**Informe de Avance - Módulo de Gestión de Usuarios**

**Sistema:** Hieribal - Panel Administrativo  
**Módulo:** Gestión de Usuarios (CRUD completo)

Realizado por Gustavo Alexis Cuevas Morales

**Objetivo General**

**Mostrar una tabla con todos los usuarios (administrativos) registrados en el sistema, permitiendo visualizar su información básica y dejando preparados los botones para futuras acciones CRUD.**

**Descripción general**

**Se desarrolló e integró completamente el módulo de gestión de usuarios, el cual permite visualizar, crear, editar, eliminar y cambiar el estado de los usuarios dentro del sistema. Este módulo está construido siguiendo el patrón MVC y se comunica mediante AJAX para una experiencia fluida sin recargar la página.**

**Estructura y archivos utilizados**

**modelo/ModeloUsuario.php**Contiene todos los métodos para interactuar con la base de datos (listar, crear, editar, eliminar, cambiar estado).

**<?php**

**require\_once 'ModeloBase.php';**

**class ModeloUsuario extends ModeloBase {**

**public function loginUsuario(string $usuario, string $password): false|array {**

**$sql = "SELECT \* FROM usuarios WHERE usuario = :usuario LIMIT 1";**

**$stmt = $this->preparar($sql, [':usuario' => $usuario]);**

**$user = $this->obtenerUno($stmt);**

**if ($user && password\_verify($password, $user['password'])) {**

**return $user;**

**}**

**return false;**

**}**

**public function contarUsuarios(): int {**

**$sql = "SELECT COUNT(\*) FROM usuarios";**

**$stmt = $this->preparar($sql);**

**$stmt->execute();**

**return (int) $stmt->fetchColumn();**

**}**

**public function contarUsuariosPorEstado(string $estado): int {**

**$sql = "SELECT COUNT(\*) FROM usuarios WHERE LOWER(estado) = LOWER(:estado)";**

**$stmt = $this->pdo->prepare($sql);**

**$stmt->execute([':estado' => $estado]);**

**return (int) $stmt->fetchColumn();**

**}**

**public function contarUsuariosPorRol($rol) {**

**$sql = "SELECT COUNT(\*) as total FROM usuarios WHERE rol = :rol";**

**$stmt = $this->pdo->prepare($sql);**

**$stmt->bindParam(':rol', $rol);**

**$stmt->execute();**

**$row = $stmt->fetch(PDO::FETCH\_ASSOC);**

**return $row['total'];**

**}**

**public function buscarPorCorreo(string $correo): false|array {**

**$correo = strtolower(trim($correo));  // Asegura limpieza**

**$sql = "SELECT \* FROM usuarios WHERE LOWER(TRIM(correo)) = :correo LIMIT 1";**

**$stmt = $this->preparar($sql, [':correo' => $correo]);**

**return $this->obtenerUno($stmt);**

**}**

**public function guardarTokenRecuperacion(int $id\_usuario, string $token): void {**

**$sql = "UPDATE usuarios SET token\_recuperacion = :token WHERE id\_usuario = :id";**

**$this->preparar($sql, [':token' => $token, ':id' => $id\_usuario])->execute();**

**}**

**public function verificarToken(string $token): false|array {**

**$sql = "SELECT \* FROM usuarios WHERE token\_recuperacion = :token LIMIT 1";**

**$stmt = $this->preparar($sql, [':token' => $token]);**

**return $this->obtenerUno($stmt);**

**}**

**public function actualizarContraseñaConToken(string $token, string $nueva): void {**

**$nuevaHash = password\_hash($nueva, PASSWORD\_BCRYPT);**

**$sql = "UPDATE usuarios SET password = :password WHERE token\_recuperacion = :token";**

**$stmt = $this->preparar($sql, [**

**':password' => $nuevaHash,**

**':token' => $token**

**]);**

**$stmt->execute();**

**}**

**public function listarUsuarios(): array {**

**$sql = "SELECT id\_usuario, usuario, rol, nombres, apellidos, correo, estado FROM usuarios";**

**$stmt = $this->preparar($sql);**

**$stmt->execute();**

**return $stmt->fetchAll(PDO::FETCH\_ASSOC);**

**}**

**public function actualizarEstado(int $id\_usuario, string $estado): bool {**

**$sql = "UPDATE usuarios SET estado = :estado WHERE id\_usuario = :id";**

**$stmt = $this->preparar($sql, [**

**':estado' => $estado,**

**':id' => $id\_usuario**

**]);**

**return $stmt->execute();**

**}**

**public function actualizarUsuario(array $datos): bool {**

**$sql = "UPDATE usuarios SET**

**usuario = :usuario,**

**rol = :rol,**

**nombres = :nombres,**

**apellidos = :apellidos,**

**correo = :correo**

**WHERE id\_usuario = :id";**

**$stmt = $this->preparar($sql, [**

**':usuario'   => $datos['usuario'],**

**':rol'       => $datos['rol'],**

**':nombres'   => $datos['nombres'],**

**':apellidos' => $datos['apellidos'],**

**':correo'    => $datos['correo'],**

**':id'        => $datos['id\_usuario']**

**]);**

**return $stmt->execute();**

**}**

**public function usuarioExisteEdit(string $usuario, int $excluirId = 0): bool {**

**$sql = "SELECT COUNT(\*) FROM usuarios WHERE usuario = :usuario AND id\_usuario != :id";**

**$stmt = $this->preparar($sql, [':usuario' => $usuario, ':id' => $excluirId]);**

**return $stmt->fetchColumn() > 0;**

**}**

**public function eliminarUsuario(int $id): bool {**

**$sql = "DELETE FROM usuarios WHERE id\_usuario = :id";**

**$stmt = $this->preparar($sql, [':id' => $id]);**

**return $stmt->execute();**

**}**

**public function crearUsuario(array $datos): bool {**

**$sql = "INSERT INTO usuarios (usuario, nombres, apellidos, correo, rol, password, estado)**

**VALUES (:usuario, :nombres, :apellidos, :correo, :rol, :password, :estado)";**

**$stmt = $this->preparar($sql, $datos);**

**return $stmt->execute();**

**}**

**public function usuarioExiste(string $usuario): bool {**

**$sql = "SELECT COUNT(\*) FROM usuarios WHERE usuario = :usuario";**

**$stmt = $this->preparar($sql, [':usuario' => $usuario]);**

**return $stmt->fetchColumn() > 0;**

**}**

**public function correoExiste(string $correo): bool {**

**$sql = "SELECT COUNT(\*) FROM usuarios WHERE correo = :correo";**

**$stmt = $this->preparar($sql, [':correo' => $correo]);**

**return $stmt->fetchColumn() > 0;**

**}**

**}**

**Vista: vista/usuarios.php:  
Archivo principal que contiene la interfaz de gestión de usuarios. Incluye la tabla con DataTables, modales de crear y editar, y los botones de acción.  
  
<?php**

**require\_once '../modelo/ModeloUsuario.php';**

**$modelo = new ModeloUsuario();**

**$usuarios = $modelo->listarUsuarios();**

**?>**

**<link rel="stylesheet" href="../assets/css/usuarios.css">**

**<!-- Encabezado con botón Crear -->**

**<div class="d-flex justify-content-between align-items-center mb-3">**

**<h4>Gestión de Usuarios</h4>**

**<button class="btn btn-success btn-sm" data-bs-toggle="modal" data-bs-target="#modalCrearUsuario">**

**<i class="bi bi-plus-lg"></i> Crear Usuario**

**</button>**

**</div>**

**<!-- Tabla -->**

**<div class="table-responsive">**

**<table id="tablaUsuarios" class="table table-bordered table-hover w-100">**

**<thead class="table-light">**

**<tr>**

**<th>ID</th>**

**<th>Usuario</th>**

**<th>Rol</th>**

**<th>Nombres</th>**

**<th>Apellidos</th>**

**<th>Correo</th>**

**<th>Estado</th>**

**<th>Acciones</th>**

**</tr>**

**</thead>**

**<tbody>**

**<?php foreach ($usuarios as $usuario): ?>**

**<tr>**

**<td><?= $usuario['id\_usuario'] ?></td>**

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

**<td><?= ucfirst($usuario['rol']) ?></td>**

**<td><?= $usuario['nombres'] ?></td>**

**<td><?= $usuario['apellidos'] ?></td>**

**<td><?= $usuario['correo'] ?></td>**

**<!-- Estado visible y también en texto oculto para que lo indexe la búsqueda -->**

**<td>**

**<span style="display:none;"><?= $usuario['estado'] ?></span>**

**<button**

**class="btn btn-sm <?= $usuario['estado'] === 'Activo' ? 'btn-success' : 'btn-secondary' ?> toggle-estado"**

**data-id="<?= $usuario['id\_usuario'] ?>"**

**data-estado="<?= $usuario['estado'] ?>">**

**<?= $usuario['estado'] ?>**

**</button>**

**</td>**

**<td>**

**<button**

**class="btn btn-sm btn-warning btn-editar"**

**data-id="<?= $usuario['id\_usuario'] ?>"**

**data-usuario="<?= $usuario['usuario'] ?>"**

**data-rol="<?= $usuario['rol'] ?>"**

**data-nombres="<?= $usuario['nombres'] ?>"**

**data-apellidos="<?= $usuario['apellidos'] ?>"**

**data-correo="<?= $usuario['correo'] ?>"**

**>**

**Editar**

**</button>**

**<button class="btn btn-sm btn-danger btn-eliminar" data-id="<?= $usuario['id\_usuario'] ?>">Eliminar</button>**

**</td>**

**</tr>**

**<?php endforeach; ?>**

**</tbody>**

**</table>**

**</div>**

**<!-- Modal Editar Usuario -->**

**<div class="modal fade" id="modalEditarUsuario" tabindex="-1" aria-labelledby="modalEditarLabel" aria-hidden="true">**

**<div class="modal-dialog">**

**<form id="formEditarUsuario">**

**<div class="modal-content">**

**<div class="modal-header">**

**<h5 class="modal-title">Editar Usuario</h5>**

**<button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Cerrar"></button>**

**</div>**

**<div class="modal-body">**

**<input type="hidden" name="id\_usuario" id="edit\_id\_usuario">**

**<div class="mb-3">**

**<label class="form-label">Usuario</label>**

**<input type="text" class="form-control" id="edit\_usuario" name="usuario" required pattern="^[a-zA-Z0-9\_]{4,20}$">**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Nombres</label>**

**<input type="text" class="form-control" id="edit\_nombres" name="nombres" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Apellidos</label>**

**<input type="text" class="form-control" id="edit\_apellidos" name="apellidos" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Correo</label>**

