



UTPL
La Universidad Católica de Loja

50 AÑOS
UTPL

UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

Facultad de Ingenierías y Arquitectura

Carrera de Ingeniería en Tecnologías de la Información

Víctor Marcelo Ludeña Pardo

OCT/2024 - FEB/2025

[ACDB1-35%] Foro: Desarrollar un sistema de autenticación de usuarios (login) que permita a los usuarios acceder a una aplicación web utilizando PHP, MySQL y HTML.



Informe de Análisis del Código

Este código implementa un sistema de autenticación básico utilizando PHP, MySQL y HTML, permitiendo que los usuarios inicien sesión, se registren y gestionen su sesión en una aplicación web. A continuación, detallaré cada archivo y su función dentro del sistema. para realizar esto se ha tomado el siguiente esquema solicitado.

Lineamientos generales

Configure la base de datos

Cree el formulario en HTML

Maneje el registro de sesión Maneje la autenticación

Cierre sesión

1. Archivo: estilos.css

Este archivo contiene los estilos CSS utilizados en las páginas web para proporcionar un diseño limpio y moderno.

Este archivo es fundamental para la presentación visual coherente de las páginas web, asegurando que todos los formularios y botones tengan un estilo uniforme y atractivo.

```
1 body {
2     font-family: Arial, sans-serif;
3     margin: 0;
4     padding: 0;
5     background-color: #f4f4f4;
6 }
7
8 h2 {
9     text-align: center;
10    color: #333;
11 }
12
13 form {
14     max-width: 400px;
15     margin: 50px auto;
16     background-color: white;
17     padding: 20px;
18     box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
19 }
20
21 input[type="text"], input[type="password"], input[type="submit"] {
22     width: 100%;
23     padding: 10px;
24     margin: 10px 0;
25 }
26
27 input[type="submit"] {
28     background-color: #007BFF;
29     color: white;
30     border: none;
31     cursor: pointer;
32 }
33
34 input[type="submit"]:hover {
35     background-color: #0056b3;
36 }
37
38 a {
39     text-decoration: none;
40     color: #007BFF;
```



```
41     display: block;
42     text-align: center;
43 }
```

2. Archivo: bienvenida.php

Este archivo se encarga de mostrar la página de bienvenida a los usuarios una vez que hayan iniciado sesión correctamente.

session_start(): Inicia la sesión de PHP para gestionar las variables de sesión.

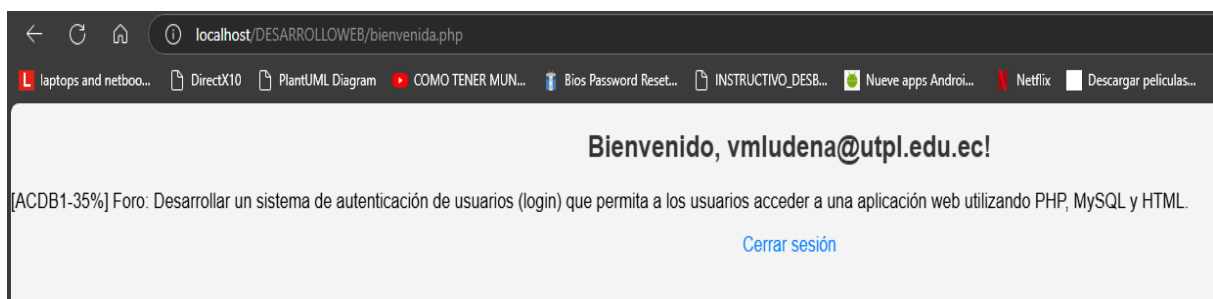
Verificación de sesión: Si no hay una sesión activa (si no está definido \$_SESSION['usuario']), redirige al usuario a la página de inicio de sesión (index.html).

HTML: La página muestra un saludo personalizado con el nombre del usuario (almacenado en la variable de sesión) y un enlace para cerrar sesión.

