

Nombre de la práctica	Creación de usuarios			No.	01
Asignatura:	Fundamentos De Ingeniería De Software	Carrera:	Ingeniería en sistemas	Duración de la práctica (Hrs)	

Nombre del alumno: Jesús Navarrete Martínez

Grupo: 3501

I. Competencia(s) específica(s):

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro): Aula

III. Material empleado:

- Equipo de computo

## PRACTICA FINAL

### REGISTRO FAMILIAR

ID:

NOMBRE :

PARENTESCO:

FOTO:

No se eligió archivo

1. Crear la base de datos: Base: listado
2. Crear la Tabla: familia (id, nombre, parentesco, foto)
3. id de tipo int llave primaria, autoincrementa
4. nombre de tipo varchar (50)
5. parentesco tipo varchar (50)
6. foto tipo varchar (200)
7. Botón **SUBIR IMAGEN** permitirá buscar y seleccionar la imagen a registrar, la cual se registrará en el campo foto, al seleccionar la imagen la etiqueta de no se eligió archivo, cambiara por el nombre de la imagen.
8. EL programa permitirá almacenar la información en la base de datos y manejarla de acuerdo a los botones de **REGISTRAR**, **ELIMINAR**, **ACTUALIZAR**.
9. Botón **LISTADO FAMILIAR** permitirá mostrar una lista donde incluya todos los campos de la tabla familia id,nombre,parentesco,foto (imagen de la persona).

## IV. Desarrollo de la práctica:

### 1. Página principal (registroFamiliar.html):

- **Formulario para Registrar un Familiar:**
  - El formulario usa el método POST para enviar datos al script PHP alojado en registrar.php.
  - El atributo enctype="multipart/form-data" es necesario para permitir la subida de archivos (como fotos).
- **Los campos del formulario:**
  - **Nombre:** Un campo de texto requerido.
  - **Parentesco:** Otro campo de texto requerido.
  - **Foto:** Un campo para subir una imagen. El atributo accept="image/\*" asegura que solo se puedan subir archivos de imagen.
- **Botón de Envío del Formulario:**
- Al presionar el botón "Registrar", los datos ingresados se envían al script PHP especificado en el atributo action.
- **Botones de Navegación:**
- Los botones "Consultar", "Eliminar" y "Modificar" son enlaces (<a>) que apuntan a otros scripts PHP.
- Cada enlace está envuelto en un <button> para dar la apariencia de un botón.

### Código

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Registro Familiar</title>
5   <!-- Vinculación de un archivo CSS externo para aplicar estilos -->
6   <link rel="stylesheet" href="style1.css">
7 </head>
8 <body>
9   <!-- Título principal de la página -->
10  <h1>Registro Familiar</h1>
11
12  <!-- Formulario para registrar un nuevo familiar -->
13  <form action="http://localhost/EjerciciosPHPyBD/registrar.php" method="POST" enctype="multipart/form-data">
14    <!-- Campo para ingresar el nombre -->
15    <label for="nombre">Nombre:</label>
16    <input type="text" name="nombre" id="nombre" required>
17
18    <!-- Campo para ingresar el parentesco -->
19    <label for="parentesco">Parentesco:</label>
20    <input type="text" name="parentesco" id="parentesco" required>
21
22    <!-- Campo para subir una foto -->
23    <label for="foto">Foto:</label>
24    <input type="file" name="foto" id="foto" accept="image/*" required>
25
26    <!-- Botón para enviar el formulario -->
27    <button type="submit">Registrar</button>
28  </form>
29
30  <!-- Contenedor para los botones de acciones adicionales -->
31  <div class="botones">
32    <div class="button-group">
33      <!-- Enlaces a diferentes funcionalidades: Consultar, Eliminar y Modificar -->
34      <a href="http://localhost/EjerciciosPHPyBD/consultar.php">
35        <button type="submit">Consultar</button>
36      </a>
37      <a href="http://localhost/EjerciciosPHPyBD/eliminar.php">
38        <button type="submit">Eliminar</button>
39      </a>
40      <a href="http://localhost/EjerciciosPHPyBD/modificar.php">
41        <button type="submit">Modificar</button>
42      </a>
43    </div>
44  </div>
45 </body>
46 </html>
47
```

## Visualización de la pagina

### Registro Familiar

Nombre:

Parentesco:

Foto:

Sin archivos seleccionados

## 2. Archivo registrar.php:

El código permite registrar información de un familiar (nombre, parentesco y foto) en una base de datos. Primero, valida que se haya subido una imagen y la guarda en una carpeta específica. Luego, utiliza una consulta SQL para insertar los datos proporcionados en el formulario y la ruta de la imagen en la tabla familia. Finalmente, confirma el registro o informa errores si algo falla durante el proceso.

## Código