**<input type="email" class="form-control" id="edit\_correo" name="correo" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Rol</label>**

**<select class="form-select" id="edit\_rol" name="rol" required>**

**<option value="admin">Admin</option>**

**<option value="empleado">Empleado</option>**

**<option value="cajero">Cajero</option>**

**</select>**

**</div>**

**</div>**

**<div class="modal-footer">**

**<button type="submit" class="btn btn-success">Guardar cambios</button>**

**<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancelar</button>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**<!-- Modal Crear Usuario -->**

**<div class="modal fade" id="modalCrearUsuario" tabindex="-1" aria-labelledby="modalCrearLabel" aria-hidden="true">**

**<div class="modal-dialog">**

**<form id="formCrearUsuario">**

**<div class="modal-content">**

**<div class="modal-header">**

**<h5 class="modal-title">Crear Nuevo Usuario</h5>**

**<button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Cerrar"></button>**

**</div>**

**<div class="modal-body">**

**<div class="mb-3">**

**<label class="form-label">Usuario</label>**

**<input type="text" class="form-control" id="nuevo\_usuario" name="usuario" required pattern="^[a-zA-Z0-9\_]{4,20}$">**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Nombres</label>**

**<input type="text" class="form-control" id="nuevo\_nombres" name="nombres" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Apellidos</label>**

**<input type="text" class="form-control" id="nuevo\_apellidos" name="apellidos" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Correo</label>**

**<input type="email" class="form-control" id="nuevo\_correo" name="correo" required>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Rol</label>**

**<select class="form-select" id="nuevo\_rol" name="rol" required>**

**<option value="admin">Admin</option>**

**<option value="empleado">Empleado</option>**

**<option value="cajero">Cajero</option>**

**</select>**

**</div>**

**<div class="mb-3">**

**<label class="form-label">Contraseña</label>**

**<input type="password" class="form-control" id="nuevo\_password" name="password" required minlength="6">**

**</div>**

**</div>**

**<div class="modal-footer">**

**<button type="submit" class="btn btn-primary">Crear Usuario</button>**

**<button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancelar</button>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**<!-- JavaScript y DataTables -->**

**<script src="https://code.jquery.com/jquery-3.7.0.min.js"></script>**

**<script src="https://cdn.datatables.net/1.13.6/js/jquery.dataTables.min.js"></script>**

**<script src="https://cdn.datatables.net/1.13.6/js/dataTables.bootstrap5.min.js"></script>**

**<link rel="stylesheet" href="https://cdn.datatables.net/1.13.6/css/dataTables.bootstrap5.min.css">**

**<script>**

**$(document).ready(function () {**

**$('#tablaUsuarios').DataTable({**

**language: {**

**url: 'https://cdn.datatables.net/plug-ins/1.13.6/i18n/es-ES.json'**

**},**

**pageLength: 1,**

**searching: true,**

**responsive: true,**

**scrollX: true**

**});**

**});**

**</script>**

**<script src="../assets/js/usuarios.js"></script>**

**Controladores AJAX: Ubicados en la carpeta ajax/, estos scripts reciben las peticiones del frontend y llaman al modelo correspondiente.  
  
crear\_usuario.php: <?php**

**require\_once '../modelo/ModeloUsuario.php';**

**$modelo = new ModeloUsuario();**

**if (**

**isset($\_POST['usuario'], $\_POST['nombres'], $\_POST['apellidos'], $\_POST['correo'], $\_POST['rol'], $\_POST['password'])**

**) {**

**$datos = [**

**'usuario'   => trim($\_POST['usuario']),**

**'nombres'   => trim($\_POST['nombres']),**

**'apellidos' => trim($\_POST['apellidos']),**

**'correo'    => trim($\_POST['correo']),**

**'rol'       => trim($\_POST['rol']),**

**'password'  => password\_hash($\_POST['password'], PASSWORD\_DEFAULT),**

**'estado'    => 'Activo'**

**];**

**// Validar duplicados**

**if ($modelo->usuarioExiste($datos['usuario'])) {**

**echo 'usuario\_duplicado';**

**exit;**

**}**

**if ($modelo->correoExiste($datos['correo'])) {**

**echo 'correo\_duplicado';**

**exit;**

**}**

**$exito = $modelo->crearUsuario($datos);**

**echo $exito ? 'ok' : 'error';**

**} else {**

**echo 'faltan\_datos - DEBUG: ';**

**var\_dump($\_POST); // 👈 esto mostrará qué datos llegan realmente**

**}**

**editar\_usuario.php: <?php**

**require\_once '../modelo/ModeloUsuario.php';**

**$modelo = new ModeloUsuario();**

**// Validar que se recibieron todos los datos necesarios**

**if (**

**isset($\_POST['id\_usuario'], $\_POST['usuario'], $\_POST['nombres'], $\_POST['apellidos'], $\_POST['correo'], $\_POST['rol'])**

