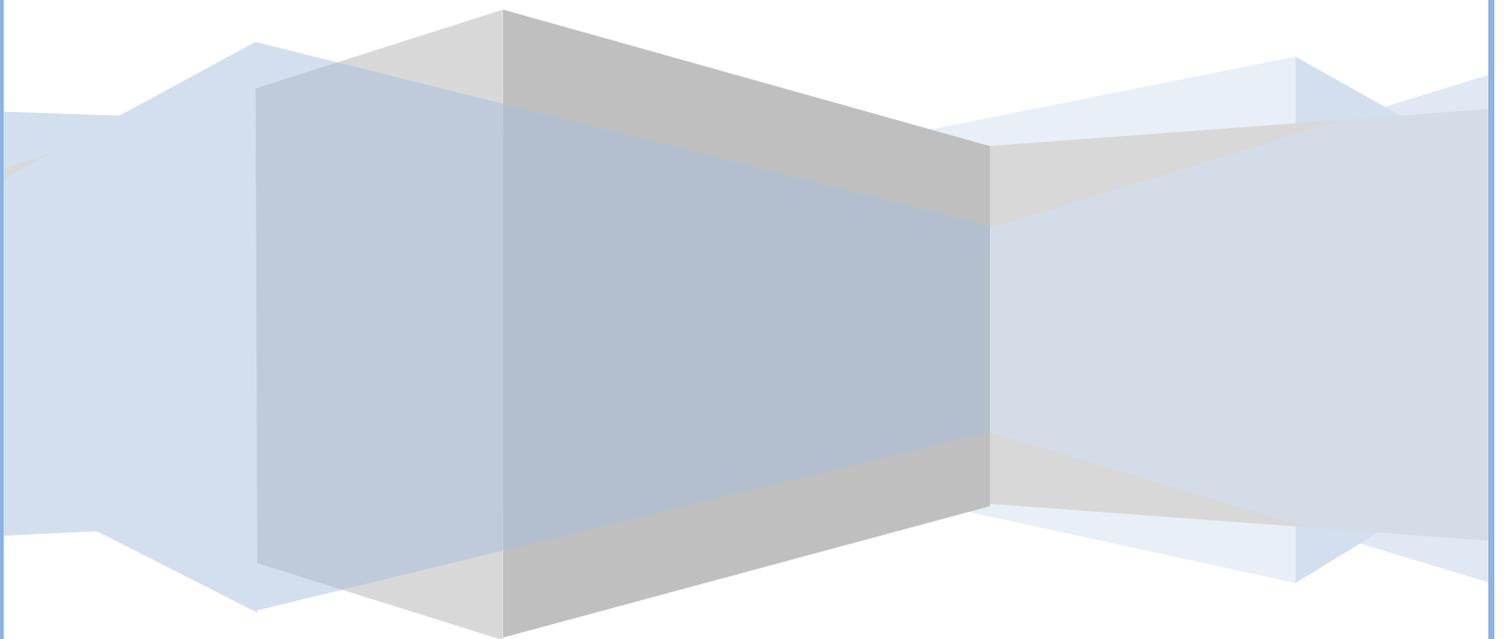


# Capítulo 3

## Implementación



# *CAPÍTULO 3*



## *3.1 INTRODUCCIÓN*

En la etapa de Implementación se dan a conocer los puntos principales para la creación del sistema, partiendo de las especificaciones de hardware junto con el sistema operativo que se utilizará, de acuerdo a los recursos con los que cuenta en la División de Ingeniería en Ciencias de la Tierra (DICT) en el momento de plantear el problema. Además de explicar la creación de la Base de Datos junto con la interfaz de usuario.

De acuerdo a lo explicado en el capítulo anterior referente al software utilizado y diseño planteado en base a los requerimientos realizados, se utilizó como sistema operativo Windows Server 2003, además de que la implementación se realizó con un lenguaje de programación PHP 5, manejador de Base de Datos MySQL y con el servidor web Apache 5.

A continuación se detallara más esta etapa abarcando las herramientas utilizadas, así como el tipo de seguridad manejado en el sistema.

**3.1.1 ESPECIFICACIONES DEL HARDWARE**

En la instalación de Windows Server 2003, fue necesario conocer los requerimientos mínimos de hardware previos a la instalación del Sistema Operativo, todo esto con el fin de evitar fallas, tanto en la implementación del proyecto como en su uso. Las características principales son las siguientes:

	Windows Server 2003 Web Edition	Windows Server 2003 Standard Edition	Windows Server 2003 Enterprise Edition	Windows Server 2003 Datacenter Edition
Nº Máximo de procesadores soportados	2	4	8	64
Memoria máxima	2 GB	4 GB	32 GB (32 bit) 64 GB (64 bit)	32 GB (32 bit) 512 GB (64 bit)
Requerimientos del sistema recomendados	Procesador a 550 Mhz, 256 MB RAM, 1.5 GB de espacio en disco	Procesador a 550 Mhz, 256 MB RAM, 1.5 GB de espacio en disco	Procesador a 550 Mhz, 256 MB RAM, 1.5-2.0 GB de espacio en disco	Procesador a 550 Mhz, 1 GB RAM, 1.5 GB de espacio en disco

Teniendo en cuenta las características enunciadas anteriormente se tomó la decisión de tener el servidor con la versión de Windows Server 2003 Standard Edition y en una computadora con las siguientes características:

DISPOSITIVO	CARACTERÍSTICAS
Marca	DELL
Procesador	Intel Core 2, 1.86 GHz
Memoria RAM	1.00 GB
Disco duro	120 GB

### 3.1.2 SEGURIDAD DEL SERVIDOR

La instalación del sistema operativo Windows Server se hizo satisfactoriamente basándose en los manuales de operación y manuales publicados en la red. La seguridad del servidor se basó en la que tiene Apache y la programación que se utilizó en el sistema dado que se utiliza un usuario para cada uno de los diferentes bloques del sistema, esto con el fin de que no se puedan descifrar con facilidad las cuentas de usuario junto con sus contraseñas para proporcionar a la información mayor seguridad. Para la creación de la Base de Datos la configuración corrió a cargo de PHP y MySQL. Dando de esta forma por terminado la parte de la configuración para la seguridad del servidor y con ello la seguridad del sistema.

