

## HITO 2 DEL 2º TRIMESTRE DE Programación

ÁLVARO ORGAZ 1DAM

14/02/2025



Aquí te dejo en enlace a mi repositorio de github:

<https://github.com/alvaroorgaz/HITO-2-2T-PROGRAMACION>

<b>Preparación previa:</b> .....	<b>2</b>
<b>1. Página de Inicio:</b> .....	<b>4</b>
- Crear una página de inicio que presente la opción de registro o inicio de sesión:.....	4
<b>2. Registro de Usuarios:</b> .....	<b>5</b>
- Implementar un formulario de registro con los campos obligatorios: nombre de usuario, correo electrónico y contraseña:.....	5
- Validar que el correo electrónico sea único en la base de datos:.....	6
- Almacenar la información del usuario en la base de datos de MySQL de manera segura (hash de contraseña):.....	6
<b>3. Inicio de Sesión:</b> .....	<b>8</b>
- Crear un formulario de inicio de sesión que valide las credenciales ingresadas:.....	8
- Utilizar sesiones para mantener al usuario autenticado durante su sesión.....	10
<b>4. CR de Tareas: sólo será visible para los usuarios validados.....</b>	<b>10</b>
- Desarrollar una página donde los usuarios puedan agregar nuevas tareas:.....	10
- Mostrar el listado de tareas asociados al usuario identificado en el sistema.....	11
<b>5. Gestión de Sesiones:</b> .....	<b>11</b>
- Crear un sistema que maneje el cierre de sesión de forma segura.....	11
<b>WEBGRAFÍA.....</b>	<b>13</b>

## Preparación previa:

Para el correcto funcionamiento de esta web, es necesario crear la base de datos que gestione usuarios y tareas:

-- Crear la base de datos

```
CREATE DATABASE IF NOT EXISTS gestion_tareas;
```

```
USE gestion_tareas;
```

-- Tabla de usuarios

```
CREATE TABLE usuarios (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre_usuario VARCHAR(50) NOT NULL,  
    correo VARCHAR(100) NOT NULL UNIQUE,  
    contrasena VARCHAR(255) NOT NULL,  
    fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- Tabla de tareas

```
CREATE TABLE tareas (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    usuario_id INT NOT NULL,  
    descripcion TEXT NOT NULL,  
    estado ENUM('pendiente', 'completada') DEFAULT 'pendiente',  
    fecha_creacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (usuario_id) REFERENCES usuarios(id) ON DELETE CASCADE  
);
```

Con este script SQL, creamos las dos tablas necesarias.

Por otro lado, se necesita un archivo común bbdd.php para que las páginas de la web puedan conectarse a la base de datos, pudiendo reutilizar el mismo archivo para hacer esto, sin repetir código:

Este archivo contiene la cabecera PHP con las variables necesarias para conectar con la base de datos, e inicia la conexión a esta:

```
<?php  
// Configuración de la base de datos  
$servidor = "localhost";  
$usuario_db = "root";  
$contrasena_db = "root";  
$base_datos = "gestion_tareas";
```

```
// Crear conexión
$conn = new mysqli($servidor, $usuario_db, $contrasena_db,
$base_datos);

// Verificar conexión
if ($conn->connect_error) {
    die("Error de conexión: " . $conn->connect_error);
}

// Configurar el conjunto de caracteres
$conn->set_charset("utf8");

?>
```

El archivo bbdd.php es utilizado por el resto de páginas.

## 1. Página de Inicio:

- Crear una página de inicio que presente la opción de registro o inicio de sesión:

Para ello, he creado un index.html como página principal:



La página principal muestra las opciones de Iniciar sesión o Registrarse, además del menú de navegación.

La página conecta con la base de datos mediante PHP, y reutiliza el menu.php para mostrar la barra de navegación:

```

<?php
session_start();

// Si el usuario ya está autenticado, redirigir a la página de tareas
if (isset($_SESSION["usuario_id"])) {
    header("Location: tareas.php");
    exit();
}
?>

<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Gestión de Tareas</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <?php include "menu.php"; ?>

    <h1>Bienvenido a la Gestión de Tareas</h1>
    <p>Organiza tus pendientes de manera sencilla.</p>