**) {**

**$datos = [**

**'id\_usuario' => (int) $\_POST['id\_usuario'],**

**'usuario'    => trim($\_POST['usuario']),**

**'nombres'    => trim($\_POST['nombres']),**

**'apellidos'  => trim($\_POST['apellidos']),**

**'correo'     => trim($\_POST['correo']),**

**'rol'        => trim($\_POST['rol'])**

**];**

**// ✅ Verificar si el usuario ya existe (excluyendo el actual)**

**if ($modelo->usuarioExisteEdit($datos['usuario'], $datos['id\_usuario'])) {**

**echo 'usuario\_duplicado';**

**exit;**

**}**

**$exito = $modelo->actualizarUsuario($datos);**

**echo $exito ? 'ok' : 'error';**

**} else {**

**echo 'faltan\_datos';**

**}**

**eliminar\_usuario.php: <?php**

**require\_once '../modelo/ModeloUsuario.php';**

**$modelo = new ModeloUsuario();**

**if (isset($\_POST['id'])) {**

**$id = (int) $\_POST['id'];**

**$resultado = $modelo->eliminarUsuario($id);**

**echo $resultado ? 'ok' : 'error';**

**} else {**

**echo 'faltan\_datos';**

**}**

**cambiar\_estado.php: <?php**

**require\_once '../modelo/ModeloUsuario.php';**

**if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {**

**$id = isset($\_POST['id']) ? intval($\_POST['id']) : 0;**

**$estado = isset($\_POST['estado']) ? trim($\_POST['estado']) : '';**

**if ($id > 0 && in\_array($estado, ['Activo', 'Inactivo'])) {**

**$modelo = new ModeloUsuario();**

**$exito = $modelo->actualizarEstado($id, $estado);**

**echo $exito ? 'ok' : 'error';**

**} else {**

**echo 'error';**

**}**

**}**

**?>**

**JavaScript: assets/js/usuarios.js: Se encarga de manejar todos los eventos de la tabla, botones y formularios con AJAX y jQuery.  
document.addEventListener('DOMContentLoaded', () => {**

