Registro Diario de Avances – Sprint 1 TaskManager

**Fecha:**

**Autor:**

**Versión:** 1.0

# 1. Introducción

Este informe describe el progreso alcanzado durante el Sprint 1 del proyecto Sistema de Gestión de Accesos y Roles. Durante este primer sprint, nos enfocamos en desarrollar la funcionalidad básica de inicio de sesión y autenticación, lo que constituye una parte crucial del sistema. El trabajo realizado permite a los usuarios registrarse, iniciar sesión y garantizar que sus credenciales estén seguras, siguiendo buenas prácticas de autenticación. El documento incluye las actividades realizadas, tareas completadas, evidencias de código, burndown chart y un análisis del rendimiento del equipo en relación con esta historia de usuario.

# 2. Lineamientos de Diseño

Durante el Sprint 1, el objetivo principal fue crear una interfaz, gestionar accesos y roles y registrar clientes para que los usuarios pudieran **registrarse** e **iniciar sesión** en la plataforma, garantizando que la autenticación fuera segura y fácil de usar. Las tareas principales fueron:

* Diseño de la interfaz de inicio de sesión para facilitar el acceso a los usuarios.
* Diseñar formulario en inicio de sesión
* Implementar autenticación backend
* Pruebas de inicio de sesión y seguridad
* Gestion de accesos y roles
* Crear módulo de gestión de roles
* Diseñar interfaz para asignación de permisos
* Validar asignación y control de acceso
* Registro de clientes en venta

- Crear formulario de registro de clientes

- Implementar validación y almacenamiento

- Pruebas de registro de clientes

# 3. Historias de Usuario 1.1: Asignar permisos por Área

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Backlog ID | Historia de Usuario | Tiempo Estimado (h) | Fecha de Inicio | Fecha de Finalización |
| HU-1 | Diseñar de la interfaz de Inicio de sesión | 4 h | 24/04/2025 | 24/04/2025 |
| HU-2 | Gestion de Accesos y Roles | 8 h | 28/04/2025 | 30/04/2025 |
| HU-3 | Registro de clientes de venta | 4 h | 30/04/2025 | 30/04/2025 |

# 4. Evidencias de Código

## 4.1 Historia de Usuario 1 — Diseñar formulario de Incio de sesión

## Archivo: TALLER\_DE\_PROYECTOS\_2/PHP RAP TRAVEL/views/login.php

<?php

session\_start();

require\_once('../config/db\_config.php');

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

// Obtener datos del formulario

$email = $\_POST['email'];

$password = $\_POST['password'];

// Buscar usuario en la base de datos

$stmt = $pdo->prepare("SELECT \* FROM usuarios WHERE correo = :email LIMIT 1");

$stmt->execute(['email' => $email]);

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

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

// Iniciar sesión si las credenciales son correctas

$\_SESSION['user\_id'] = $user['id'];

$\_SESSION['role\_id'] = $user['rol\_id'];

$\_SESSION['email'] = $user['correo'];

header('Location: index.php'); // Redirige al panel principal

exit;

} else {

$error = "❌ Credenciales incorrectas.";

}

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<title>Iniciar Sesión | Rap Travel Perú</title>

<!-- Fuentes y estilos externos -->

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;700&display=swap" rel="stylesheet">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css">

<style>

:root {

--color-primario-rojo: #D60000;

--color-fondo-verde: #0A4F32;

--color-blanco: #FFFFFF;

--color-texto-gris: #E0E0E0;

}

\* { margin: 0; padding: 0; box-sizing: border-box; }

body, html { height: 100%; font-family: 'Montserrat', sans-serif; overflow: hidden; }

.main-container {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

position: relative;

}

.background-image {

position: fixed;

top: 0; left: 0; width: 100%; height: 100%; z-index: -2;

background-image: url('https://images.pexels.com/photos/2929906/pexels-photo-2929906.jpeg');

background-size: cover;

background-position: center;

animation: kenBurns 25s ease-out infinite;

