

PRACTICA: “Uso de Login con bootstrap, php y mysql”

Objetivo: Realizar el dise1o y codificaci3n de Registro y acceso de usuarios con bootstrap, php y mysql”

Fundamentos Te3ricos

Bootstrap es una biblioteca multiplataforma o conjunto de herramientas de c3digo abierto para dise1o de sitios y aplicaciones web. Contiene plantillas de dise1o con tipograf1a, formularios, botones, cuadros, men1s de navegaci3n y otros elementos de dise1o basado en HTML y CSS, as1 como extensiones de JavaScript adicionales. A diferencia de muchos frameworks web, solo se ocupa del desarrollo front-end.

Fundamentos Te3ricos

Bootstrap tiene un soporte relativamente incompleto para HTML5 y CSS3, pero es compatible con la mayor1a de los navegadores web. La informaci3n b1sica de compatibilidad de sitios web o aplicaciones est1 disponible para todos los dispositivos y navegadores. Existe un concepto de compatibilidad parcial que hace disponible la informaci3n b1sica de un sitio web para todos los dispositivos y navegadores. Por ejemplo, las propiedades introducidas en CSS3 para las esquinas redondeadas, gradientes y sombras son usadas por Bootstrap a pesar de la falta de soporte de navegadores antiguos. Esto extiende la funcionalidad de la herramienta, pero no es requerida para su uso.

Cookies y Token aleatorio.

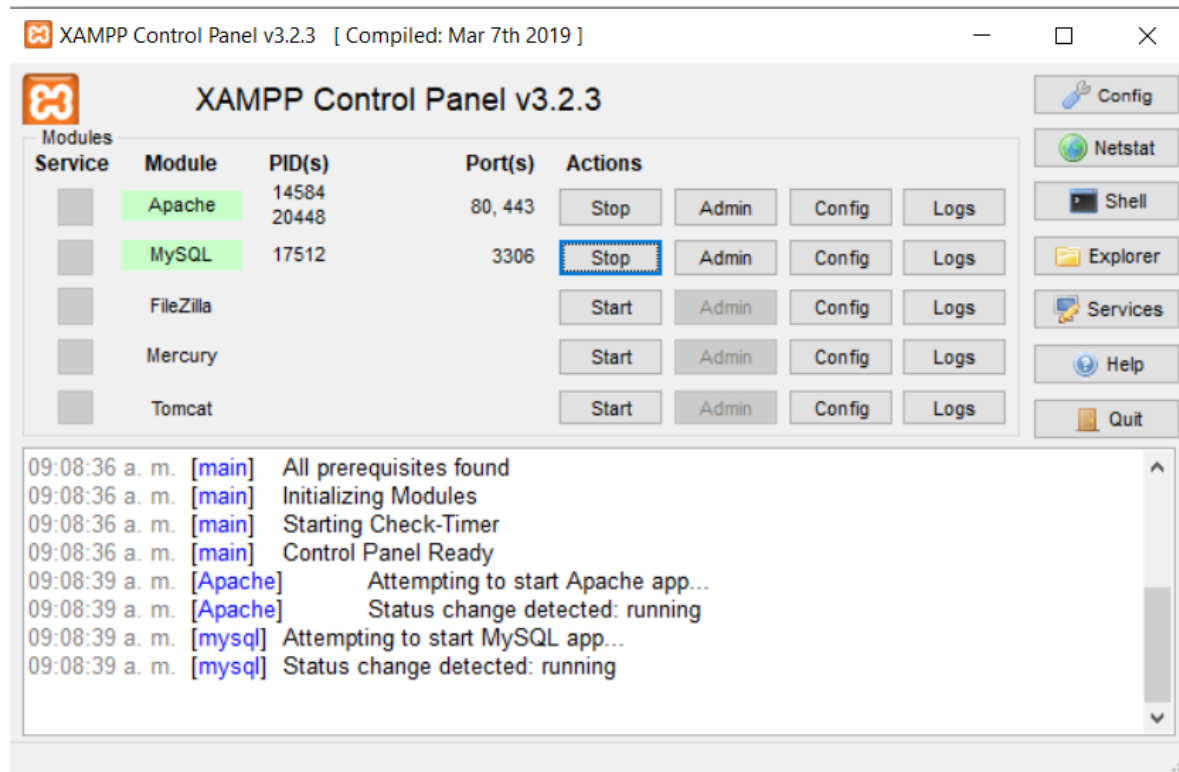
En el momento del registro, a la vez que se guarda en la base de datos el email del usuario y su contrase1a, codificada en base64 , se crea un token hexadecimal aleatorio de 12 d1gitos.