```
registrar.php > ...
1  <?php
2  // Establece la conexión a la base de datos
3  $conexion = mysqli_connect("localhost", "root", "", "listado")
4  or die("Problemas con la conexión");
5
6  // Validación de la subida de imagen
7  if (isset($_FILES['foto']) && $_FILES['foto']['error'] == UPLOAD_ERR_OK) {
8      // Obtener información del archivo subido
9      $nombreImagen = $_FILES['foto']['name']; // Nombre original de la imagen
10     $rutaTemporal = $_FILES['foto']['tmp_name']; // Ruta temporal donde se almacena el archivo en el servidor
11     $rutaDestino = "imagenes/" . $nombreImagen; // Ruta donde se guardará permanentemente la imagen
12
13     // Mueve el archivo desde la ruta temporal a la carpeta destino
14     if (move_uploaded_file($rutaTemporal, $rutaDestino)) {
15         // Si la imagen se movió con éxito, se procede a insertar los datos en la base de datos
16         $nombre = $_POST['nombre']; // Nombre del familiar, enviado desde el formulario
17         $parentesco = $_POST['parentesco']; // Parentesco, enviado desde el formulario
18
19         // Inserta los datos del registro en la tabla 'familia'
20         mysqli_query($conexion, "INSERT INTO familia (nombre, parentesco, foto) VALUES ('$nombre', '$parentesco', '$rutaDestino')")
21         or die("Problemas en el insert: " . mysqli_error($conexion));
22
23         // Mensaje de éxito si la operación fue exitosa
24         echo "El registro fue exitoso.";
25     } else {
26         // Error al mover el archivo a la carpeta destino
27         echo "Error al subir la imagen.";
28     }
29 } else {
30     // Si no se seleccionó una imagen o ocurrió un error
31     echo "Debe seleccionar una imagen.";
32 }
33
34 // Cierra la conexión a la base de datos
35 mysqli_close($conexion);
36 >
```

## 3. Archivo consultar.php:

Este código genera una página web que permite consultar registros almacenados en la base de datos.

### 1. Consulta general o específica:

- Si el usuario ingresa un ID y presiona el botón "Consultar", se realiza una búsqueda específica en la tabla familia para mostrar el registro correspondiente a ese ID.
- Si no se proporciona un ID, se muestran todos los registros disponibles.

### 2. Visualización de resultados:

- Los registros se despliegan en una tabla con columnas para ID, nombre, parentesco y una imagen representando la foto almacenada.

### 3. Conexión a la base de datos:

- Se conecta a la base de datos listado, ejecuta la consulta SQL correspondiente y cierra la conexión.

### 4. Mensajes dinámicos:

- Si no se encuentran registros, se informa que "No hay registros disponibles".

Este archivo es útil para listar y buscar información de manera eficiente.

## Código

