HITO 2 DEL 2º TRIMESTRE DE PROGRAMACIÓN

Rodrigo García Ortiz

14 - 2 – 2025

Índice

Contenido

[conexión.php 1](#_Toc190076104)

[index.php 1](#_Toc190076105)

[class\_usuario.php 2](#_Toc190076106)

[usuariosController.php 2](#_Toc190076107)

[registrar\_usuario.php 3](#_Toc190076108)

[login.php 4](#_Toc190076109)

[class\_tarea.php 6](#_Toc190076110)

[tareasController.php 7](#_Toc190076111)

[lista\_tareas.php 8](#_Toc190076112)

[alta\_tarea.php 9](#_Toc190076113)

[completar\_tarea.php 11](#_Toc190076114)

[eliminar\_tarea.php 11](#_Toc190076115)

[logout.php 11](#_Toc190076116)

[tareas.sql 11](#_Toc190076117)

[Webgrafía 13](#_Toc190076118)

# conexión.php

<?php

class Conexion {

    private $servidor = 'localhost';

    private $usuario = 'root';

    private $password = 'curso';

    private $base\_datos = 'tareas';

    public $conexion;

    public function \_\_construct() {

        $this->conexion = new mysqli($this->servidor, $this->usuario, $this->password, $this->base\_datos);

        if ($this->conexion->connect\_error) {

            die("Error de conexión: " . $this->conexion->connect\_error);

        }

    }

    public function cerrar() {

        $this->conexion->close();

    }

}

?>

# index.php

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

</head>

<body>

    <h1>Inicio</h1>

    <div class="container mt-4">

        <div class="mb-3">

            <a href="login.php" class = "btn btn-primary">Iniciar sesión</a> <!-- Elige con que usuario iniciar sesión -->

        </div>

        <div class="mb-3"> 1.

            <a href="vista/registrar\_usuario.php" class = "btn btn-primary">Registrarse</a> <!-- Registra un nuevo usuario -->

        </div>

    </div>

</body>

</html>

# class\_usuario.php

<?php

require\_once '../config/conexion.php';

class Usuario {

    private $conexion;

    public function \_\_construct() {

        $this->conexion = new Conexion();

    }

    public function registrarUsuario($nombre, $correo, $contrasena) { //Guarda los datos del usuario en la base de datos

        $query = "INSERT INTO usuarios (nombre, correo, contrasena) VALUES (?, ?, ?)";

        $stmt = $this->conexion->conexion->prepare($query);

        $stmt->bind\_param("sss", $nombre, $correo, $contrasena);

        if ($stmt->execute()) {

            header("Location: ../index.php"); //Al terminar regresa al indice

        } else {

            echo "Error al registrar usuario: " . $stmt->error;

        }

        $stmt->close();

    }

}

?>

# usuariosController.php

<?php

require\_once '../modelo/class\_usuario.php';

class UsuariosController {

    private $modelo;

2.

    public function \_\_construct() {

        $this->modelo = new Usuario();

    }

    public function registrarUsuario($nombre, $correo, $contrasena) { //Guarda los datos del usuario en la base de datos

        $this->modelo->registrarUsuario($nombre, $correo, $contrasena);

    }

}

?>

# registrar\_usuario.php

<?php

require\_once '../controlador/usuariosController.php';

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') { //Guarda los datos recibidos en la base de datos

    $controller = new UsuariosController();

    $controller->registrarUsuario(

        $\_POST['usuario'],

        $\_POST['correo'],

        password\_hash($\_POST['contrasena'], PASSWORD\_DEFAULT),

    );

    exit();

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

    <title>Registrar Usuario</title>

</head>

<body>

    <div class="container mt-4">

        <h1>Registrar Usuario</h1>

        <form method = "POST">

            <div class="mb-3">

                <label for="usuario" class="form-label">Usuario</label> <!-- Guarda el nombre del usuario --> 3.

                <input type="text" class="form-control" id="usuario" name="usuario" required>

            </div>

            <div class="mb-3">

                <label for="correo" class="form-label">Correo</label> <!-- Guarda el correo del usuario -->

                <input type="email" class="form-control" id="correo" name="correo" required>

            </div>

            <div class="mb-3">

                <label for="contrasena" class="form-label">Contraseña</label> <!-- Guarda la contraseña del usuario -->

                <input type="password" class="form-control" id="contrasena" name="contrasena" required>

            </div>

            <div class="mb-3">

                <label><h5>He leído y acepto las políticas de privacidad</h5></label><br> <!-- Acepta las políticas de privacidad -->

                <label>Sí</label>

                <input type="radio" id="privacidad" name="privacidad" value="si">

                <br>

                <label>No</label>

                <input type="radio" id="privacidad" name="privacidad" value="no">

            </div>

            <button type="submit" class="btn btn-primary">Registrar</button>

        </form>

    </div>

</body>

</html>

# login.php

<?php

session\_start();

require\_once 'config/conexion.php';

$conexion = new Conexion();

$query = "SELECT \* FROM usuarios";

        $resultado = $conexion->conexion->query($query);

        $usuarios = [];

        while ($fila = $resultado->fetch\_assoc()) {

            $usuarios[] = $fila; 4.