El id del usuario y 3ste nuevo token, se a1aden a unas cookies que quedar1n almacenadas en el ordenador de tus usuarios para que, en caso de volver a visitar tu web, no tengan que volver a logearse.

Para a1adir un punto m1s de seguridad y evitar que un usuario intente modificar su cookie de usuario o token aleatorio para acceder como otro usuario, el sistema, cada vez que entra el usuario, hace una consulta a la base de datos y compara si los datos obtenidos de esa consulta corresponden con las cookies internas. En caso contrario, autom1ticamente elimina las cookies y obliga al usuario a volver a logearse.

DESARROLLO

1.-Levantar los servicios de servidores web y mysql



2.-Crear las siguientes carpetas en htdocs de xampp. La carpeta de la práctica se llama login

Este equipo > Acer (C:) > xampp > htdocs > login

| Nombre | Fecha de modificación | Tipo | Tamaño |
|----------|------------------------|---------------------|--------|
| css | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| usuarios | 13/11/2019 05:53 a. m. | Carpeta de archivos | |

3.- En css copiar los archivos siguientes:

Este equipo > Acer (C:) > xampp > htdocs > login > css

| Nombre | Fecha de modificación | Tipo | Tamaño |
|--------------|------------------------|---------------------|--------|
| bootstrap431 | 13/11/2019 05:54 a. m. | Carpeta de archivos | |
| fontawesome | 13/11/2019 05:54 a. m. | Carpeta de archivos | |
| form | 12/11/2019 01:59 p. m. | Documento de hoj... | 1 KB |
| style | 12/11/2019 02:33 p. m. | Documento de hoj... | 1 KB |

4.- Dentro de usuarios la siguiente carpeta js

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| Nombre | Fecha de modificación | Tipo | Tamaño |
|--------|------------------------|---------------------|--------|
| js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |

5.- Crear una base de datos llamada database y una tabla llamada users

The screenshot shows the phpMyAdmin interface. On the left, the database structure is visible, including 'database', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The 'database' database is selected, and the 'users' table is highlighted. The main panel displays the table structure for 'users' with the following columns:

| # | Nombre | Tipo | Cotejamiento | Atributos | Nulo | Predeterminado | Comentarios | Extra | Acción |
|---|------------|--------------|-------------------|-----------|------|-------------------|-------------|----------------|--|
| 1 | id | int(11) | | | No | Ninguna | | AUTO_INCREMENT | Cambiar Eliminar |
| 2 | username | varchar(50) | latin1_swedish_ci | | No | Ninguna | | | Cambiar Eliminar |
| 3 | password | varchar(255) | latin1_swedish_ci | | No | Ninguna | | | Cambiar Eliminar |
| 4 | token | varchar(255) | latin1_swedish_ci | | No | Ninguna | | | Cambiar Eliminar |
| 5 | created_at | datetime | | | Sí | CURRENT_TIMESTAMP | | | Cambiar Eliminar |

At the bottom, there are options to 'Seleccionar todo' (Select all) and 'Eliminar de las columnas centrales' (Remove from central columns). The interface also includes a sidebar with navigation links like 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar', 'Exportar', 'Importar', 'Privilegios', 'Operaciones', and 'Segu'.

```
CREATE TABLE `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) NOT NULL,
  `password` varchar(255) NOT NULL,
  `token` varchar(255) NOT NULL,
  `created_at` datetime DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=latin1;
```