**// Activar/Desactivar estado**

**document.querySelectorAll('.toggle-estado').forEach(boton => {**

**boton.addEventListener('click', () => {**

**const id = boton.dataset.id;**

**const estadoActual = boton.dataset.estado;**

**const nuevoEstado = estadoActual === 'Activo' ? 'Inactivo' : 'Activo';**

**fetch('../ajax/actualizar\_estado\_usuario.php', {**

**method: 'POST',**

**headers: { 'Content-Type': 'application/x-www-form-urlencoded' },**

**body: `id=${id}&estado=${nuevoEstado}`**

**})**

**.then(res => res.text())**

**.then(respuesta => {**

**if (respuesta === 'ok') {**

**boton.textContent = nuevoEstado;**

**boton.dataset.estado = nuevoEstado;**

**boton.classList.toggle('btn-success');**

**boton.classList.toggle('btn-secondary');**

**} else {**

**alert('Error al actualizar el estado');**

**}**

**})**

**.catch(error => {**

**console.error('Error en la solicitud AJAX', error);**

**});**

**});**

**});**

**// Abrir modal con datos al hacer clic en "Editar"**

**document.querySelectorAll('.btn-editar').forEach(btn => {**

**btn.addEventListener('click', () => {**

**document.getElementById('edit\_id\_usuario').value = btn.dataset.id;**

**document.getElementById('edit\_usuario').value = btn.dataset.usuario;**

**document.getElementById('edit\_nombres').value = btn.dataset.nombres;**

**document.getElementById('edit\_apellidos').value = btn.dataset.apellidos;**

**document.getElementById('edit\_correo').value = btn.dataset.correo;**

**document.getElementById('edit\_rol').value = btn.dataset.rol;**

**const modal = new bootstrap.Modal(document.getElementById('modalEditarUsuario'));**

**modal.show();**

**});**

**});**

**// Enviar formulario para actualizar usuario con validación**

**const formularioEditar = document.getElementById('formEditarUsuario');**

**formularioEditar.addEventListener('submit', function (e) {**

**e.preventDefault();**

**const nombres = document.getElementById('edit\_nombres').value.trim();**

**const apellidos = document.getElementById('edit\_apellidos').value.trim();**

**const correo = document.getElementById('edit\_correo').value.trim();**

**const usuario = document.getElementById('edit\_usuario').value.trim();**

**const soloLetras = /^[a-zA-ZÁÉÍÓÚáéíóúÑñ\s]+$/;**

**const correoValido = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;**

**const usuarioValido = /^[a-zA-Z0-9\_]{4,20}$/;**

**if (!soloLetras.test(nombres) || !soloLetras.test(apellidos)) {**

**alert('❌ Nombres y apellidos solo deben contener letras.');**

**return;**

**}**

**if (!correoValido.test(correo)) {**

**alert('❌ Correo inválido.');**

**return;**

**}**

**if (!usuarioValido.test(usuario)) {**

**alert('❌ Usuario inválido. Solo letras, números y guiones bajos (mínimo 4 caracteres).');**

**return;**

**}**

**const formData = new FormData(this);**

**fetch('../ajax/actualizar\_usuario.php', {**

**method: 'POST',**

**body: formData**

**})**

**.then(res => res.text())**

**.then(respuesta => {**

**if (respuesta === 'usuario\_duplicado') {**

**alert('❌ El nombre de usuario ya existe.');**

**return;**

**}**

**if (respuesta === 'ok') {**

**alert('✅ Usuario actualizado correctamente');**

**location.reload();**

**} else {**

**alert('❌ Error al actualizar el usuario: ' + respuesta);**

**}**

**})**

**.catch(error => {**

**console.error('Error al enviar los datos:', error);**

**alert('❌ Ocurrió un error inesperado.');**

**});**

**});**

**});**

**// Eliminar usuario con confirmación**

**document.querySelectorAll('.btn-eliminar').forEach(btn => {**

**btn.addEventListener('click', () => {**

**const id = btn.dataset.id;**

**if (confirm('¿Estás seguro de que deseas eliminar este usuario? Esta acción no se puede deshacer.')) {**

**fetch('../ajax/eliminar\_usuario.php', {**

**method: 'POST',**

**headers: { 'Content-Type': 'application/x-www-form-urlencoded' },**

**body: `id=${id}`**

**})**

**.then(res => res.text())**

**.then(respuesta => {**

**if (respuesta === 'ok') {**

**alert('✅ Usuario eliminado correctamente');**

**location.reload();**

**} else {**

**alert('❌ Error al eliminar el usuario: ' + respuesta);**

**}**

**})**

**.catch(error => {**

**console.error('Error en la solicitud:', error);**

**alert('❌ Ocurrió un error inesperado.');**

**});**

**}**

**});**

**});**

**const formCrear = document.getElementById('formCrearUsuario');**

**formCrear.addEventListener('submit', function (e) {**

**e.preventDefault();**

**const formData = new FormData(this);**

**fetch('../ajax/crear\_usuario.php', {**

**method: 'POST',**

**body: formData**

**})**

**.then(res => res.text())**

**.then(respuesta => {**

**console.log("Respuesta del servidor:", respuesta); // 👈 agrega esto**

**alert(respuesta); // 👈 agrega esto para ver el mensaje real**

**if (respuesta === 'ok') {**

**alert('✅ Usuario creado correctamente');**

**location.reload();**

**} else if (respuesta === 'usuario\_duplicado') {**

**alert('❌ El nombre de usuario ya existe');**

**} else if (respuesta === 'correo\_duplicado') {**

**alert('❌ El correo ya está en uso');**

**} else if (respuesta === 'faltan\_datos') {**

**alert('❌ Faltan datos en el formulario');**

**} else {**

**alert('❌ Error al crear el usuario');**

**}**

**})**

**.catch(error => {**

**console.error('Error al crear el usuario:', error);**

**alert('❌ Ocurrió un error inesperado');**

**});**

**});**

**Estilos:** **assets/css/usuarios.css: Archivo CSS dedicado al diseño visual del módulo, mejorando colores, márgenes, botones y espaciado de la paginación.**/\* RESET BÁSICO \*/

body {

  font-family: 'Segoe UI', sans-serif;

  background-color: #f4f6f9;

  color: #333;

  margin: 0;

  padding: 20px;

}

/\* TÍTULO \*/

h4 {

  font-weight: 700;

  margin-bottom: 20px;

}

/\* CONTENEDOR \*/

.table-responsive {

  background: #fff;

  border-radius: 12px;

  padding: 20px;

  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.06);

}

/\* TABLA \*/

#tablaUsuarios {

  width: 100%;

  border-collapse: collapse;

}

#tablaUsuarios thead {

  background-color: #007bff;

  color: white;

}

#tablaUsuarios th, #tablaUsuarios td {

  padding: 12px;

  text-align: center;

  vertical-align: middle;

}

#tablaUsuarios tbody tr:nth-child(even) {

  background-color: #f9f9f9;

}

#tablaUsuarios tbody tr:hover {

  background-color: #eef2f7;

}

/\* BOTÓN ESTADO \*/

.toggle-estado.btn-success::before {

  content: "✅ ";

}

.toggle-estado.btn-secondary::before {

  content: "❌ ";

}

/\* BOTONES ACCIÓN \*/

.btn-editar {

  background-color: #ffc107;

  color: #000;

  border: none;

  padding: 6px 12px;

  font-weight: bold;

  border-radius: 5px;

}

.btn-editar::before {

  content: "✏️ ";

}

.btn-eliminar {

  background-color: #dc3545;

  color: #fff;

  border: none;

  padding: 6px 12px;

  font-weight: bold;

  border-radius: 5px;

}

.btn-eliminar::before {

  content: "🗑️ ";

}

/\* BOTÓN CREAR \*/

button[data-bs-target="#modalCrearUsuario"] {

  font-weight: bold;

  font-size: 14px;

  padding: 8px 14px;

  border-radius: 8px;

}

/\* PAGINACIÓN \*/

.dataTables\_wrapper .dataTables\_paginate {

  margin-top: 30px !important;

  text-align: center !important;

}

.dataTables\_wrapper .dataTables\_paginate .paginate\_button {

  background-color: #fff !important;

  border: 1px solid #dee2e6 !important;

  border-radius: 6px !important;

  padding: 6px 12px !important;

  margin: 0 3px;

  color: #007bff !important;

}

.dataTables\_wrapper .dataTables\_paginate .paginate\_button.current {

  background-color: #007bff !important;

  color: #fff !important;

  font-weight: bold;

}

/\* INPUT BUSCAR \*/

.dataTables\_filter input {

  border-radius: 20px;

  border: 1px solid #ccc;

  padding: 6px 12px;

  outline: none;

  transition: 0.3s;

}

.dataTables\_filter input:focus {

  border-color: #007bff;

}

/\* SELECT CANTIDAD \*/

.dataTables\_length select {

  border-radius: 6px;

  padding: 4px 8px;

}

/\* INFO \*/

.dataTables\_info {

  font-size: 14px;

  margin-top: 10px;

}

/\* MODALES BONITOS \*/

.modal-content {

  border-radius: 12px;

  border: none;

  box-shadow: 0 6px 18px rgba(0, 0, 0, 0.2);

}

.modal-header {

  background-color: #007bff;

  color: white;

  border-bottom: none;

  border-top-left-radius: 12px;

  border-top-right-radius: 12px;

}

.modal-title {

  font-weight: bold;

}

.modal-footer {

  border-top: none;

}

.btn-close {

  filter: invert(1);

}

**Funcionalidades implementadas**  
**Listado dinámico de usuarios:**  
Se muestra la tabla con todos los usuarios registrados, con soporte para búsqueda, ordenamiento y paginación mediante DataTables.