Cerrar sesión: El enlace redirige al archivo cerrar_sesion.php para finalizar la sesión.

Esta página tiene un propósito importante en el flujo de trabajo del sistema, ya que solo es accesible si el usuario ha iniciado sesión correctamente.

```
<?php
1 session_start();
2 if (!isset($_SESSION['usuario'])) {
3     header("Location: index.html");
4     exit;
5 }
6 ?>
7 <!DOCTYPE html>
8 <html lang="es">
9 <head>
10     <meta charset="UTF-8">
11     <title>Bienvenida</title>
12     <link rel="stylesheet" href="CSS/estilos.css">
13 </head>
14 <body>
15     <h2>Bienvenido, <?php echo $_SESSION['usuario']; ?>!</h2>
16     <p>[ACDB1-35%] Foro: Desarrollar un sistema de autenticación de
17 usuarios (login) que permita a los usuarios acceder a una
18 aplicación web utilizando PHP, MySQL y HTML.</p>
19     <a href="cerrar_sesion.php">Cerrar sesión</a>
20 </body>
</html>
```



3. Archivo: cerrar_sesion.php

Este archivo se encarga de cerrar la sesión del usuario.

session_start(): Inicia la sesión.

session_unset() y session_destroy(): Limpia las variables de sesión y destruye la sesión.

Redirección: Una vez que se destruye la sesión, el usuario es redirigido a la página de inicio de sesión (index.html).

Este archivo es esencial para permitir que el usuario cierre sesión de forma segura.



```
1 <?php
2 session_start();
3 session_unset();
4 session_destroy();
5 header("Location: index.html");
6 exit;
7 ?>
```

Cerrar sesión

4. Archivo: conexion.php

Este archivo establece la conexión con la base de datos MySQL.

Datos de conexión: Se definen los parámetros de conexión como el servidor (127.0.0.1), el usuario (root), la contraseña vacía y la base de datos login_user.

Conexión a MySQL: Utiliza la extensión mysqli para conectar al servidor de base de datos.

Verificación de conexión: Si la conexión falla, se muestra un mensaje de error.

Este archivo es clave para la funcionalidad del sistema, ya que permite la comunicación con la base de datos donde se almacenan los usuarios y sus contraseñas.

```
1 <?php
2 $servername = "127.0.0.1";
3 $username = "root";
4 $password = "";
5 $dbname = "login_user";
6
7 $conn = new mysqli($servername, $username, $password, $dbname);
8
9 if ($conn->connect_error) {
10     die("Conexión fallida: " . $conn->connect_error);
11 }
12 ?>
```

5. Archivo: guardar_usuario.php

Este archivo se encarga de guardar un nuevo usuario en la base de datos.

Obtención de datos: Recibe el nombre de usuario y la contraseña mediante el método POST.

Verificación de existencia de usuario: Se realiza una consulta a la base de datos para verificar si el usuario ya existe.