6.-Crear el primer archivo llamado index.php en la carpeta raíz de la practica

Este equipo > Acer (C:) > xampp > htdocs > login

| | Nombre | Fecha de modificación | Tipo | Tamaño |
|--|----------|------------------------|---------------------|--------|
| | css | 27/06/2022 10:29 a. m. | Carpeta de archivos | |
| | usuarios | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| | index | 27/06/2022 10:21 a. m. | Archivo PHP | 2 KB |

Index.php

```
<?php
require_once "usuarios/config.php";
require_once "usuarios/functions.php";
// Verifique si el usuario ha iniciado sesión, si no, rediríjalo a la página de inicio de sesión
if(!isset($_COOKIE["userCookie"]) && (!isset($_COOKIE["userToken"]))) {
    header("location: usuarios/login.php");
    exit;
    echo $_COOKIE["userCookie"];die();
}
$user_id = $_COOKIE["userCookie"];
$token = $_COOKIE["userToken"];
//-----si manipulan cookies, se borran y redireccionar a login-----
```

```
?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title></title>
    <link rel="stylesheet" href="css/bootstrap431/css/bootstrap.min.css">
```

```

<link rel="stylesheet" type="text/css" href="css/fontawesome/css/all.min.css">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" type="text/css" href="css/style.css">
</head>
<header >
  <div class="header_left">
    <h1><b><?php echo htmlspecialchars($user_id); ?></b></h1>
  </div>
  <div class="header_right">
    <h1><b><?php echo htmlspecialchars($user_id); ?></b></h1>
  </div>
</header>
<body>
  <p>
    <a href="usuarios/reset-password.php" class="btn btn-warning">Resetear Contraseña</a>
    <a href="usuarios/logout.php" class="btn btn-danger">Salir de la Cuenta</a>
  </p>
</body>
</html>

```

7.- En la carpeta usuarios codificar la conexión a la base de datos

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| | Nombre | Fecha de modificación | Tipo | Tamaño |
|--|--------|------------------------|---------------------|--------|
| | js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| | config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |

Config.php

```

<?php
/* Credenciales de la base de datos. Suponiendo que está ejecutando MySQL
servidor con configuración predeterminada (usuario 'root' sin contraseña) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'database');

/* Intento de conexión a la base de datos MySQL */
$link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}

function conectar() {
    $server = "localhost";
    $user = "root";

```

```

$password = "";
$db = "database";
try {
    $conn = new PDO("mysql:host=$server;dbname=$db", $user, $password,
[PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8'"]);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    return $conn;
} catch (Exception $ex) {
    echo $ex->getMessage();
}
}

```

8.-Crear el archivo functions.php para verificar el token para la seguridad al acceso de usuarios

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| Nombre | Fecha de modificación | Tipo | Tamaño |
|-----------|------------------------|---------------------|--------|
| js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| functions | 27/06/2022 09:55 a. m. | Archivo PHP | 1 KB |

Functions.php

```

<?php
// Incluir archivo de configuración
require_once "config.php";

//-----Comprobar token usuario-----
function comprobarToken($user_id, $token_id){
    try {
        $conn = conectar();
        $token = $conn->prepare("select id from users where token = '$token_id' ");
        $token->execute();
        $exist = $token->fetch();
        if($exist['id']!= $user_id){
            //borrar las cookies();
            setcookie("userCookie","", time() - 3600, "/");
            setcookie("userToken", "", time() - 3600, "/");
            // Redirigir a la página de inicio de sesión
            header("location: usuarios/login.php");
        }
    } catch (Exception $ex) {
    }
}
}

```

9.- Codificar el archivo login que permitirá visualizar el formulario de acceso a usuarios