}

@keyframes kenBurns {

0% { transform: scale(1.0) translateX(0); }

50% { transform: scale(1.1) translateX(-2%); }

100% { transform: scale(1.0) translateX(0); }

}

.background-overlay {

position: fixed;

top: 0; left: 0; width: 100%; height: 100%;

background: linear-gradient(to top, rgba(10, 79, 50, 0.6), rgba(10, 79, 50, 0.3));

z-index: -1;

}

.login-box {

background: rgba(0, 0, 0, 0.7);

padding: 40px 30px;

border-radius: 15px;

width: 100%;

max-width: 380px;

text-align: center;

color: var(--color-blanco);

box-shadow: 0 10px 30px rgba(0, 0, 0, 0.5);

backdrop-filter: blur(10px);

border: 1px solid rgba(255, 255, 255, 0.1);

animation: fadeInFromBottom 0.8s ease-out forwards;

opacity: 0;

z-index: 1;

}

.logo { max-width: 200px; margin-bottom: 20px; }

.login-box h2 { font-weight: 700; margin-bottom: 10px; font-size: 1.8rem; }

.login-box p { color: var(--color-texto-gris); margin-bottom: 30px; font-weight: 500; }

.input-group { position: relative; margin-bottom: 25px; }

.input-group .icon { position: absolute; left: 15px; top: 50%; transform: translateY(-50%); color: var(--color-texto-gris); transition: color 0.3s ease; }

.input-field { width: 100%; padding: 15px 15px 15px 45px; background: rgba(255, 255, 255, 0.1); border: 1px solid rgba(255, 255, 255, 0.2); border-radius: 8px; color: var(--color-blanco); font-size: 1rem; transition: all 0.3s ease; }

.input-field::placeholder { color: var(--color-texto-gris); }

.input-field:focus { outline: none; border-color: var(--color-primario-rojo); box-shadow: 0 0 10px rgba(214, 0, 0, 0.5); }

.input-field:focus + .icon { color: var(--color-primario-rojo); }

.submit-btn { width: 100%; padding: 15px; background: var(--color-primario-rojo); border: none; border-radius: 8px; color: var(--color-blanco); font-size: 1.1rem; font-weight: 700; cursor: pointer; transition: all 0.3s ease; text-transform: uppercase; letter-spacing: 1px; }

.submit-btn:hover { background: #B80000; transform: translateY(-3px); box-shadow: 0 5px 15px rgba(214, 0, 0, 0.4); }

.submit-btn:active { transform: translateY(0); }

.extra-options { display: flex; justify-content: space-between; align-items: center; margin-top: 20px; font-size: 0.9rem; }

.extra-options a { color: var(--color-texto-gris); text-decoration: none; transition: color 0.3s ease; }

.extra-options a:hover { color: var(--color-primario-rojo); text-decoration: underline; }

@keyframes fadeInFromBottom {

from { opacity: 0; transform: translateY(30px); }

to { opacity: 1; transform: translateY(0); }

}

</style>

</head>

<body>

<div class="main-container">

<div class="background-image"></div>

<div class="background-overlay"></div>

<div class="login-box">

<img src="https://www.raptravelperu.com/wp-content/uploads/Rap-Travel-Peru-1.png" alt="Logo Rap Travel Perú" class="logo">

<h2>Bienvenido de Nuevo</h2>

<p>Inicia sesión para continuar tu aventura</p>

<?php if (isset($error)): ?>

<p style="color: red; font-weight: bold;"><?php echo $error; ?></p>

<?php endif; ?>

<form method="POST" action="login.php">

<div class="input-group">

<input type="email" name="email" class="input-field" placeholder="Correo electrónico" required>

<i class="fas fa-user icon"></i>

</div>

<div class="input-group">

<input type="password" name="password" class="input-field" placeholder="Contraseña" required>

<i class="fas fa-lock icon"></i>

</div>

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

<div class="extra-options">

<a href="register.php">Crear cuenta</a>

<a href="#">¿Olvidaste tu contraseña?</a>

</div>

</form>

</div>

</div>

</body>

</html>

## 4.2 Historia de Usuario 2 — Implementar Autenticación backend

Archivo:TALLER\_DE\_PROYECTOS\_2/PHP RAP TRAVEL/views

/register.php

<?php

session\_start();

require\_once('../config/db\_config.php'); // Conexión a la DB

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

$nombre = $\_POST['nombre'];