```
consultar.php > ...
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Consultar Registros</title>
5 <!-- Enlace a una hoja de estilos CSS externa -->
6 <link rel="stylesheet" href="style2.css">
7 </head>
8 <body>
9 <!-- Contenedor principal de la página -->
10 <div class="container">
11 <h1>Listado Familiar</h1>
12
13 <!-- Formulario para consultar registros por ID -->
14 <form method="GET">
15 <!-- Campo para ingresar el ID a consultar -->
16 <label for="id">Consultar por ID:</label>
17 <input type="number" name="id" id="id" placeholder="Ingrese el ID">
18 <!-- Botón para enviar la consulta -->
19 <button type="submit" name="consulta">Consultar</button>
20 </form>
21
22 <!-- Tabla para mostrar los registros consultados -->
23 <table>
24 <thead>
25 <!-- Encabezados de las columnas -->
26 <tr>
27 <th>ID</th>
28 <th>Nombre</th>
29 <th>Parentesco</th>
30 <th>Foto</th>
31 </tr>
32 </thead>
33 <tbody>
34 <?php
35 // Conexión a la base de datos
36 // Se conecta al servidor de base de datos MySQL utilizando los parámetros: servidor, usuario, contraseña y nombre de la base de datos
37 $conexion = mysqli_connect("localhost", "root", "", "listado")
38 or die("Error de conexión: " . mysqli_connect_error());
39
40 // Verifica si se ha enviado una solicitud de consulta y si el campo ID no está vacío
41 if (isset($_GET['consulta']) && !empty($_GET['id'])) {
42 // Convierte el valor ingresado de ID en un número entero para evitar inyección SQL
43 $id = intval($_GET['id']);
44 // Consulta SQL para obtener el registro correspondiente al ID ingresado
45 $query = "SELECT * FROM familia WHERE id = $id";
46 } else {
47 // Si no se proporciona un ID, se realiza una consulta para obtener todos los registros
48 $query = "SELECT * FROM familia";
49 }
```






```
51 // Ejecuta la consulta SQL y almacena el resultado
52 $resultado = mysqli_query($conexion, $query);
53
54 // Verifica si hay registros en el resultado de la consulta
55 if (mysqli_num_rows($resultado) > 0) {
56     // Recorre cada registro obtenido y lo muestra en la tabla
57     while ($fila = mysqli_fetch_assoc($resultado)) {
58         echo "<tr>"; // Inicia una nueva fila
59         echo "<td>" . $fila['id'] . "</td>"; // Muestra el ID
60         echo "<td>" . $fila['nombre'] . "</td>"; // Muestra el nombre
61         echo "<td>" . $fila['parentesco'] . "</td>"; // Muestra el parentesco
62         // Muestra la foto como una imagen utilizando la ruta almacenada en la base de datos
63         echo "<td><img src='" . $fila['foto'] . "' alt='Foto' class='foto'></td>";
64         echo "</tr>"; // Cierra la fila
65     }
66 } else {
67     // Si no se encuentran registros, muestra un mensaje en una fila de la tabla
68     echo "<tr><td colspan='4'>No hay registros disponibles</td></tr>";
69 }
70
71 // Cierra la conexión a la base de datos
72 mysqli_close($conexion);
73 ?>
74 </tbody>
75 </table>
76 </div>
77 </body>
78 </html>
```

## Visualización de la pagina

### Listado Familiar

Consultar por ID:

[Consultar](#)

ID	Nombre	Parentesco	Foto
1	Hannia	Hermana	
2	Luis	Hermano	
3	Liseth	Hermana mayor	

## 4. Archivo eliminar.php :

### 1. Conexión a la base de datos:

- a) El código se conecta a una base de datos llamada listado utilizando las credenciales root (usuario) y una contraseña vacía.
- b) Si la conexión falla, muestra un mensaje de error.

### 2. Búsqueda de un registro por ID:

- a) Cuando el usuario ingresa un ID en el formulario de búsqueda, el código realiza una consulta SQL para encontrar el registro correspondiente en la tabla familia.
- b) Si se encuentra el registro, sus datos (ID, nombre, parentesco y foto) se almacenan en variables y se muestran en el formulario de confirmación.
- c) Si no se encuentra, se muestra un mensaje indicando que no hay registros con ese ID.

### 3. Eliminación del registro:

- a) Cuando el usuario confirma la eliminación, el código ejecuta una consulta SQL para eliminar el registro cuyo ID fue proporcionado.
- b) Si la operación es exitosa, muestra un mensaje de confirmación; si falla, muestra un mensaje de error.

### 4. Interfaz gráfica: La interfaz permite al usuario:

- a) Buscar un registro mediante su ID.
- b) Ver los detalles del registro encontrado, incluida la foto.
- c) Confirmar la eliminación de dicho registro.

Este código combina interacción con el usuario y manejo dinámico de datos desde una base de datos, ofreciendo una forma práctica de eliminar registros después de una verificación visual.



## Código

