. El web es una forma más de acceder a la información que se encuentra en Internet, información que en su gran mayoría se compone de documentos HTML, comúnmente llamados páginas web.

**URL**

Siglas de Localizador Uniforme de Recursos (*Uniform Resource Locator*). Uno de los dos tipos básicos de identificadores universales de recursos, y consta de una cadena de caracteres que identifica con precisión el tipo y localización de un recurso de Internet.

## Bibliografía

Dirección General de Estadística y Desarrollo Institucional  
*Memorias del V encuentro internacional de responsables de información estadística de las instituciones de educación superior.*  
México, 2001.

Kenneth E. Kendall y Julie E. Kendall.  
*Análisis y diseño de sistemas.*  
Prentice-Hall, México, 1997.

Centro de Computación Profesional de México  
*Análisis y diseño de sistemas*  
McGraw-Hill, México, 2001.

David M. Kroenke  
*Procesamiento de bases de datos*  
Prentice-Hall, México, 1996.

Héctor Facundo Arena  
*Desarrollo de sitios web bajo Linux*  
MP Ediciones, Argentina, 2001.

Tobias Ratschiller y Till Gerken  
*Creación de aplicaciones web con PHP 4*  
Pearson Educación, España, 2001.

José Luis Raya Cabrera, José Antonio Moreno Gutiérrez y Antonio López Sastre  
*HTML 4 guía de referencia y tutorial*  
Alfaomega grupo editor, México, 1999.

Rainer Kolbeck  
*El gran libro de JavaScript*  
Marco Polo, España, 1997.

## Sitios de interés

**Apache** <http://www.apache.org>

El sitio web del servidor http más popular de los últimos tiempos. Contiene información acerca del servidor, la fundación Apache y otros proyectos relacionados.

**PostgreSQL** <http://www.postgresql.org>

Es el sitio oficial de PostgreSQL.

**PHP** <http://www.php.net>

El sitio oficial de PHP. En éste se encuentra la última versión estable disponible y documentación en diversos formatos y en diversas traducciones.

**PHP México** <http://www.php.org.mx>

Es el sitio web del grupo de usuarios de PHP en México. En este sitio se puede encontrar ayuda, asesoría, noticias, una lista de correo y mucha más información relacionada con PHP completamente en castellano. Está enfocado principalmente a la comunidad mexicana de usuarios de PHP.

**Linux** <http://www.linux.org>

Uno de los tantos sitios acerca de Linux. En éste en particular, se encuentran revisiones de libros, comentarios acerca de las distribuciones y una cantidad muy grande de enlaces relacionados con el proyecto Linux.

**Linux Documentation Project** <http://www.linuxdoc.org>

El sitio oficial del proyecto de documentación de Linux. Aquí se encuentran artículos, tutoriales, preguntas frecuentes, páginas de manual, etc.

**INSFLUG** <http://www.insflug.org>

Este es un proyecto que se dedica a la traducción al castellano de los HOWTO's del proyecto oficial de documentación de Linux. Los HOWTO's (COMO's en castellano) son documentos que explican cómo realizar ciertas tareas paso a paso.

**Proyecto Lucas** <http://lucas.ctv.es>

Lucas es un proyecto que se dedica a la realización y compilación de documentos explicativos de Linux en castellano. En este sitio se encuentran cientos de artículos, libros, preguntas frecuentes y COMO's absolutamente en castellano y totalmente libres.

**Linuxpreview** <http://www.linuxpreview.org>

Un sistema de noticias actualizadas a cada instante sobre el mundo Linux y otras tecnologías. Se incluyen además comentarios de los visitantes acerca de las noticias publicadas.

**Source Forge** <http://www.sourceforge.net>

Este sitio intenta servir de apoyo a los desarrolladores de aplicaciones de software libre. Provee todos los elementos necesarios para montar, desarrollar y administrar un proyecto.

**Red Hat** <http://www.redhat.com>  
Es el sitio oficial de Linux Red Hat

**RPM find** <http://www.rpmfind.net>  
Se encuentran paquetes RPMs para Linux Red Hat.

**W3 consortium** <http://www.w3.org>  
El consorcio del WWW engloba a varias instituciones y promueve la elaboración y discusión de los estándares y tecnologías relacionadas con el web. En este sitio se encuentran las especificaciones para el HTML, el XML, CSS, etc.

**Estadística en la UNAM** <http://www.estadistica.unam.mx>  
Es el sitio web de la Dirección General de Estadística y Desarrollo Institucional.

TESIS CON  
FALLA DE ORIGEN