        }

foreach ($usuarios as $usuario):

    if ($\_POST['usuario'] == $usuario['nombre']) { //Si la contraseña y el usuario están en la base de datos permite acceder a la página

        if ($\_POST['contrasena'] == $usuario['contrasena']){

            $\_SESSION['usuario'] = $usuario['nombre']; //El usuario introducido se vuelve el usuario de la sesión

            $\_SESSION['id'] = $usuario['id\_usuario']; //El id del usuario introducido se vuelve el id de la sesión

            header("Location: vista/lista\_tareas.php");

        } else {

            echo "Contraseña incorrecta.";

        }

    } else {

        echo "Usuario incorrecto.";

    }

endforeach

?>

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

</head>

<body>

    <div class="container mt-4">

        <h1>Iniciar sesión</h1>

        <form method = "POST">

            <div class="mb-3">

                <label for="usuario" class="form-label">Usuario</label> <!-- Recibe el nombre del usuario a utilizar -->

                <input type="text" class="form-control" id="usuario" name="usuario" required>

            </div>

5.

            <div class="mb-3">

                <label for="contrasena" class="form-label">Contraseña</label> <!-- Recibe la contraseña del usuario a utilizar -->

                <input type="password" class="form-control" id="contrasena" name="contrasena">

            </div>

            <button type="submit" class="btn btn-primary">Iniciar sesión</button>

        </form>

    </div>

</body>

</html>

# class\_tarea.php

<?php

require\_once '../config/conexion.php';

class Tarea {

    private $conexion;

    public function \_\_construct() {

        $this->conexion = new Conexion();

    }

    public function agregarTarea($usuario, $nombre, $estado) { //Guarda los datos de la tarea en la base de datos

        $query = "INSERT INTO tarea (usuario, nombre, estado) VALUES (?, ?, ?)";

        $stmt = $this->conexion->conexion->prepare($query);

        $stmt->bind\_param("sss", $usuario, $nombre, $estado);

        if ($stmt->execute()) {

            header("Location: ../vista/lista\_tareas.php"); //Al terminar regresa a la lista de tareas

        } else {

            echo "Error al agregar tarea: " . $stmt->error;

        }

        $stmt->close();

    }

    public function obtenerTareas($usuario) { //Recoge los datos de todas las tareas del usuario en uso

        $query = "SELECT \* FROM tarea where usuario = ?";

        $stmt = $this->conexion->conexion->prepare($query); 6.

        $stmt->bind\_param("i", $usuario);

        $stmt->execute();

        $resultado = $stmt->get\_result();

        $tareas = [];

        while ($fila = $resultado->fetch\_assoc()) {

            $tareas[] = $fila;

        }

        return $tareas;

    }

    public function completarTarea($id\_tarea, $estado) { //Marca la tarea como completada

        $query = "UPDATE tarea SET estado = ? WHERE id\_tarea = ?";

        $stmt = $this->conexion->conexion->prepare($query);

        $stmt->bind\_param("si", $estado, $id\_tarea);

        if ($stmt->execute()) {

            header("Location: ../vista/lista\_tareas.php"); //Al terminar regresa a la lista de tareas

        } else {

            echo "Error al actualizar la tarea: " . $stmt->error;

        }

        $stmt->close();

    }

    public function eliminarTarea($id\_tarea) { //Elimina los datos de la tarea indicada

        $query = "DELETE FROM tarea WHERE id\_tarea = ?";

        $stmt = $this->conexion->conexion->prepare($query);

        $stmt->bind\_param("i", $id\_tarea);

        if ($stmt->execute()) {

            header("Location: ../vista/lista\_tareas.php"); //Al terminar regresa a la lista de tareas

        } else {

            echo "Error al eliminar la tarea: " . $stmt->error;

        }

        $stmt->close();

    }

}

?>

# tareasController.php

<?php

require\_once '../modelo/class\_tarea.php';

7.

class TareasController {

    private $modelo;

    public function \_\_construct() {

        $this->modelo = new Tarea();

    }

    public function agregarTarea($usuario, $nombre, $estado) { //Guarda los datos de la tarea en la base de datos

        $this->modelo->agregarTarea($usuario, $nombre, $estado);

    }

    public function obtenerTareas($usuario) { //Recoge los datos de las tareas del usuario indicado

        return $this->modelo->obtenerTareas($usuario);

    }

    public function completarTarea($id\_tarea, $estado) { //Marca la tarea como completada

        $this->modelo->completarTarea($id\_tarea, $estado);

    }

    public function eliminarTarea($id\_tarea) { //Elimina los datos de la tarea indicada

        $this->modelo->eliminarTarea($id\_tarea);

    }

}

?>

# lista\_tareas.php

<?php

session\_start();

if ($\_SESSION['usuario'] == "") {

    header("Location: ../index.php");

}

$usuario = $\_SESSION['id'];

require\_once '../controlador/tareasController.php';

$controller = new TareasController();

$tareas = $controller->obtenerTareas($usuario);

?>

<!DOCTYPE html>

<html lang="es">

<head> 8.