## 3.2 CREACIÓN DE LA BASE DE DATOS

El funcionamiento del sistema parte de una base de datos llamada control, como se mencionó anteriormente en el capítulo 2 de esta tesis, la cual ayuda a administrar e identificar a los usuarios que ingresen al mismo. A través de ésta el sistema reconoce el tipo de usuario registrado para con ello proporcionarle los distintos tipos de consulta existentes, ya sea de la base Principal o la Confidencial.

Posteriormente, dependiendo del tipo de privilegios con los que cuenten los usuarios registrados, se podrá manipular la base llamada Confidencial, en la cual podrán consultar, modificar, insertar o eliminar tablas de la misma.

La forma en que el sistema interactúa con esta Base de Datos se explica a continuación (tabla 3.1)

Tabla 3.1 Código para la conexión del sistema a la BD

<b>Código</b>	<b>Descripción</b>
<pre> &lt;?PHP //Definicion objeto ConexionDB  class ConexionBD {      protected \$ENLACE;     protected \$BASE;      //informacion de acceso a la base     private \$USR;     private \$PW;     private \$HOST;      function __construct(\$BD,\$Usuario,\$Password)     {         \$this-&gt;HOST = "localhost";         \$this-&gt;USR = \$Usuario;         \$this-&gt;PW = \$Password;         \$this-&gt;Conectar(\$BD);     }      private function Conectar(\$BD)     {         \$this-&gt;BASE = \$BD;          if      (!(\$this-&gt;ENLACE=mysql_connect(\$this-&gt;HOST,\$this-&gt;USR,\$this-&gt;PW))) //validacion de conexion             die("Error conectando a la base de datos.");         else    if(!mysql_select_db(\$this-&gt;BASE,\$this-&gt;ENLACE)) //validacion a la base de datos             die("Error seleccionando la base de datos.");     }      function Get_Link()     {         return \$this-&gt;ENLACE;     }      function __destroy()     {         if(!mysql_close(\$this-&gt;ENLACE)) echo "ERROR en la desconexion de la base.";     } } ?&gt; </pre>	<p>En esta parte del sistema lo que hace el código es la conexión a la base de datos utilizando las variables estándar del manejador de Base de Datos utilizado (phpMyAdmin)</p> <p>Esta conexión se realizó mediante programación orientada a objetos de dichas variables utilizando el lenguaje de programación elegido (php)</p>

### 3.3 CREACIÓN DE LA INTERFAZ DE USUARIO

La parte fundamental del sistema es la interfaz de usuario dado que es la parte visual y la que el usuario utiliza, la interfaz está dividida de tal forma que sea fácil para el usuario su utilización en la inserción y consulta de información (tabla 3.2).

El acceso al Sistema (tabla 3.3) dependerá de los privilegios que se tenga como usuario, dado que las personas que no cuentan con privilegios en el Sistema solo podrán visualizar la información básica del personal de la DICT alojada en la Base de Datos.

La consulta, dependerá de la forma en la que se acceda al Sistema esto es, si no se cuentan con privilegios solo se podrá tener una consulta pública, pero si se cuentan con privilegios se podrá tener una consulta completa, además de poder insertar nueva información o mas información de las personas que existentes en la Base, también se puede eliminar la información que así se requiera

Tabla 3.2 Código de la librería de funciones	
<b>Código</b>	<b>Descripción</b>
<pre> &lt;? /* Libreria de funciones complementarias al Sistema de Gestion de Personal */  /*      Inicion de sesion y declaracion de variables de sesion      */ session_start();  if(!session_is_registered('Sesion_Act'))     {         session_register('Sesion_Act');         \$_SESSION['Sesion_Act']=true;     }  if(!session_is_registered("TipoUSR"))     {         session_register("TipoUSR");         \$_SESSION["TipoUSR"]="C";     }  if(!session_is_registered('AccesoSist'))     {         session_register('AccesoSist');         \$_SESSION['AccesoSist'] = false;     } </pre>	<p>La librería de funciones es la parte del código del sistema que permite la seguridad del mismo ya que cada módulo implementado manda llamar las funciones que necesite para dar la funcionalidad a lo requerido en el sistema.</p>

```

/*          Funcion de autocarga de clases          */

function __autoload($Class)
{
    $Class_file="Obj/Class_".$Class.".php";

    if(file_exists($Class_file)) require_once $Class_file;
    else die(sprintf("Archivo no encontrado %s",$Class_file));

}

/*          funciones */

function ValidacionUSR($usr,$pw)
{
    if(empty($usr)or empty($pw))
    {
        $val=false;
    }
    else
    {
        $q = new ConsultaBD("personal_dict");

        $SQL = sprintf("SELECT TAcSGP FROM
admusr WHERE NomUsr = '%s' AND Password =
PASSWORD('%s')",$usr,$pw);

        $q->consulta($SQL);

        if($q->RenAfec > 0)
        {
            $val=true;
            $_SESSION['TipoUSR']=$q-
>Array_Reg[0]->TAcSGP;
        }
        else
        {
            $val=false;
        }
    }

    return $val;
}

function ValidarInfoUSR($Datos)
{
    $valusr['val'] = false;

    $Con = new ConsultaBD("personal_dict");