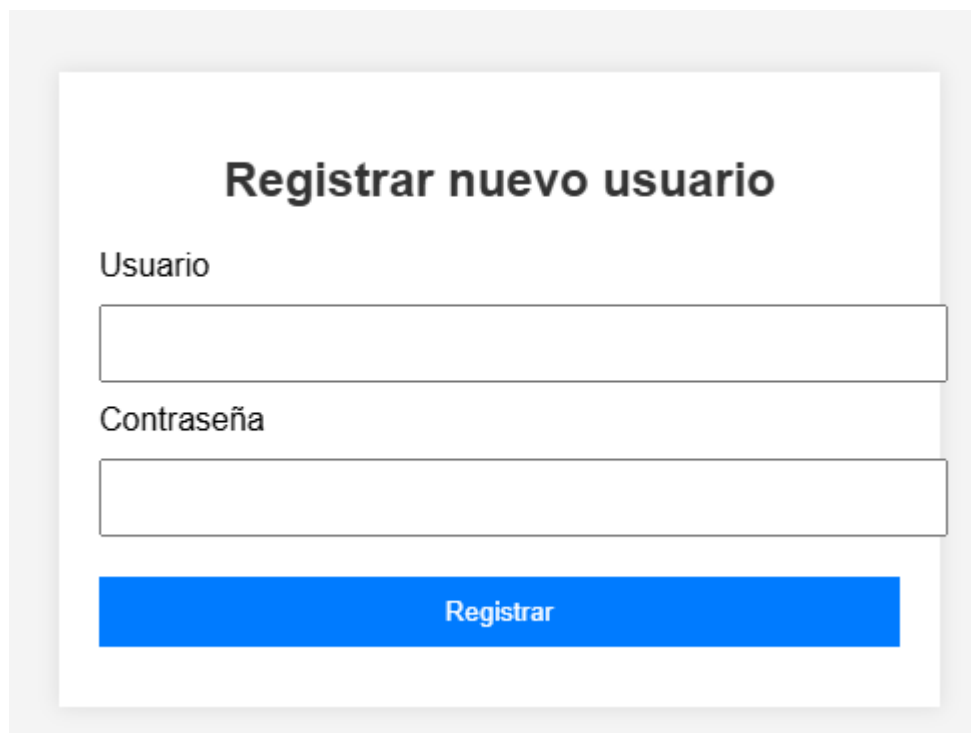
Insertión en la base de datos: Si el usuario no existe, se inserta un nuevo registro con los datos del usuario.

Redirección: Si la inserción es exitosa, se redirige al usuario a la página de inicio (index.html), o se muestra un mensaje de error si la inserción falla.

Este archivo maneja el proceso de registro de nuevos usuarios.

```
1 <?php
2 include('conexion.php');
3
4 if ($_SERVER["REQUEST_METHOD"] == "POST") {
5     $usuario = $_POST['usuario'];
6     $password = $_POST['password'];
7
8     // Verificar si el usuario ya existe
```

```
9 $query = "SELECT * FROM usuarios WHERE usuario = '$usuario'";
10 $result = mysqli_query($conn, $query);
11 $nr = mysqli_num_rows($result);
12
13 if ($nr == 0) {
14     // Insertar el nuevo usuario
15     $query_insert = "INSERT INTO usuarios (usuario, password)
16 VALUES ('$usuario', '$password')";
17     if (mysqli_query($conn, $query_insert)) {
18         header("Location: index.html");
19     } else {
20         echo "Error al registrar el usuario.";
21     }
22 } else {
23     echo "El usuario ya existe.";
24 }
25 }
?>
```



El formulario se encuentra en un recuadro gris con un título "Registrar nuevo usuario" en negrita. Debajo del título hay dos campos de entrada de texto: "Usuario" y "Contraseña". Debajo de estos campos hay un botón azul con el texto "Registrar".

6. Archivo: index.html

Esta es la página de inicio de sesión.

Formulario de inicio de sesión: Permite al usuario ingresar su nombre de usuario y contraseña. Los datos se envían a través del método POST al archivo login.php para su verificación.

Enlace de registro: Si el usuario no tiene cuenta, se ofrece un enlace para registrar un nuevo usuario (reg_usuario.php).

Esta página es crucial para que el usuario ingrese a la aplicación web. Si los datos ingresados son correctos, se redirige al usuario a la página de bienvenida.