**Búsqueda en vivo:**  
Permite encontrar coincidencias aunque estén en páginas diferentes.

**Creación de usuarios:**  
Formulario modal con validaciones y envío por AJAX. Se utiliza password\_hash() para almacenar contraseñas de forma segura.

**Edición de usuarios:**  
El modal se llena con los datos actuales del usuario y permite actualizar campos mediante AJAX.

**Eliminación de usuarios:**  
Botón que permite borrar un registro con confirmación, todo sin recargar la página.

**Cambio de estado (activo/inactivo):**  
Botón con colores e íconos que permite activar o desactivar usuarios fácilmente.

**Diseño visual**

Tabla mejorada visualmente con colores en los botones (Editar, Eliminar, Estado).

Paginación separada y más accesible.

Íconos/emojis para una experiencia más clara e intuitiva.

Se aplicó Bootstrap 5 y CSS personalizado para mejorar el aspecto general de la tabla y modales.

**Tecnologías utilizadas**

PHP con POO y PDO.

MySQL.

Bootstrap 5.

jQuery y AJAX.

DataTables.

CSS personalizado

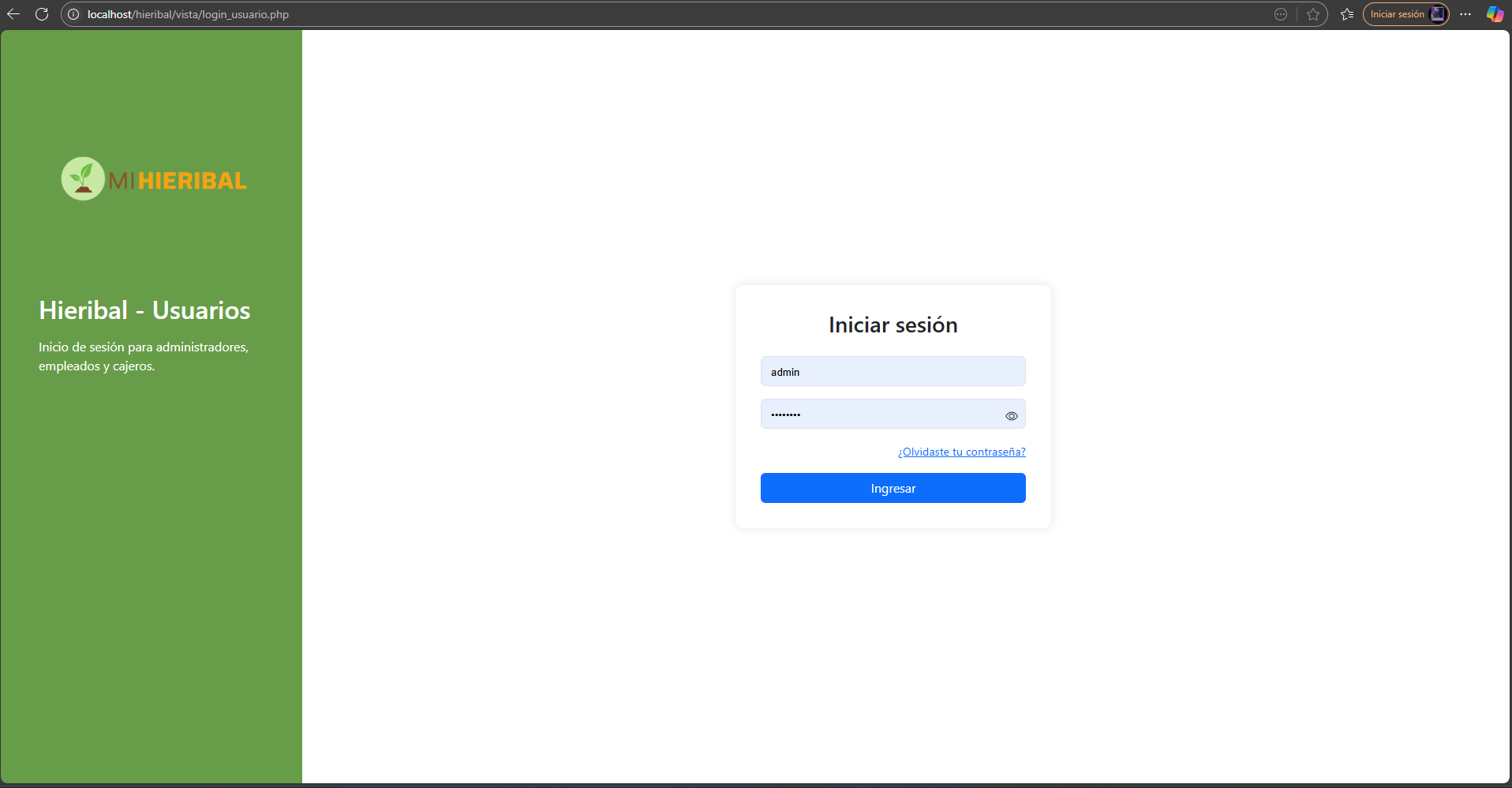
**Pendientes o mejoras futuras**

* Mejorar el diseño visual de los modales.
* Agregar filtros por rol o estado.
* Validación en tiempo real para evitar correos o usuarios duplicados.
* Exportar a Excel o PDF.
* Mejorar el responsive en móviles.