```

<pre>                 \$SQL = sprintf("SELECT `NomUsr` FROM `admusr` WHERE `NomUsr` = '%s'", \$Datos[1]);                  \$Con-&gt;consulta(\$SQL);                  \$Con-&gt;__destroy();                  if(\$Con-&gt;NumReg &gt; 0)                 {                          printf("&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;label class='ERROR' &gt;El nombre de usuario %s ya existe seleccione otro.&lt;/label&gt;", \$Datos[1]);                         \$valusr[0] = "class='incorrecto'";                         \$valusr['val'] = true;                 }                 else if(strlen(\$Datos[1]) &lt; 8)                 {                         printf("&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;label class='ERROR' &gt;El nombre de usuario (%s) no tiene una longitud de 8 caracteres.&lt;/label&gt;", \$Datos[1]);                         \$valusr[0] = "class='incorrecto'";                         \$valusr['val'] = true;                 }                  if(\$Datos[2] != \$Datos[3])                 {                         printf("&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;label class='ERROR' &gt;La Contraseña no coincide con la de confirmación.&lt;/label&gt;");                         \$valusr[1] = "class='incorrecto'";                         \$valusr['val'] = true;                 }                 else if(strlen(\$Datos[2]) &lt; 8)                 {                         printf("&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;label class='ERROR' &gt;La Contraseña no tiene una longitud de 8 caracteres.&lt;/label&gt;");                         \$valusr[1] = "class='incorrecto'";                         \$valusr['val'] = true;                 }                  if(\$Datos[4] == "N")                 {                         printf("&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;label class='ERROR' &gt;No a seleccionado un tipo de usuario.&lt;/label&gt;");                         \$valusr[2] = "class='incorrecto'";                         \$valusr['val'] = true;                 } </pre>	
---	--

<pre> return \$valor; }  function MensajeAlerta(\$mensaje) { echo "&lt;script language=javascript&gt;"; printf("window.alert('%s');", \$mensaje); echo "&lt;/script&gt; "; }  function insertar_Acc_Sist(\$Datos) { \$ObjIns = new InsertBD("personal_dict"); \$Valores = sprintf("NULL , %d, '%s', PASSWORD( '%s' ) , '%s'", \$Datos[0], \$Datos[1], \$Datos[2], \$Datos[3]); \$ObjIns-&gt;insertar("admusr", \$Valores); \$ObjIns-&gt;__destroy(); }  function validacionPrivilegios(\$ide) { \$validacion = false;  if(\$_SESSION['TipoUSR'] == "A") \$validacion = true; else { \$InfPer = new ConsultaBD("personal_dict");  \$SQL = sprintf("SELECT `Tipoc` FROM `inf_laboral`, `departamento` WHERE `inf_laboral`.`Id_Depto` = `departamento`.`Id_Depto` AND Id_persona = %d ", \$ide);  \$InfPer-&gt;Consulta(\$SQL);  if(\$_SESSION['TipoUSR'] == \$InfPer- &gt;Array_Reg[0]-&gt;Tipoc) \$validacion = true;  \$InfPer-&gt;__destroy(); }  return \$validacion; } ?&gt; </pre>	
--	--

Tabla 3.3 Código del módulo de seguridad para el acceso al sistema

<b>Código</b>	<b>Descripción</b>
<pre> &lt;? class AccesoUSR extends QueryBD {     public \$TipoUsr;      private \$SesionActiva = true;     //private \$NomUsr;      function __construct(\$Enlace)     {         parent::__construct(\$Enlace);         \$this-&gt;SesionActiva=\$_SESSION['objeto'];         \$this-&gt;TipoUsr="comun";     }      function Login()     {         if(\$this-&gt;SesionActiva)         {             \$this-&gt;SesionActiva = false;             header('WWW-Authenticate: Basic realm="Acceso restringido");             header('HTTP/1.0 401 Unauthorized');             echo '- Authorization Required -';             exit;         }         else if(ValidacionUSR())         {             echo "Ha introducido el nombre de usuario: \$PHP_AUTH_USER&lt;br&gt;";             echo "Ha introducido la contraseña: \$PHP_AUTH_PW&lt;br&gt;";         }         else         {             echo "Nombre de usuario o contraseña incorrectos, porfavor intente de nuevo!!";         }     }      private function ValidacionUSR()     { </pre>	<p>Este módulo permite identificar el tipo de usuario registrado, para con ello poder proporcionar el tipo de información permitida dependiendo los privilegios con los que cuenta dicho usuario.</p>

```

        $this->Query("SELECT TAcSGP FROM
admusr WHERE NomUsr = ".$PHP_AUTH_USER." AND
Password = PASSWORD(".$PHP_AUTH_PW.");

        if($this->RenAfec > 0)
        {
            $val = true;
            $tipo = mysql_fetch_array($this-
>Result);
            $this->TipoUsr = $tipo[0];
        }
        else $val = false;

        return $val;
    }

    function __destroy()
    {
        parent::__construct($Enlace);
        $_SESSION['objeto']=$this->SesionActiva;
        $this->TipoUsr="comun";
    }

}

?>

```

### 3.3.1 TIPOS DE USUARIO

El Sistema está diseñado para contemplar tres tipos de usuarios: administrador, propietarios y cualquiera. Cada uno tendrá distintos privilegios para poder manipular el sistema:

- Administrador: El Administrador tiene a su disposición la funcionalidad total del sistema, con la capacidad de consultar la información completa de cualquier persona de la base así como modificar o eliminar al personal, también tiene la facultad de administrar a los usuarios que tengan acceso al sistema.