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| | Nombre | Fecha de modificación | Tipo | Tamaño |
|---|-----------|------------------------|---------------------|--------|
| ★ | js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| ★ | config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| ★ | functions | 27/06/2022 09:55 a. m. | Archivo PHP | 1 KB |
| ★ | login | 27/06/2022 09:52 a. m. | Archivo PHP | 5 KB |

login.php

```
<?php
```

```
// Incluir archivo de configuración
require_once "config.php";
```

```
// Definir variables e inicializar con valores vacíos
$username = $password = "";
$username_err = $password_err = "";
```

```
// Procesamiento de datos del formulario cuando se envía el formulario
if($_SERVER["REQUEST_METHOD"] == "POST"){
```

```
    // Comprobar si el nombre de usuario está vacío
    if(empty(trim($_POST["username"]))) {
        $username_err = "Debe escribir su email.";
    } else {
        $username = trim($_POST["username"]);
    }
}
```

```
    // Comprobar si la contraseña está vacía
    if(empty(trim($_POST["password"]))) {
        $password_err = "Es necesario un password.";
    } else {
        $password = trim($_POST["password"]);
    }
}
```

```
// Validar credenciales
if(empty($username_err) && empty($password_err)){
    // Preparar una declaración selecta
    $sql = "SELECT id, username, password, token FROM users WHERE username = ?";
```

```
if($stmt = mysqli_prepare($link, $sql)){
    // Vincular variables a la declaración preparada como parámetros
    mysqli_stmt_bind_param($stmt, "s", $param_username);

    // Establecer parámetros
    $param_username = $username;

    // Intento de ejecutar la declaración preparada
    if(mysqli_stmt_execute($stmt)){
        // Guardar resultado
        mysqli_stmt_store_result($stmt);

        // Verifique si existe el nombre de usuario, si es así, verifique la contraseña
        if(mysqli_stmt_num_rows($stmt) == 1){
            // Vincular variables de resultado
            mysqli_stmt_bind_result($stmt, $id, $username, $db_password,$usertoken);
            if(mysqli_stmt_fetch($stmt)){

                if($password == base64_decode($db_password)){
                    $caducidad = $year = 60 * 60 * 24 * 365 + time();
                    setcookie('userCookie', $id, $caducidad, '/' );
                    setcookie('userToken', $usertoken, $caducidad, '/' );

                    // Redirigir al usuario a la página de bienvenida
                    header("location: ../index.php");
                }else{
                    // Dmostrar un mensaje de error si la contraseña no es válida
                    $password_err = "El password no es válido.";
                }
            }
        } else{
            // Mostrar un mensaje de error si el nombre de usuario no existe
            $username_err = "No existe este usuario.";
        }
    } else{
        echo "Ha habido un error, inténtelo más tarde.";
    }
}

// Cerrar declaración
mysqli_stmt_close($stmt);
}

// cerrar conexion
mysqli_close($link);
}
```



```

?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Login</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
  <link rel="stylesheet" type="text/css" href="../css/form.css">
</head>
<body>

  <div class="wrapper">
    <h2>Acceso a Usuarios</h2>
    <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
      <div class="form-group <?php echo (!empty($username_err)) ? 'has-error' : ''; ?>">
        <label>Email</label>
        <input type="email" name="username" class="form-control" value="<?php echo
$username; ?>">
        <span class="help-block"><?php echo $username_err; ?></span>
      </div>
      <div class="form-group <?php echo (!empty($password_err)) ? 'has-error' : ''; ?>">
        <label>Password</label>
        <input type="password" name="password" class="form-control">
        <span class="help-block"><?php echo $password_err; ?></span>
      </div>
      <div class="form-group">
        <input type="submit" class="btn btn-primary" value="Login">
      </div>
      <p>No tienes una cuenta? <a href="register.php">Registrarse</a>.</p>
      <p>Has olvidado tu password? <a href="recuperar_pass.php">Recuperar
password</a>.</p>
    </form>
  </div>

</body>
</html>
10.-

```

11.- Codificar el archivo de logout que destruye todas las sesiones utilizadas y regresa al inicio