    <div class="container">
        <h2>Acceder</h2>
        <p><button><a href="login.php">Iniciar Sesión</a></button> |
<button><a href="registro.php" >Registrarse</a></button></p></div>
</body>
</html>

```

## 2. Registro de Usuarios:

- Implementar un formulario de registro con los campos obligatorios: nombre de usuario, correo electrónico y contraseña:

En el apartado Registrarse, el usuario debe rellenar un formulario con sus datos y aceptar las políticas:

The screenshot shows a web interface with a dark blue header containing three buttons: 'Inicio', 'Iniciar Sesión', and 'Registrarse'. Below the header is a white box titled 'Registro de Usuario'. Inside this box, there are three input fields: 'Nombre de usuario', 'Correo electrónico', and 'Contraseña'. Below the 'Contraseña' field is a checkbox labeled 'Acepto las políticas de privacidad'. At the bottom of the white box is a large blue button labeled 'Registrarse'.

- Validar que el correo electrónico sea único en la base de datos:

Si el usuario con ese correo ya está registrado, el formulario muestra un error e impide la creación del usuario repetido.

This screenshot shows the same 'Registro de Usuario' form, but with an error message. A red rectangular box highlights the text 'El correo ya está registrado.' which appears above the input fields. The input fields now contain the text 'alvaro', 'alvaro@gmail.com', and '\*\*\*\*'. The checkbox 'Acepto las políticas de privacidad' is now checked, and the blue 'Registrarse' button remains at the bottom.

- Almacenar la información del usuario en la base de datos de MySQL de manera segura (hash de contraseña):

La contraseña que introduce el usuario se encripta:

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	id	nombre_usuario	correo	contrasena	fecha_registro
▶	1	alvaro	alvaro@gmail.com	\$2y\$10\$ug6AHBo03mfuKckC.VzWF.nhbIz2jDIn...	2025-02-13 20:59:33
	2	alvaro	mario@gmail.com	\$2y\$10\$jubj7kLHz8HjRu3aGN7PheBfa8jMuMXQ...	2025-02-13 21:03:28
*	NULL	NULL	NULL	NULL	NULL

El código PHP del registro comprueba que los datos del usuario son válidos, y de ser así, hace hash de la contraseña, si no lo son, evalúa los errores y avisa como se muestra en las capturas:

```
require_once "bbdd.php";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nombre_usuario = trim($_POST["nombre_usuario"]);
    $correo = trim($_POST["correo"]);
    $contrasena = trim($_POST["contrasena"]);
    $acepta_politicas = isset($_POST["acepta_politicas"]); //
    Verificar si el checkbox está marcado

    if (empty($nombre_usuario) || empty($correo) ||
    empty($contrasena)) {
        echo "<p>Todos los campos son obligatorios.</p>";
    } elseif (!$acepta_politicas) {
        echo "<p>Debes aceptar las políticas para
    registrarte.</p>";
    } else {
        // Verificar que el correo sea único
        $sql = "SELECT id FROM usuarios WHERE correo = ?";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param("s", $correo);
        $stmt->execute();
        $stmt->store_result();

        if ($stmt->num_rows > 0) {
            echo "<p class='label-form'>El correo ya está
    registrado.</p>";
        } else {
            // Insertar nuevo usuario
            $hash_contrasena = password_hash($contrasena,
    PASSWORD_DEFAULT);
```

```

        $sql = "INSERT INTO usuarios (nombre_usuario,
correo, contrasena) VALUES (?, ?, ?)";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param("sss", $nombre_usuario, $correo,
$hash_contrasena);

        if ($stmt->execute()) {
            echo "<p>Registro exitoso. <a
href='login.php'>Iniciar sesión</a></p>";
        } else {
            echo "<p>Error en el registro.</p>";
        }
    }
    $stmt->close();
}
$conn->close();
}

```

### 3. Inicio de Sesión:

- Crear un formulario de inicio de sesión que valide las credenciales ingresadas:

En el apartado Iniciar Sesión, el usuario puede utilizar sus datos del registro para acceder a las tareas:



Si la contraseña es incorrecta, muestra un error:



**Iniciar Sesión**

**Contraseña incorrecta.**

Correo Electrónico:

alvaro@gmail.com

Contraseña:

.....

**Iniciar Sesión**

¿No tienes cuenta? [Regístrate aquí](#)

El código PHP del inicio de sesión, al igual que el registro, valida los datos que ha introducido el usuario y los consulta en la base de datos, para mostrar un error o enviar al usuario a sus tareas:

```
// Verificar si se envió el formulario
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $correo = trim($_POST["correo"]);
    $contrasena = trim($_POST["contrasena"]);