```
eliminar.php > ...
1 <?php
2 // Establecer la conexión a la base de datos
3 $conexion = mysqli_connect("localhost", "root", "", "listado")
4 or die("Problemas con la conexión: " . mysqli_connect_error());
5
6 // Inicializar variables para manejar los datos del registro
7 $id = $nombre = $parentesco = $foto_actual = ""; // Variables para almacenar los datos del registro
8 $mensaje = ""; // Variable para mostrar mensajes al usuario
9
10 // Bloque para buscar el registro
11 if (isset($_POST['buscar'])) { // Verifica si se envió el formulario de búsqueda
12     $id_buscar = $_POST['id_buscar']; // Captura el ID ingresado
13     $query = "SELECT * FROM familia WHERE id = $id_buscar"; // Consulta para buscar el registro por ID
14     $resultado = mysqli_query($conexion, $query);
15
16     if ($fila = mysqli_fetch_assoc($resultado)) { // Si encuentra un registro, almacena sus datos
17         $id = $fila['id'];
18         $nombre = $fila['nombre'];
19         $parentesco = $fila['parentesco'];
20         $foto_actual = $fila['foto'];
21     } else { // Si no encuentra un registro, muestra un mensaje de error
22         $mensaje = "<p style='color: red;'>Registro no encontrado.</p>";
23     }
24 }
25
26 // Bloque para eliminar el registro
27 if (isset($_POST['eliminar'])) { // Verifica si se confirmó la eliminación
28     $id = $_POST['id']; // Captura el ID del registro a eliminar
29     $query = "DELETE FROM familia WHERE id=$id"; // Consulta para eliminar el registro
30
31     if (mysqli_query($conexion, $query)) { // Si la eliminación es exitosa, muestra un mensaje
32         $mensaje = "<p style='color: green;'>Registro eliminado correctamente.</p>";
33     } else { // Si ocurre un error durante la eliminación, muestra un mensaje de error
34         $mensaje = "<p style='color: red;'>Error al eliminar: " . mysqli_error($conexion) . "</p>";
35     }
36
37     // Limpia las variables después de la eliminación
38     $id = $nombre = $parentesco = $foto_actual = "";
39 }
40
41 // Cierra la conexión con la base de datos
42 mysqli_close($conexion);
43 >
```