- Propietario: El propietario puede consultar la información completa de cualquier persona pero sólo puede agregar, modificar y eliminar al personal que esté en su departamento, para esto existen 4 tipos de propietarios:
  - P – Petrolera
  - G – Geología
  - F – Geofísica
  - M – Minas
- Cualquiera: Un usuario cualquiera puede acceder al sistema sin poder realizar ningún tipo de modificación al mismo; no puede agregar o eliminar personal, solamente puede observar cierta información de los trabajadores.

El no poder agregar personal de otro departamento que no sea el suyo, se debe a que cada propietario sólo tiene la posibilidad de manejar a su propio personal y no el de los demás, esto se manipula de esta manera por seguridad de cada departamento.

### **3.3.2 TIPOS DE CONSULTA**

La consulta en el Sistema se divide en dos tipos, esto debido a que no toda la información puede ser vista por todos los usuarios que ingresen al sistema, los tipos de consulta son: consulta pública y consulta completa.

#### ***Consulta pública***

En la figura 3.1 la sección marcada hace mención a la consulta, ésta consiste en buscar a personas que trabajan en la DICT (División de Ingeniería en Ciencias de la Tierra) y que se encuentran registradas en el sistema. Este apartado se puede realizar de tres formas diferentes: por nombre, por número de trabajador y de una forma avanzada.

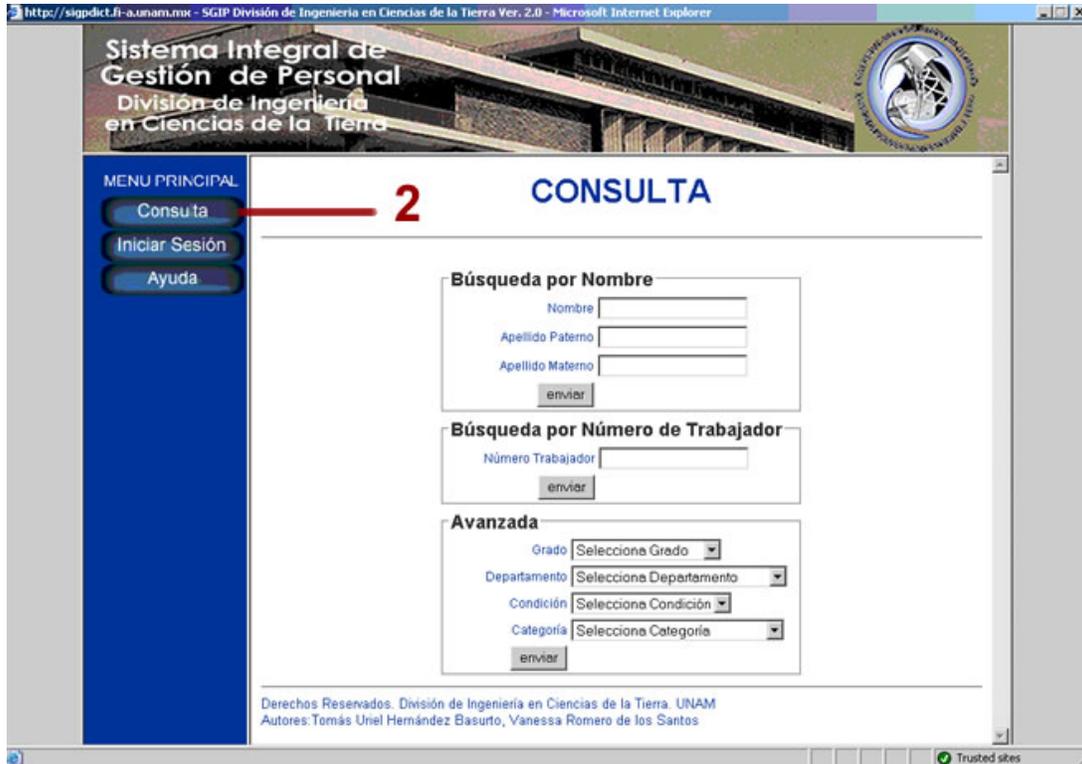


Figura 3.1 Consulta pública

- **Por Nombre:** La consulta se puede realizar por nombre, por apellido paterno o por apellido materno de la persona que se desea buscar en el sistema, la búsqueda se realiza escribiendo el nombre o apellidos de la persona deseada o simplemente introduciendo la letra inicial de cualquiera de los campos mencionados.
- **Búsqueda por Número de Trabajador:** Sólo se necesita escribir el número de trabajador de quien se desea buscar.
- **Avanzada:** Seleccionar una o más opciones del menú en cualquiera de estas categorías: Grado, Departamento, Condición o Categoría, con esto se desplegará la información de la(s) persona(s) que se encuentran en los campos seleccionados.

Se realiza la consulta al ingresar cualquiera de los datos anteriores, y se mostrará por resultado el desglose de los datos más significativos de la persona a quien se estaba buscando.