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4     <meta charset="UTF-8">
5     <title>Inicio de sesión</title>
6     <link rel="stylesheet" href="CSS/estilos.css">
```



```
7 </head>
8 <body>
9     <form method="post"
10 action="http://localhost/DESARROLLOWEB/login.php">
11         <h2>Iniciar sesión</h2>
12         <label>Usuario</label>
13         <input type="text" name="usuario" required>
14         <label>Contraseña</label>
15         <input type="password" name="password" required>
16         <input type="submit" value="Iniciar sesión">
17         <p><a href="reg_usuario.php">Registrarse</a></p>
18     </form>
19 </body>
</html>
```

Iniciar sesión

Usuario

Contraseña

Iniciar sesión

[Registrarse](#)

7. Archivo: login.php

Este archivo gestiona el proceso de autenticación de usuario.

Obtención de datos: Se reciben el usuario y la contraseña a través del método POST.

Consulta a la base de datos: Se realiza una consulta para verificar si existe un usuario con el nombre de usuario y la contraseña proporcionados.

Autenticación: Si el usuario es encontrado, se inicia una sesión y se redirige al archivo bienvenida.php. Si las credenciales son incorrectas, se muestra un mensaje de error.

Este archivo es el núcleo del sistema de autenticación, ya que valida las credenciales del usuario.

```
1 <?php
2 session_start();
3 include('conexion.php');
4
5 if ($_SERVER["REQUEST_METHOD"] == "POST") {
6     $usuario = $_POST['usuario'];
7     $password = $_POST['password'];
```

```
8
9     $query = "SELECT * FROM usuarios WHERE usuario = '$usuario' AND
10 password = '$password'";
11     $result = mysqli_query($conn, $query);
12     $row = mysqli_fetch_assoc($result);
13
14     if ($row) {
15         $_SESSION['usuario'] = $usuario;
16         header("Location: bienvenida.php");
17     } else {
18         echo "Usuario o contraseña incorrectos.";
19         echo '<a href="index.html">Intentar de nuevo</a>';
20     }
21 }
```

8. Archivo: reg_usuario.php

Esta página permite a los usuarios registrarse en la aplicación.

Formulario de registro: Permite al usuario ingresar un nombre de usuario y una contraseña. Los datos se envían a guardar_usuario.php para ser procesados y almacenados en la base de datos. Estilo y diseño: Al igual que las demás páginas, utiliza el archivo estilos.css para mantener la coherencia visual.

Este archivo es fundamental para que los nuevos usuarios puedan crear una cuenta y acceder a la aplicación posteriormente.

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4     <meta charset="UTF-8">
5     <title>Registrar Usuario</title>
6     <link rel="stylesheet" href="CSS/estilos.css">
7 </head>
8 <body>
9     <form method="post" action="guardar_usuario.php">
10         <h2>Registrar nuevo usuario</h2>
11         <label>Usuario</label>
12         <input type="text" name="usuario" required>
13         <label>Contraseña</label>
14         <input type="password" name="password" required>
15         <input type="submit" value="Registrar">
16     </form>
17 </body>
18 </html>
```

Estos procedimientos se ejecutan en la base de datos y se mostrarían como el siguiente código SQL que realizaría el trabajo realizado en la base.

```
1 create database login_user;
2
3 use login_user;
4
5 CREATE TABLE `usuarios` (
6   `usuario` varchar(50) NOT NULL,
7   `password` varchar(50) NOT NULL,
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
9
10
11 INSERT INTO `usuarios` (`usuario`, `password`) VALUES
12 ('admin', '1234'),
```



```
13 ('vmludena@utpl.edu.ec', '1234');
```

phpMyAdmin

Reciente Favoritas

Nueva

information_schema

login_user

Nueva

usuarios

Columnas

mysql

performance_schema

phpmyadmin

test

Examinar Estructura SQL Buscar Insertar Exportar

La selección actual no contiene una columna única. La edición de la grilla y los enlaces de copiado, eli

Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0008 segundos.)

SELECT * FROM `usuarios`

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

+ Opciones

usuario	password
admin	1234
vmludena@utpl.edu.ec	12345

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Iniciar sesión

Usuario

vmludena@utpl.edu.ec

Contraseña

12345



Iniciar sesión

Registrarse



UTPL
La Universidad Católica de Loja



UTPL
La Universidad Católica de Loja