```
44 <!DOCTYPE html>
45 <html>
46 <head>
47     <title>Eliminar Registro</title>
48     <!-- Enlace a una hoja de estilos CSS externa -->
49     <link rel="stylesheet" href="style3.css">
50 </head>
51 <body>
52     <h1>Eliminar Registro</h1>
53
54     <!-- Muestra mensajes al usuario -->
55     <?php echo $mensaje; ?>
56
57     <!-- Formulario para buscar un registro por ID -->
58     <form method="POST">
59         <label for="id_buscar">ID del registro a eliminar:</label>
60         <input type="number" name="id_buscar" id="id_buscar" required>
61         <button type="submit" name="buscar">Buscar</button>
62     </form>
63
64     <!-- Si se encontró un registro, muestra el formulario de confirmación -->
65     <?php if ($id) { ?>
66         <form method="POST">
67             <!-- Campo oculto para enviar el ID del registro a eliminar -->
68             <input type="hidden" name="id" value="<?php echo $id; ?>">
69
70             <!-- Muestra los datos del registro encontrado -->
71             <p><strong>Nombre:</strong> <?php echo $nombre; ?></p>
72             <p><strong>Parentesco:</strong> <?php echo $parentesco; ?></p>
73             <?php if ($foto_actual) { ?>
74                 <p><strong>Foto:</strong></p>
75                 
76             <?php } ?>
77             <br>
78
79             <!-- Botón para confirmar la eliminación -->
80             <button type="submit" name="eliminar">Confirmar Eliminación</button>
81         </form>
82     <?php } ?>
83 </body>
84 </html>
```

## Eliminar Registro

ID del registro a eliminar:

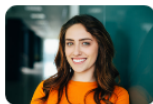
1

Buscar

**Nombre:** Hannia

**Parentesco:** Hermana

**Foto:**



Confirmar Eliminación

### 5. Archivo modificar.php:

#### 1. Conexión a la base de datos:

- El código establece una conexión a la base de datos listado utilizando el usuario root y una contraseña vacía.
- Muestra un mensaje de error si no puede establecer la conexión.

#### 2. Búsqueda de un registro por ID:

- El usuario ingresa el ID del registro que desea modificar.
- El código realiza una consulta SQL para obtener los datos del registro.
- Si se encuentra, los datos (ID, nombre, parentesco, foto) se cargan en variables para que puedan editarse.
- Si no se encuentra el registro, se muestra un mensaje de error.

#### 3. Modificación de un registro:

- El usuario puede modificar el nombre, el parentesco y (opcionalmente) cargar una nueva foto.
- Si se sube una nueva foto, el archivo se guarda en una carpeta llamada imagenes/, y la ruta se almacena en la base de datos.
- Se actualizan los campos correspondientes en la tabla familia.

#### 4. Interfaz gráfica:

- La página tiene un formulario inicial para buscar registros por ID.





- Si se encuentra un registro, se muestra un segundo formulario donde se pueden modificar los datos.
  - Incluye un botón para guardar los cambios.
5. **Mensajes al usuario:**
- El código muestra mensajes de éxito o error según el resultado de las operaciones.
- **Flujo del programa:**
1. El usuario busca un registro por su ID.
  2. Si el registro existe, se cargan sus datos y se muestra un formulario para editar.
  3. El usuario realiza los cambios y confirma la modificación.
  4. Los cambios se actualizan en la base de datos, y se informa al usuario si la operación fue exitosa.

## Código

```
1 <?php
2 <doctype html>
3 <html>
4 <head>
5 <title>Modificar Registro</title>
6 <!-- Hoja de estilos externa -->
7 <link rel="stylesheet" href="style4.css">
8 </head>
9 <body>
10 <h1>Modificar Registro</h1>
11
12 <?php
13 // Establecer la conexión con la base de datos
14 $conexion = mysqli_connect("localhost", "root", "", "listado")
15 or die("Error de conexión: " . mysqli_connect_error());
16
17 // Declaración de variables
18 $id = $nombre = $parentesco = $foto_actual = ""; // Para manejar los datos del registro
19
20 // **Búsqueda del registro por ID**
21 if (isset($_POST['buscar'])) { // Si el usuario envió el formulario de búsqueda
22     $id_buscar = $_POST['id_buscar']; // Captura el ID del registro a buscar
23     $query = "SELECT * FROM familia WHERE id = $id_buscar"; // Consulta para buscar por ID
24     $resultado = mysqli_query($conexion, $query);
25
26     if ($fila = mysqli_fetch_assoc($resultado)) { // Si se encuentra el registro
27         $id = $fila['id'];
28         $nombre = $fila['nombre'];
29         $parentesco = $fila['parentesco'];
30         $foto_actual = $fila['foto']; // Almacena los datos del registro en variables
31     } else { // Si no se encuentra el registro
32         echo "<p style='color: red;'>Registro no encontrado.</p>";
33     }
34 }
35
36 // **Modificación del registro**
37 if (isset($_POST['modificar'])) { // Si el usuario envió el formulario de modificación
38     $id = $_POST['id']; // Captura el ID del registro a modificar
39     $nombre = $_POST['nombre']; // Captura el nuevo nombre
40     $parentesco = $_POST['parentesco']; // Captura el nuevo parentesco
41     $foto = $_FILES['foto']; // Captura el archivo de foto cargado (si existe)
42
43     if ($foto['name']) { // Si se subió una nueva foto
44         $rutaFoto = "imagenes/" . basename($foto['name']); // Define la ruta donde se guardará la foto
45         move_uploaded_file($foto['tmp_name'], $rutaFoto); // Mueve la foto a la ruta especificada
46         $query = "UPDATE familia SET nombre='$nombre', parentesco='$parentesco', foto='$rutaFoto' WHERE id=$id"; // Actualiza con la nueva foto
47     } else { // Si no se subió una nueva foto
48         $query = "UPDATE familia SET nombre='$nombre', parentesco='$parentesco' WHERE id=$id"; // Actualiza solo los campos de texto
49     }
50 }
```