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| | Nombre | Fecha de modificación | Tipo | Tamaño |
|---|-----------|------------------------|---------------------|--------|
| ★ | js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| ★ | config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| ★ | functions | 27/06/2022 10:48 a. m. | Archivo PHP | 1 KB |
| ★ | login | 27/06/2022 09:52 a. m. | Archivo PHP | 5 KB |
| ★ | logout | 27/06/2022 09:57 a. m. | Archivo PHP | 1 KB |

```
logout.php
<?php
// Inicializar la sesión
session_start();
// Desarmar todas las variables de sesión
$_SESSION = array();
// Destruye la sesión.
session_destroy();
//borramos las cookies();
setcookie("userCookie","", time() - 3600, "/");
setcookie("userToken","", time() - 3600, "/");

// Redirigir a la página de inicio de sesión
header("location: login.php");
exit;
?>
```

12.- Codificar el archivo recuperar contraseña

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| | Nombre | Fecha de modificación | Tipo | Tamaño |
|---|----------------|------------------------|---------------------|--------|
| ➤ | js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| ➤ | config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| ➤ | functions | 27/06/2022 10:48 a. m. | Archivo PHP | 1 KB |
| ➤ | login | 27/06/2022 09:52 a. m. | Archivo PHP | 5 KB |
| ➤ | logout | 27/06/2022 09:57 a. m. | Archivo PHP | 1 KB |
| | recuperar_pass | 27/06/2022 09:59 a. m. | Archivo PHP | 2 KB |

```

<?php
// Incluir archivo de configuración
require_once "config.php";
// Definir variables e inicializar con valores vacíos
$username_err = "";
// Procesamiento de datos del formulario cuando se envía el formulario
if($_SERVER["REQUEST_METHOD"] == "POST"){
    $stmt = $link->prepare("SELECT * FROM users WHERE username = ?");
    $stmt->bind_param("s", $_POST['username']);
    $stmt->execute();
    $result = $stmt->get_result();

    // Comprobar si el nombre de usuario está vacío
    if($result->num_rows === 0) {
        $username_err = "Este email no existe.";
    }else{
        $row = $result->fetch_assoc();
        $email = $row['username'];

        $password = base64_decode($row['password']);
    }

    $stmt->close();
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Login</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="../css/form.css">
</head>
<body>
    <div class="wrapper">
        <h2>Recuperar Contraseña</h2>

```

```

<form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
    <div class="form-group <?php echo (!empty($username_err)) ? 'has-error' : ''; ?>">
        <label>Email</label>
        <input type="email" name="username" class="form-control">
        <span class="help-block"><?php echo $username_err; ?></span>
    </div>

    <div class="form-group">
        <input type="submit" class="btn btn-primary" value="Recover">
        <a type="reset" href="login.php" class="btn btn-default" >Volver</a>
    </div>
</form>
</div>
</body>
</html>

```

13.- Codificar el archivo register que tiene como función introducir los datos de un usuario a la tabla users del proyecto de registro

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| Nombre | | Fecha de modificación | Tipo | Tamaño |
|----------------|--|------------------------|---------------------|--------|
| js | | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| config | | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| functions | | 27/06/2022 10:48 a. m. | Archivo PHP | 1 KB |
| login | | 27/06/2022 09:52 a. m. | Archivo PHP | 5 KB |
| logout | | 27/06/2022 09:57 a. m. | Archivo PHP | 1 KB |
| recuperar_pass | | 27/06/2022 09:59 a. m. | Archivo PHP | 2 KB |
| register | | 27/06/2022 09:48 a. m. | Archivo PHP | 6 KB |

register.php

```

<?php
// Incluye el archivo de configuracion
require_once "config.php";