### Consulta completa

Primero se requiere iniciar una sesión (figura 3.2). Una vez iniciada, en la sección de lado izquierdo (figura 3.3) en el icono "Consulta completa" se puede consultar de una forma más detallada los datos de las personas que se desea buscar, y posteriormente se realizan los mismos pasos que en la consulta pública (figura 3.1): ingresar nombre, apellido paterno o ingresar algún otro campo conocido de la persona.



Figura 3.2 Inicio de sesión

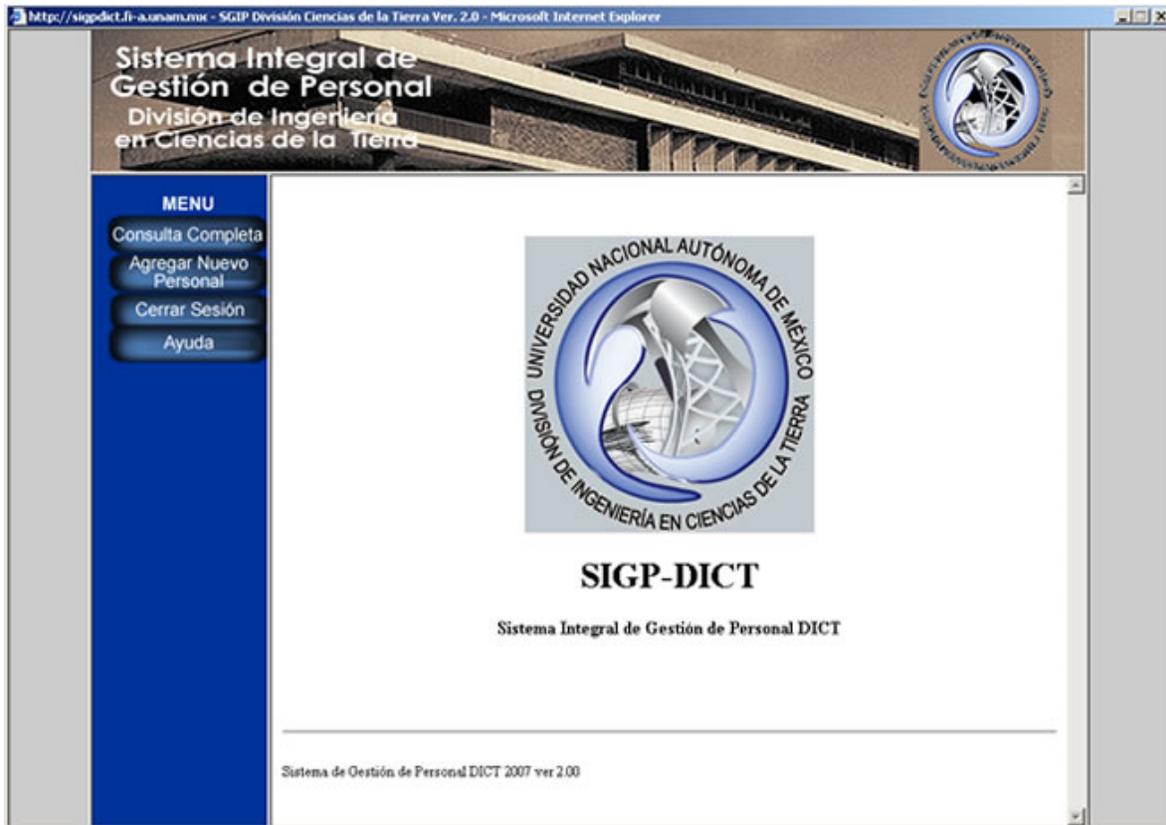


Figura 3.3 Interfaz principal del sistema.

Al realizar la consulta, el sistema desplegará toda la información simple de la persona requerida.

### 3.3.3 ALTA Y BAJA DEL PERSONAL

La figura 3.4 muestra las opciones que tiene el sistema para dar de alta y/o baja la información del personal registrado en el Sistema. Es importante señalar, como ya se escribió anteriormente, que ésta acción sólo la pueden realizar los propietarios a su respectivo personal o el administrador de la Base de Datos bajo las condiciones expuestas.

## CONSULTA

0 a 30 de 1

SEL	NOMBRE	DEPARTAMENTO	UBICACIÓN	CATEGORÍA Y GRADO	TELÉFONO	CORREO
<input checked="" type="radio"/>	Correa Fernández Tania	Explotación del Petróleo	Principal Ala A PB, 56	Investigador Asociado Maestría	2651 245 Ext 123	tania@hotmail.com

Derechos Reservados. División de Ingeniería en Ciencias de la Tierra. UNAM  
 Autores: Tomás Uriel Hernández Basurto, Vanessa Romero de los Santos

Figura 3.4 Opciones para dar de alta y/o baja

### Alta

Para dar de alta a una persona se tiene que seleccionar a la persona que se quiere dar de alta y después dar clic en “alta”, para comprobar dicha acción seleccione la opción “personal activo” y agregue algún campo (figura 3.5).

## CONSULTA

**Situación**

Personal activo.  Personal inactivo.  Ambos

**Búsqueda por Nombre**

Nombre

Apellido Paterno

Apellido Materno

**Búsqueda por Número de Trabajador**

Número Trabajador

**Avanzada**

Grado

Departamento

Condición

Categoría

Figura 3.5 Alta del personal

Para verificar la acción de alta basta con revisar en la parte superior de la página principal de consulta para ver que este seleccionada la parte de "Activos".