```
50 // Ejecuta la consulta y muestra mensajes según el resultado
51 if (mysqli_query($conexion, $query)) {
52     echo "<p style='color: green;'>Registro modificado correctamente.</p>";
53 } else {
54     echo "<p style='color: red;'>Error al modificar: " . mysqli_error($conexion) . "</p>";
55 }
56 }
57 ?>
58
59 <!-- Formulario para buscar un registro -->
60 <form method="POST" enctype="multipart/form-data">
61     <label for="id_buscar">ID del registro a modificar:</label>
62     <input type="number" name="id_buscar" id="id_buscar" required>
63     <button type="submit" name="buscar">Buscar</button>
64 </form>
65
66 <?php if ($id) { ?>
67     <!-- Formulario para modificar los datos -->
68     <form method="POST" enctype="multipart/form-data">
69         <input type="hidden" name="id" value="<?php echo $id; ?>">
70
71         <label for="nombre">Nombre:</label>
72         <input type="text" name="nombre" id="nombre" value="<?php echo $nombre; ?>">
73
74         <label for="parentesco">Parentesco:</label>
75         <input type="text" name="parentesco" id="parentesco" value="<?php echo $parentesco; ?>">
76
77         <label for="foto">Nueva Foto:</label>
78         <input type="file" name="foto" id="foto">
79         <?php if ($foto_actual) { ?>
80             <p>Foto actual:</p>
81             
82         <?php } ?>
83         <br>
84         <button type="submit" name="modificar">Guardar Cambios</button>
85     </form>
86 <?php } ?>
87
88 <?php mysqli_close($conexion); // Cierra la conexión con la base de datos ?>
89 </body>
90 </html>
```

## Visualización de la pagina

### Modificar Registro

ID del registro a modificar:

Nombre:

Parentesco:

Nueva Foto:

 Sin archivos seleccionados

Foto actual:





## V. Conclusiones:

El desarrollo de este sistema CRUD marca un avance significativo en la gestión de datos al ofrecer un conjunto de funcionalidades completas y bien integradas: registrar, modificar, consultar y eliminar información de manera eficiente. Cada operación desempeña un papel esencial en el manejo de registros, permitiendo a los usuarios interactuar con la base de datos de forma intuitiva y segura.

La funcionalidad de registro asegura un crecimiento estructurado del sistema al permitir la inserción de nuevos datos, mientras que la consulta facilita un acceso rápido y preciso a la información almacenada. Por otro lado, la opción de modificar registros garantiza que la base de datos se mantenga actualizada, reflejando los cambios necesarios de manera ágil. Finalmente, la capacidad de eliminar datos obsoletos o innecesarios contribuye a mantener una base de datos limpia y organizada.

El sistema no solo destaca por su funcionalidad, sino también por su diseño modular y escalable. Esto asegura que sea fácil de mantener y expandir en el futuro, abriendo la posibilidad de incluir mejoras como autenticación de usuarios, validaciones más avanzadas o exportación de datos. Asimismo, la interfaz amigable para el usuario simplifica las operaciones, permitiendo que personas con diferentes niveles de experiencia tecnológica puedan utilizarlo sin complicaciones.