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

    <title>Listado de Tareas</title>

</head>

<body>

    <h1>Tareas</h1>

    <table border="1" class = "table table-striped"> <!-- Muestra las tareas del usuario elegido -->

        <tr>

            <th>Usuario</th>

            <th>Nombre</th>

            <th>Estado</th>

            <th>Opciones</th>

        </tr>

        <?php foreach ($tareas as $tarea): ?>

            <tr>

                <td><?= $tarea['usuario'] ?></td>

                <td><?= $tarea['nombre'] ?></td>

                <td><?= $tarea['estado'] ?></td>

                <td>

                    <?php if ($tarea['estado'] == "incompleta") { ?>

                        <a href="completar\_tarea.php?id=<?= $tarea['id\_tarea'] ?>" class = "btn btn-primary">Marcar como completada</a>

                    <?php } else {} ?>

                    <a href="eliminar\_tarea.php?id=<?= $tarea['id\_tarea'] ?>" class = "btn btn-primary">Eliminar</a>

                </td>

            </tr>

        <?php endforeach; ?>

    </table>

    <br>

    <a href="alta\_tarea.php" class = "btn btn-primary">Agregar una nueva tarea</a>

    <a href="../logout.php" class = "btn btn-primary">Cerrar sesión</a>

</body>

</html>

# alta\_tarea.php

<?php

session\_start();

require\_once '../controlador/tareasController.php';

9.

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') { //Guarda la nueva tarea en la base de datos

    $controller = new TareasController();

    $controller->agregarTarea(

        $\_SESSION['id'],

        $\_POST['nombre'],

        $\_POST['estado'],

    );

    exit();

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

    <title>Agregar Tarea</title>

</head>

<body>

    <div class="container mt-4">

        <h1>Agregar Tarea</h1>

        <form method = "POST">

            <div class="mb-3">

                <label for="nombre" class="form-label">Nombre</label> <!-- Guarda el nombre de la tarea -->

                <input type="text" class="form-control" id="nombre" name="nombre" required>

            </div>

            <div class="mb-3">

                <label for="estado" class="form-label">Estado</label> <!-- Guarda el estado de la tarea -->

                <select class="form-control" id="estado" name="estado">

                    <option value="completada">Completada</option>

                    <option value="incompleta">Incompleta</option>

                </select>

            </div>

            <button type="submit" class="btn btn-primary">Agregar</button>

        </form>

    </div>

</body> 10.

</html>

# completar\_tarea.php

<?php

session\_start();

require\_once '../controlador/tareasController.php';

$controller = new TareasController();

$id\_tarea = $\_GET['id']; //Obtiene el id de la tarea

$estado = "completada";

$resultado = $controller->completarTarea($id\_tarea, $estado); //Cambia el estado de la tarea a completado

?>

# eliminar\_tarea.php

<?php

session\_start();

require\_once '../controlador/tareasController.php';

$controller = new TareasController();

$id\_tarea = $\_GET['id']; //Obtiene el id de la tarea

$controller->eliminarTarea($id\_tarea) //Elimina la tarea indicada

?>

# logout.php

<?php //Cierra sesión y vuelve al inicio

session\_start();

session\_unset();

session\_destroy();

header("Location: index.php");

?>

# tareas.sql

create database tareas;

use tareas;

create table usuarios(

id\_usuario int primary key auto\_increment,

nombre varchar(50),

correo varchar(50) unique, 11.

contrasena varchar(100)

);

create table tarea(

id\_tarea int primary key auto\_increment,

usuario int,

nombre varchar(50),

estado varchar(50),

foreign key (usuario) references usuarios (id\_usuario)

);

insert into usuarios (nombre, correo, contrasena) values ('Rodrigo', 'rodrigo@ejemplo.com', '1234');

insert into usuarios (nombre, correo, contrasena) values ('Juan', 'juan@ejemplo.com', 'abcd');

insert into tarea (usuario, nombre, estado) values (1, 'EJ1', 'incompleta');

insert into tarea (usuario, nombre, estado) values (2, 'EJ2', 'incompleta');

12.

# Webgrafía

PHP crypt(), password\_hash() Functions. (2018, febrero 20). GeeksforGeeks. <https://www.geeksforgeeks.org/php-crypt-password_hash-functions/>

HTML input type="radio". (s/f). W3schools.com. Recuperado el 10 de febrero de 2025, de <https://www.w3schools.com/tags/att_input_type_radio.asp>

<https://github.com/Rodrigo-Garcia-Ortiz/Programacion.git>

13.