**CONSULTA**

---

Selección: Activos / Nombre con Tania / [Imprimir](#)

0 a 30 de 1

SEL	NOMBRE	DEPARTAMENTO	UBICACIÓN	CATEGORÍA Y GRADO	TELÉFONO	CORREO
6	Correa Fernández Tania	Explotación del Petróleo	Principal Ala A PB, 56	Investigador Asociado Maestría	2651245 Ext 123	tania@hotmail.com

[Imprimir](#)

---

**Figura 3.6** Confirmación de la alta del personal

### Baja

Para dar de baja a una persona sólo requiere dar clic en "baja" (figura 3.4), para verificar que esta acción se realizó con éxito podrá ir a la ventana inicial y ver las opciones en personal inactivo (figura 3.7). En la parte inferior del nombre se mostrará entre paréntesis el estado del trabajador: "(Baja)", de esta manera comprobará que el trabajador fue dado de baja (figura 3.8).

## CONSULTA

**Situación**

Personal activo
  Personal inactivo.
  Ambos

---

**Búsqueda por Nombre**

Nombre

Apellido Paterno

Apellido Materno

---

**Búsqueda por Número de Trabajador**

Número Trabajador

---

**Avanzada**

Grado

Departamento

Condición

Categoría

Figura 3.7 Selección del personal dado de baja

## CONSULTA

[Imprimir](#)

Selección: Inactivos / Nombre con Tania / 0 a 30 de 1

SEL	NOMBRE	DEPARTAMENTO	UBICACIÓN	CATEGORÍA Y GRADO	TELÉFONO	CORREO
<input checked="" type="checkbox"/>	Correa Fernández Tania (Baja)	Explotación del Petróleo	Principal Ala A PB, 56	Investigador Asociado Maestría	2651245 Ext 123	tania@hotmail.com

[Imprimir](#)

Figura 3.8 Verificación de la baja del personal

### 3.3.4 INSERCIÓN Y MODIFICACIÓN DEL PERSONAL

#### Inserción

Para agregar o registrar una persona nueva en el sistema se requiere de llenar dos formularios (figuras 3.9 y 3.10) con información del trabajador; dicha información es personal, laboral y académica. Existen campos que son obligatorios al llenar los formularios, es decir, es información mínima que se requiere para poder registrar a una persona dentro del sistema. Existe información mínima requerida, la cual se explico en el capítulo anterior, la cual debe ser proporcionada para poder registrar a una persona

**Información Personal**
HOJA 1

Nombre(s)  RCF

Apellido Paterno  CURP

Apellido Materno  Fecha de Nacimiento día  Mes  Año

Sexo

Nacionalidad  Mexicana  Extranjero:

**Dirección particular**

Calle y Numero

Colonia  C.P.

Delegación/Municipio  Estado

Teléfono Particular  Celular\*

**Direcciones Electrónicas**

Página Web\*

Correo electrónico 1

Correo electrónico 2\*

**Información Adicional**

Grado

Categoría

Categoría de Cómputo

Departamento

¿Se localiza dentro de la División?  Si  No

Figura 3.9 Formulario de inserción, primer hoja

**Información laboral**

HOJA 2

No. Trabajador \*\*\*  Cedula Prof.

Antigüedad

Nivel

Condición

Fecha de ingreso  día  Mes  Año

Ultimo Periodo de contratación  día  Mes  Año  al  día  Mes  Año

Fotografía:

**Nota:** El formato de la Fotografía puede ser jpg, bmp, gif y png, no deberá exceder de los 160KB.

Curriculum:

**Nota:** El formato del archivo tiene que ser en PDF y no exceder los 952KB.

---

**Información Académica**

Folio\*  Cod. Prog\*

Horas

Área de conocimiento en Licenciatura \*\*:

Área de conocimiento en Posgrado \*\*:

Asignaturas:

---

**Ubicación dentro de la DICT**

Edificio

Cubículo

Teléfono  Ext

Fax\*

Figura 3.10 Formulario de inserción, segunda hoja

## Modificación

Esta opción permite ir actualizando la información que se tenga del personal de la DICT como se muestra en la figura 3.11. Se debe tener en cuenta que al igual que la inserción, la modificación también presentara las dos páginas de información para su modificación (figura 3.12).

AdminROOT

---

[Imprimir](#)

Selección: Activos / Maestria / 0 a 30 de 2

SEL	Nombre (s)	Apellido paterno	Apellido materno	Departamento	Ubicación	Categoría	Grado	Teléfono	Correo
<input checked="" type="radio"/>	Thomas Eduardo	Perez	Garcia	Explotación del Petróleo	EdificioPrincipal Ala C 3er Piso, cubículo 113	Profesor Asignatura	Maestria	56223017	teperez@dictfi.unam.mx
<input type="radio"/>	Martin Carlos	Vidal	Garcia	Geología	EdificioPrincipal Ala A 400, cubículo 9	Técnico Académico Asociado	Maestria	56220851 Ext 109	martincarlosv@yahoo.com.mx

[Imprimir](#)

---

Derechos Reservados. División de Ingeniería en Ciencias de la Tierra. UNAM  
 Autores: Tomás Uriel Hernández Basurto, Vanessa Romero de los Santos

**Figura 3.11** *Modificación del personal*

http://132.248.54.197 - Actualizar - Microsoft Internet Explorer

**Modificaciones**

**Gonzalez Matrinez Julio Cesar**

**Información Personal**

Nombre(s)  RCF   
 Apellido Paterno  CURP   
 Apellido Materno  Fecha de Nacimiento     
 Sexo   
 Nacionalidad  Mexicana  Extranjero:

**Dirección particular**

Calle y Numero   
 Colonia  C.P.   
 Delegación/Municipio  Estado   
 Teléfono Particular  Celular\*

**Direcciones Electrónicas**

Pagina Web\*   
 Correo electrónico 1   
 Correo electrónico 2\*

**Información Adicional**

Grado   
 Categoría   
 Categoría de Cómputo   
 Departamento   
 ¿Se localiza dentro de la División?  Si  No

\* Estos campos son opcionales.  
 \*\* Esta información deberá ser llenada en caso de tener alguno.  
 \*\*\* Esta información es opcional únicamente para personal con condición externa o comisionados.