    // Validar que los campos no estén vacíos
    if (empty($correo) || empty($contrasena)) {
        $mensaje = "Por favor, completa todos los campos.";
    } else {
        // Consultar usuario por correo
        $sql = "SELECT id, nombre_usuario, contrasena FROM usuarios
WHERE correo = ?";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param("s", $correo);
        $stmt->execute();
        $stmt->store_result();

        // Verificar si se encontró un usuario
        if ($stmt->num_rows > 0) {
            $stmt->bind_result($id, $nombre_usuario, $contrasena_hash);
            $stmt->fetch();

            // Verificar la contraseña
            if (password_verify($contrasena, $contrasena_hash)) {
```

```

        // Iniciar sesión y guardar datos del usuario
        $_SESSION["usuario_id"] = $id;
        $_SESSION["nombre_usuario"] = $nombre_usuario;
        header("Location: tareas.php"); // Redirigir a la
página de tareas

        exit();
    } else {
        $mensaje = "Contraseña incorrecta.";
    }
} else {
    $mensaje = "No existe una cuenta con ese correo.";
}
$stmt->close();
}
$conn->close();

```

- Utilizar sesiones para mantener al usuario autenticado durante su sesión.

Para ello, se gestiona mediante PHP la sesión:

```

require_once "bbdd.php";
session_start();

```

#### 4. CR de Tareas: sólo será visible para los usuarios validados

- Desarrollar una página donde los usuarios puedan agregar nuevas tareas:

Una vez iniciada la sesión, los usuarios pueden crear tareas con el siguiente formulario:

The screenshot shows a web application interface for task management. At the top, there is a dark navigation bar with two buttons: "Mis Tareas" (highlighted with a red box) and "Cerrar Sesión". Below this, the main header area displays "Gestión de Tareas", "Bienvenido, alvaro", and "Cerrar sesión". Underneath, there is a link "Agregar Nueva Tarea" and a white input field. A prominent blue bar with the text "Agregar" spans the width of the content area. Below this bar is a table titled "Mis Tareas" (highlighted with a red box). The table contains three rows of task data, each with a "Completar" and an "Eliminar" button. The tasks listed are: "tarea programacion web (pendiente)", "segunda tarrea (completada)", and "primera tarea (pendiente)".

Al introducir el contenido de la tarea y pulsar Agregar, la tarea se añade a la lista:

The screenshot shows the 'Gestión de Tareas' interface. At the top, there's a header with 'Gestión de Tareas', 'Bienvenido, alvaro | Cerrar sesión', and 'Agregar Nueva Tarea'. Below the header is a form with a text input field containing 'tarea bases de datos' and a blue 'Agregar' button. Below the form, under the heading 'Mis Tareas', there is a list of tasks: 'tarea programacion web (pendiente) [Completar] [Eliminar]', 'segunda tarrea (completada) [Eliminar]', and 'primera tarea (pendiente) [Completar] [Eliminar]'. The 'Agregar' button and the input field are highlighted with red boxes.

A close-up of a task card showing 'tarea bases de datos (pendiente) [Completar] [Eliminar]'. The card has a light blue background and is set against a dark grey background.

Si se pulsa Completar, la tarea cambia el texto del estado:

A close-up of a task card showing 'tarea bases de datos (completada) [Eliminar]'. The status has changed from 'pendiente' to 'completada', and the 'Completar' button is no longer visible.

También se puede eliminar:

The screenshot shows a confirmation dialog box titled 'localhost dice' with the text '¿Seguro que deseas eliminar esta tarea?'. It has two buttons: 'Aceptar' (highlighted with a red box) and 'Cancelar'. The background shows the 'Gestión de Tareas' interface with the same header and task list as before.

The screenshot shows the 'Gestión de Tareas' interface after deleting a task. The 'Agregar' button is at the top. Below it, under 'Mis Tareas', the task 'tarea bases de datos (completada) [Eliminar]' is now highlighted with a red box. The other tasks remain: 'tarea programacion web (pendiente) [Completar] [Eliminar]', 'segunda tarrea (completada) [Eliminar]', and 'primera tarea (pendiente) [Completar] [Eliminar]'.

- Mostrar el listado de tareas asociados al usuario identificado en el sistema

Las tareas se muestran justo debajo del formulario:



## 5. Gestión de Sesiones:

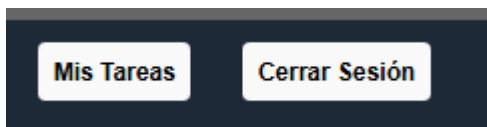
- Crear un sistema que maneje el cierre de sesión de forma segura.

La web cuenta con una opción de Cerrar Sesión, que utiliza un código PHP para cerrar la sesión del usuario:

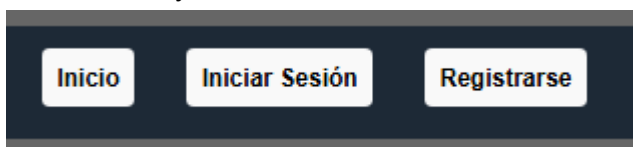
```
<?php
session_start();
session_unset(); // Eliminar todas las variables de sesión
session_destroy(); // Destruir la sesión