**Fotos y funcionamiento del modulo**

**Pantalla de inicio de sesión del módulo de usuarios**

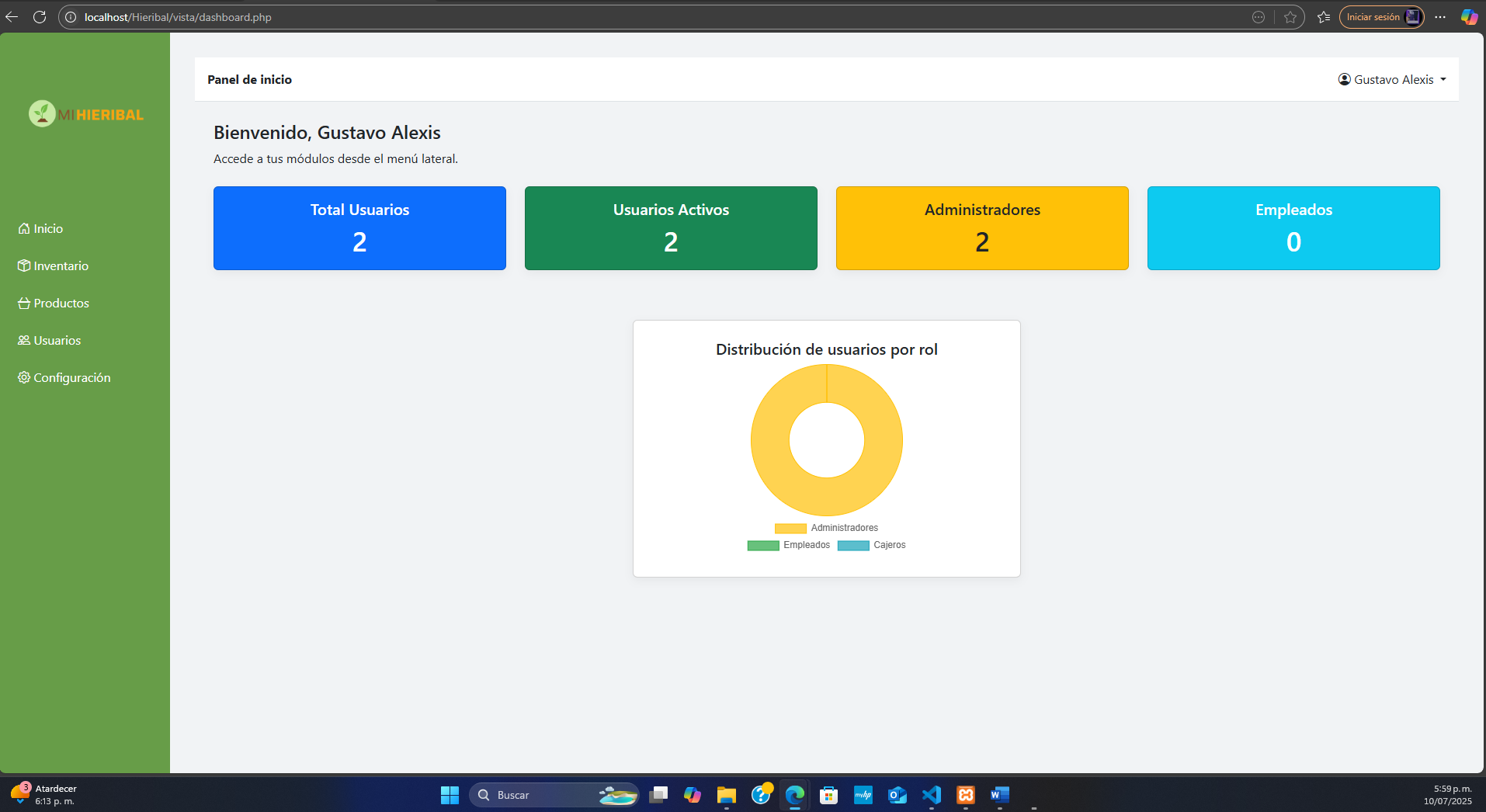
Esta interfaz permite el acceso al sistema de gestión de Hieribal para usuarios autenticados. Está diseñada para administradores, empleados y cajeros.  
El formulario incluye los campos para ingresar el nombre de usuario y la contraseña, junto con un botón para visualizar la clave digitada y un enlace para recuperación de contraseña en caso de olvido.  
Una vez autenticado correctamente, el usuario es redirigido al panel principal (dashboard.php) con acceso a los módulos habilitados según su rol (aun no esta integrado).