```

```
// Define variables e inicializa valores vacios
$username = $password = $confirm_password = "";
$username_err = $password_err = $confirm_password_err = "";

// Procesa el formulario de informaci3n cuando el formulario es enviado
if($_SERVER["REQUEST_METHOD"] == "POST"){

    // Valida nombre de usuario
    if(empty(trim($_POST["username"]))){
        $username_err = "Please enter a email.";
    } else{
        // Prepara la consulta
        $sql = "SELECT id FROM users WHERE username = ?";

        if($stmt = mysqli_prepare($link, $sql)){
            // Vincular variables a la declaraci3n preparada como par3metros
            mysqli_stmt_bind_param($stmt, "s", $param_username);

            // Establecer par3metros
            $param_username = trim($_POST["username"]);

            // ejecutar la declaraci3n preparada
            if(mysqli_stmt_execute($stmt)){
                /* almacena resultados */
                mysqli_stmt_store_result($stmt);

                if(mysqli_stmt_num_rows($stmt) == 1){
                    $username_err = "This username is already taken.";
                } else{
                    $username = trim($_POST["username"]);
                }
            } else{
                echo "Oops! Something went wrong. Please try again later.";
            }
        }

        // Cerrar declaraci3n
        mysqli_stmt_close($stmt);
    }

    // Validar contrase1a
    if(empty(trim($_POST["password"]))){
        $password_err = "Please enter a password.";
```

```
} elseif(strlen(trim($_POST["password"])) < 6){
    $password_err = "Password must have atleast 6 characters.";
} else{
    $password = trim($_POST["password"]);
}

// Validar confirmar contraseña
if(empty(trim($_POST["confirm_password"]))) {
    $confirm_password_err = "Please confirm password.";
} else{
    $confirm_password = trim($_POST["confirm_password"]);
    if(empty($password_err) && ($password != $confirm_password)){
        $confirm_password_err = "Password did not match.";
    }
}

// Verifique los errores de entrada antes de insertar en la base de datos
if(empty($username_err) && empty($password_err) && empty($confirm_password_err)){

    // Preparar una declaración de inserción
    $sql = "INSERT INTO users (username, password, token) VALUES (?, ?, ?)";

    if($stmt = mysqli_prepare($link, $sql)){
        // Vincular variables a la declaración preparada como parámetros
        mysqli_stmt_bind_param($stmt, "sss", $param_username, $param_password,
        $param_token);

        // Establecer parámetros
        $param_username = $username;
        $param_token = bin2hex(openssl_random_pseudo_bytes(6));

        // Crea un hash de contraseña
        $param_password = base64_encode($password);

        // Intento de ejecutar la declaración preparada
        if(mysqli_stmt_execute($stmt)){
            // Redirigir a la página de inicio de sesión
            header("location: login.php");exit();

        } else{
            echo "Something went wrong. Please try again later.";
        }
    }
}
```

```
// Cerrar declaraci3n
mysqli_stmt_close($stmt);
}

// Conexi3n cercana
mysqli_close($link);
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Sign Up</title>
  <link href="//maxcdn.bootstrapcdn.com/bootstrap/4.1.1/css/bootstrap.min.css"
rel="stylesheet" id="bootstrap-css">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
  <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <link rel="stylesheet" type="text/css" href="../css/form.css">
</head>
<body>
  <div class="wrapper">
    <h2>Registro de Usuarios</h2>
    <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
      <div class="form-group <?php echo (!empty($username_err)) ? 'has-error' : ''; ?>">
        <label>Email</label>
        <input type="email" name="username" class="form-control" value="<?php echo $username; ?>">
        <span class="help-block"><?php echo $username_err; ?></span>
      </div>
      <div class="form-group <?php echo (!empty($password_err)) ? 'has-error' : ''; ?>">
        <label>Password</label>
```

```
<input type="password" name="password" class="form-control" value="<?php echo $password;
?>">

    <span class="help-block"><?php echo $password_err; ?></span>

</div>

<div class="form-group <?php echo (!empty($confirm_password_err)) ? 'has-error' : ''; ?>">

    <label>Confirmar Password</label>

    <input type="password" name="confirm_password" class="form-control" value="<?php echo
$confirm_password; ?>">

        <span class="help-block"><?php echo $confirm_password_err; ?></span>

    </div>

    <div class="form-group">

        <input type="submit" class="btn btn-primary" value="Enviar">

        <input type="reset" class="btn btn-default" value="Reset">

    </div>

    <p>¿Ya tienes una cuenta? <a href="login.php">Entre aquí</a>.</p>

</form>

</div>

</body>

</html>
```