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

Además, si hay una sesión activa, la barra de navegación muestra los botones de Mis Tareas y Cerrar sesión:



Pero si no hay una sesión activa, se muestran los de Inicio, Iniciar Sesión y Registrarse:



Deberás controlar y mostrar mensajes al usuario para todas los posibles errores o excepciones.



**Registro de Usuario**

El correo ya está registrado.

alvaro

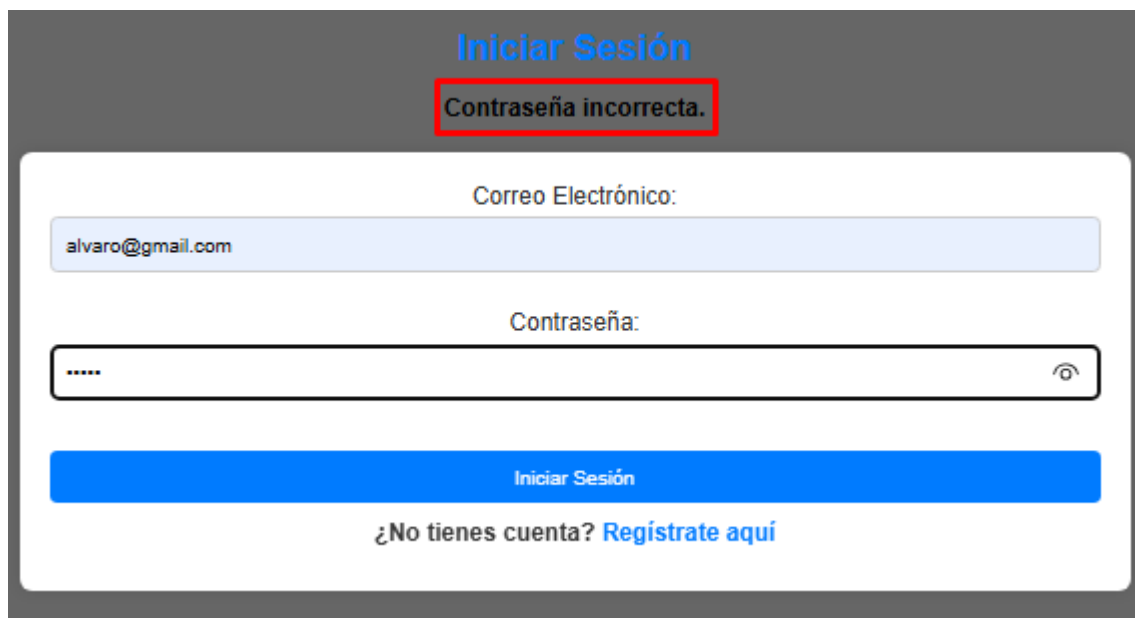
alvaro@gmail.com

\*\*\*\*

☒ Acepto las [políticas de privacidad](#)

Registrarse

Si el usuario ya está registrado, muestra un error.



**Iniciar Sesión**

Contraseña incorrecta.

Correo Electrónico:

alvaro@gmail.com

Contraseña:

\*\*\*\*\*

Iniciar Sesión

¿No tienes cuenta? [Regístrate aquí](#)

Si la contraseña es incorrecta, muestra el aviso.

## WEBGRAFÍA

**-ChatGPT** – <https://chat.openai.com>

Me ha ayudado un poco en algunas cosas de la web

**-Visual Studio** – <https://visualstudio.microsoft.com/>

Con esta herramienta he realizado mi web de gestión de usuarios

**-W3Schools - Python Tutorial** – <https://www.w3schools.com/python/>

En esta página he encontrado información y me ha ayudado para realizar mi proyecto.