**Dashboard principal del sistema**

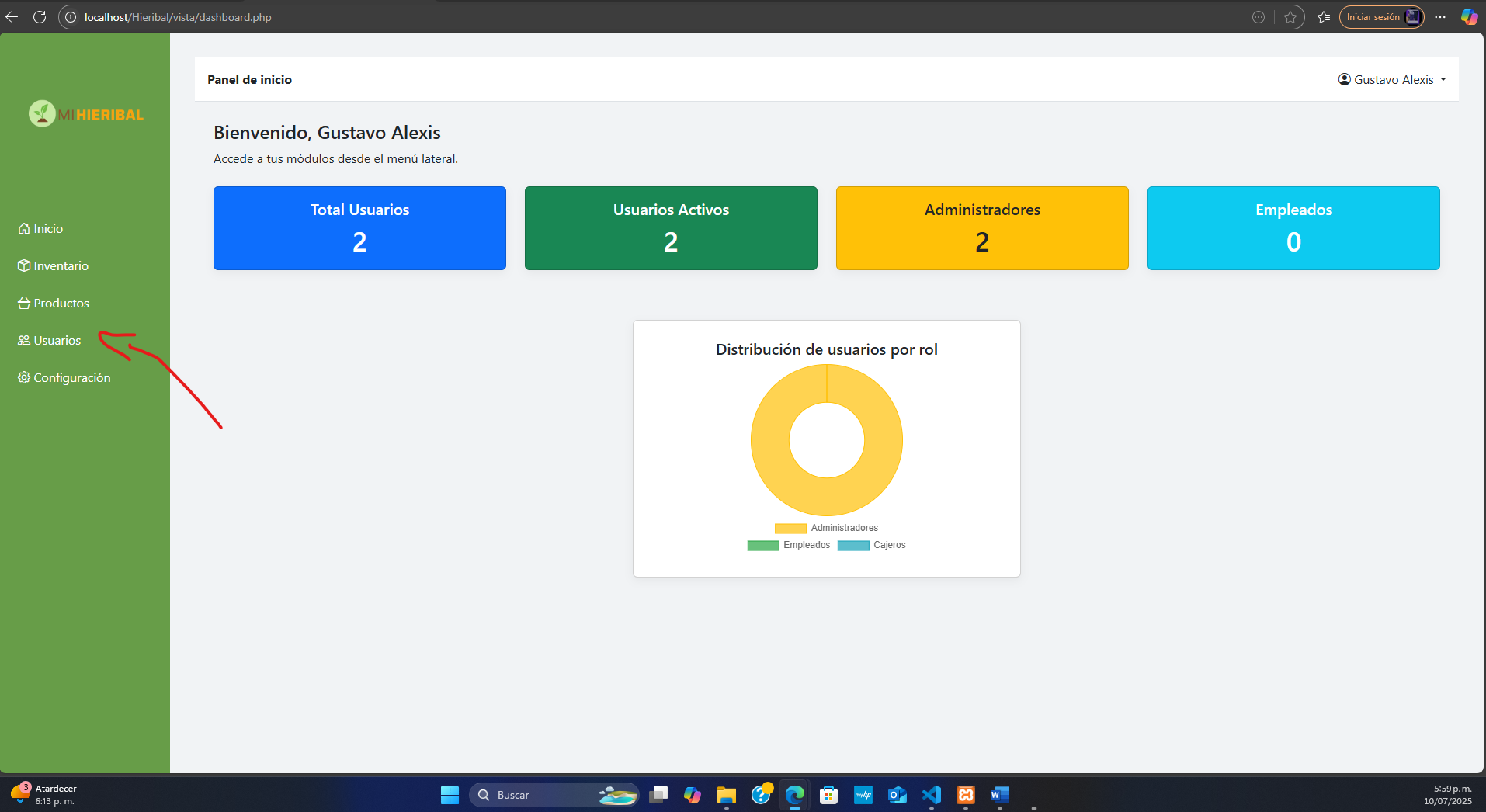
En esta vista se muestra el panel principal del usuario luego de iniciar sesión. Se despliegan estadísticas generales como el total de usuarios, los activos, los administradores y empleados registrados. También se visualiza una gráfica circular que representa la distribución por roles.

Actualmente, esta sección se encuentra en desarrollo. Se tiene previsto mejorar el diseño visual, agregar más tarjetas informativas, gráficos adicionales, accesos rápidos y métricas clave para hacer el dashboard más funcional e intuitivo.

****

**Acceso al módulo de gestión de usuarios**

Desde el panel lateral izquierdo, el usuario puede acceder al módulo "Usuarios", el cual permite administrar la información de los usuarios registrados en el sistema. Este acceso está disponible únicamente para los roles con permisos administrativos. Al hacer clic en esta opción, se redirige al listado de usuarios, donde se pueden realizar operaciones como ver, editar, eliminar, activar o crear nuevos usuarios.



**Módulo de gestión de usuarios**

En esta sección se visualiza una tabla con todos los usuarios registrados en el sistema. Cada fila muestra información clave como el nombre de usuario, rol, nombres, apellidos, correo, estado (activo/inactivo) y botones de acción.  
Desde este módulo es posible:

* Crear un nuevo usuario usando el botón verde **“+ Crear Usuario”**.
* Editar los datos de un usuario existente mediante el botón amarillo **“Editar”**.
* Eliminar usuarios con el botón rojo **“Eliminar”**.
* Buscar rápidamente registros usando el campo de búsqueda en la parte superior derecha.
* Navegar entre páginas de resultados gracias al sistema de paginación.

Este módulo fue construido con PHP, PDO y JavaScript, y permite una gestión completa y dinámica de los usuarios del sistema.  