14.-Codificar el archivo reset-password

Este equipo > Acer (C:) > xampp > htdocs > login > usuarios

| Nombre | Fecha de modificación | Tipo | Tamaño |
|----------------|------------------------|---------------------|--------|
| js | 13/11/2019 05:53 a. m. | Carpeta de archivos | |
| config | 27/06/2022 09:54 a. m. | Archivo PHP | 1 KB |
| functions | 27/06/2022 10:48 a. m. | Archivo PHP | 1 KB |
| login | 27/06/2022 09:52 a. m. | Archivo PHP | 5 KB |
| logout | 27/06/2022 09:57 a. m. | Archivo PHP | 1 KB |
| recuperar_pass | 27/06/2022 09:59 a. m. | Archivo PHP | 2 KB |
| register | 27/06/2022 09:48 a. m. | Archivo PHP | 6 KB |
| reset-password | 27/06/2022 10:18 a. m. | Archivo PHP | 4 KB |

reset-password.php

<?php

```
// Incluir archivo de configuración
require_once "config.php";
```

```
// Definir variables e inicializar con valores vacíos
$new_password = $confirm_password = "";
$new_password_err = $confirm_password_err = "";
```

```
//Procesamiento de datos del formulario cuando se envía el formulario
if($_SERVER["REQUEST_METHOD"] == "POST"){
```

```
    // Validar nueva contraseña
    if(empty(trim($_POST["new_password"]))) {
        $new_password_err = "Please enter the new password.";
    } elseif(strlen(trim($_POST["new_password"])) < 6) {
        $new_password_err = "Password must have atleast 6 characters.";
    } else {
        $new_password = trim($_POST["new_password"]);
    }
}
```

```
// Validar confirmar contraseña
if(empty(trim($_POST["confirm_password"]))) {
    $confirm_password_err = "Please confirm the password.";
} else {
    $confirm_password = trim($_POST["confirm_password"]);
    if(empty($new_password_err) && ($new_password != $confirm_password)) {
        $confirm_password_err = "Password did not match.";
    }
}
}
```

```
// Verifique los errores de entrada antes de actualizar la base de datos
if(empty($new_password_err) && empty($confirm_password_err)){
```

```

// Preparar una declaraci3n de actualizaci3n
$sql = "UPDATE users SET password = ? WHERE id = ?";

if($stmt = mysqli_prepare($link, $sql)){
    // Vincular variables a la declaraci3n preparada como par3metros
    mysqli_stmt_bind_param($stmt, "si", $param_password, $param_id);

    // Establecer par3metros
    $param_password = base64_encode($new_password);
    $param_id = $_COOKIE["BS_user_id"];

    // Intento de ejecutar la declaraci3n preparada
    if(mysqli_stmt_execute($stmt)){

        header("location: ../index.php");
        exit();
    } else{
        echo "Oops! Something went wrong. Please try again later.";
    }
}

// cerrar declaracion
mysqli_stmt_close($stmt);
}

// cerrar conexion
mysqli_close($link);
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>recuperar Password</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="../css/form.css">
</head>
<body>
    <div class="wrapper">
        <h2>Reset Password</h2>
        <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
            <div class="form-group <?php echo (!empty($new_password_err)) ? 'has-error' : ''; ?>">
                <label>New Password</label>
                <input type="password" name="new_password" class="form-control" value="<?php echo
$new_password; ?>">

```

```
<span class="help-block"><?php echo $new_password_err; ?></span>
</div>
<div class="form-group <?php echo (!empty($confirm_password_err)) ? 'has-error' : ''; ?>">
  <label>Confirm Password</label>
  <input type="password" name="confirm_password" class="form-control">
  <span class="help-block"><?php echo $confirm_password_err; ?></span>
</div>
<div class="form-group">
  <input type="submit" class="btn btn-primary" value="Submit">
  <a class="btn btn-link" href="welcome.php">Cancel</a>
</div>
</form>
</div>
</body>
</html>
```

15.- Verificar si no existen errors y ejecutar la practica en localhost