Done Trusted sites

Figura 3.12 Formularios para las modificaciones

### 3.4 EXPECTATIVAS DEL SISTEMA

Los sistemas diseñados y desarrollados necesitan una serie de pruebas para asegurar su buen funcionamiento de acuerdo a las necesidades definidas en el inicio del desarrollo y la definición de proyecto.

En este caso se define el inicio de prueba de caja blanca esto es, prueba de ruta que ayuda a saber la complejidad lógica del sistema.

#### Prueba de Caja Blanca

Filosofía de diseño de casos de prueba que usa estructura de control como parte del diseño a nivel de componentes para derivar casos de prueba. Esta filosofía puede derivar casos de prueba que:

- 1- garanticen que todas las rutas independientes de un módulo se revisaron una vez.
- 2- revisen las decisiones lógicas en los casos verdadero y falso.
- 3- ejecuten los bucles en las fronteras.
- 4- revisen estructuras de datos internas para garantiza la validez.

La **prueba de ruta o trayectoria básica** es una técnica de prueba de caja blanca, esta prueba permite derivar una medida de complejidad lógica de un diseño de procedimientos y usar dicha medida como medida para definir un conjunto básico de rutas de ejecución.

#### ➤ Notación de gráfico grafo de flujo

El método de ruta básica, debe tener una notación simple para la representación del flujo de control al cual se le denomina gráfico de flujo.

Para el sistema presentado en la tesis se tiene la siguiente figura que representa el diagrama del grafo de flujo (figura 3.13)

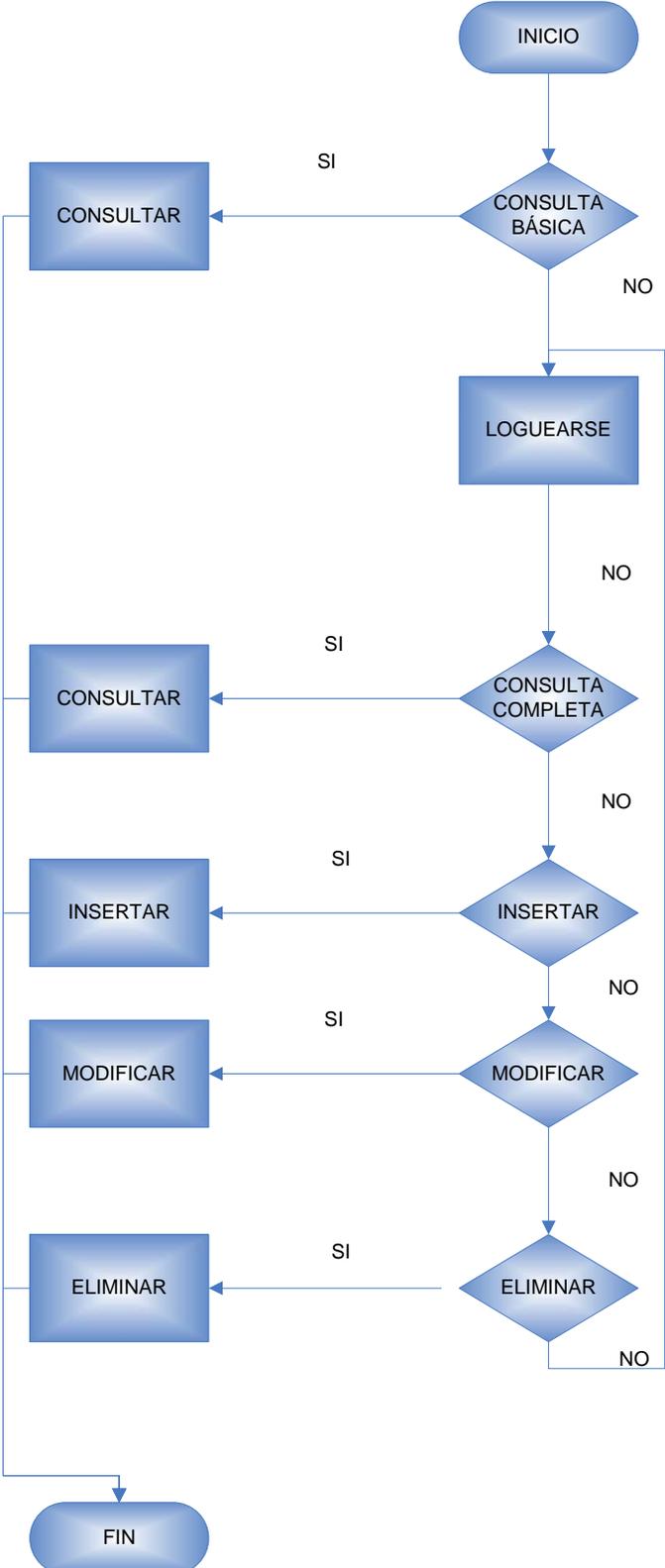


Figura 3.13 Diagrama de flujo del sistema

➤ Rutas de programa independientes

Una ruta independiente es cualquiera que introduce al menos un nuevo conjunto de enunciados de procesamiento o una nueva condición de programa. Cuando se establece un gráfico de flujo, una ruta independiente debe moverse a lo largo de al menos una arista que no se haya recorrido antes al definir la ruta. (figura 3.14)

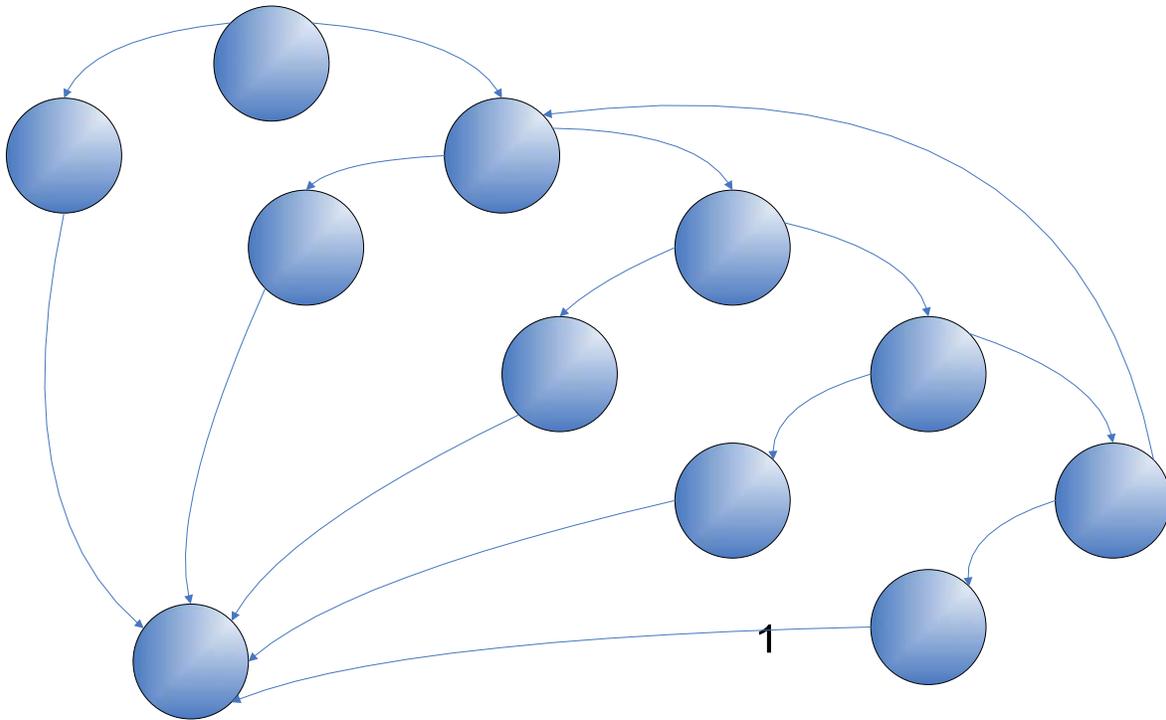


Figura 3.14 Grafo de flujo

2

3,4

De acuerdo al grafo de flujo anterior se consideran las siguientes rutas.

Ruta 1: 1-2-12

5

Ruta 2: 1-3-4-5-12

Ruta 3: 1-3-4-6-7-12

Ruta 4: 1-3-4-6-8-9-12

Ruta 5: 1-3-4-6-8-10-11-12

Ruta 6: 1-3-4-6-8-10-3-4-5-12

7

La *complejidad ciclomática* es una medición de software que proporciona una evaluación cuantitativa de la complejidad lógica del programa. La complejidad ciclomática tiene fundamentos en la teoría de gráficos y proporciona una medición de software. La complejidad se calcula de la siguiente forma, para el sistema desarrollado.

1. El número de regiones del gráfico de flujo corresponde a la complejidad ciclomática.

$$V(G)=5$$

2. La complejidad ciclomática  $V(G)$  para un gráfico de flujo se define como:

$$V(G) = E - N + 2$$

donde E es el número de aristas del gráfico de flujo y N el número de nodos del gráfico de flujo,

$$E = 15$$

$$N = 11$$

sustituyendo

$$V(G) = 15 - 11 + 2$$

$$V(G) = 6$$

3. La complejidad ciclomática  $V(G)$  para un gráfico de flujo también se define como  $V(G) = P + 1$

donde P es el número de nodos predicado contenidos en el gráfico de flujo,

$$P = 5$$

sustituyendo

$$V(G) = 5 + 1$$

$$V(G) = 6$$

Por tanto lo anterior, la complejidad ciclomática del sistema presentado en este trabajo es de  $V(G) = 6$ .

Además de la prueba definida anteriormente son necesarias otras pruebas como la prueba de arquitectura cliente-servidor.

La **prueba de software cliente-servidor** ocurre en tres niveles diferentes 1) las aplicaciones cliente individuales se prueban de un modo “desconectado”; no se considera la operación del servidor ni la red subyacente. 2) El software cliente y las aplicaciones servidor asociadas se prueban en concierto, pero las operaciones de red no se revisan de manera explícita. 3) Se prueba la arquitectura cliente-servidor completa, incluidos la operación de red y el rendimiento.