$apellido = $\_POST['apellido'];

$correo = $\_POST['correo'];

$contrasena = $\_POST['contrasena'];

// Rol por defecto: 2 (Ventas, por ejemplo). Asegúrate de tener un registro en la tabla roles con id=2

$rol\_id = 2;

// Verificar si el correo ya existe

$stmt = $pdo->prepare("SELECT \* FROM usuarios WHERE correo = :correo");

$stmt->execute(['correo' => $correo]);

if ($stmt->rowCount() > 0) {

$error = "❌ Este correo ya está registrado.";

} else {

// Encriptar la contraseña

$hash = password\_hash($contrasena, PASSWORD\_DEFAULT);

// Insertar el usuario

$stmt = $pdo->prepare("

INSERT INTO usuarios (nombre, apellido, correo, contrasena\_hash, rol\_id)

VALUES (:nombre, :apellido, :correo, :hash, :rol\_id)

");

$stmt->execute([

'nombre' => $nombre,

'apellido' => $apellido,

'correo' => $correo,

'hash' => $hash,

'rol\_id' => $rol\_id

]);

// Redirigir al login después de registrarse

header('Location: login.php');

exit;

}

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<title>Crear Cuenta | Rap Travel</title>

<style>

body { font-family: Arial, sans-serif; background: #0A4F32; display: flex; justify-content: center; align-items: center; height: 100vh; }

.box { background: rgba(0,0,0,0.7); padding: 30px; border-radius: 10px; color: white; width: 100%; max-width: 380px; }

h2 { text-align: center; margin-bottom: 20px; }

input { width: 100%; padding: 12px; margin-bottom: 15px; border-radius: 6px; border: 1px solid rgba(255,255,255,0.2); background: rgba(255,255,255,0.1); color: white; }

button { width: 100%; padding: 12px; background: #D60000; border: none; border-radius: 6px; font-weight: bold; cursor: pointer; color: white; }

button:hover { background: #B80000; }

p.error { color: #ff4d4d; font-weight: bold; text-align: center; }

.link-login { text-align: center; margin-top: 15px; }

.link-login a { color: #E0E0E0; text-decoration: none; }

.link-login a:hover { color: #D60000; text-decoration: underline; }

</style>

</head>

<body>

<div class="box">

<h2>Crear Cuenta</h2>

<?php if(isset($error)) echo "<p class='error'>$error</p>"; ?>

<form method="POST" action="register.php">

<input type="text" name="nombre" placeholder="Nombre" required>

<input type="text" name="apellido" placeholder="Apellido" required>

<input type="email" name="correo" placeholder="Correo electrónico" required>

<input type="password" name="contrasena" placeholder="Contraseña" required>

<button type="submit">Registrar</button>

</form>

<div class="link-login">

<p>¿Ya tienes cuenta? <a href="login.php">Inicia sesión</a></p>

</div>

</div>

</body>

</html>

Archivo:TALLER\_DE\_PROYECTOS\_2/PHP RAP TRAVEL/views

/registro\_cliente.php

<?php

session\_start();

require\_once('../config/db\_config.php');

// Verificar si el usuario tiene acceso (rol de gerente o ventas)

if (!isset($\_SESSION['user\_id'])) {

header('Location: login.php'); // Redirigir al login si no está autenticado

exit;

}

// Procesar el registro de cliente

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

$nombre = $\_POST['nombre'];

$apellido = $\_POST['apellido'];

$correo = $\_POST['correo'];

$telefono = $\_POST['telefono'];

$pais = $\_POST['pais'];

$fecha\_nacimiento = $\_POST['fecha\_nacimiento'];

// Insertar nuevo cliente en la base de datos

$stmt = $pdo->prepare("INSERT INTO clientes (nombre, apellido, correo, telefono, pais, fecha\_nacimiento)

VALUES (:nombre, :apellido, :correo, :telefono, :pais, :fecha\_nacimiento)");

$stmt->execute([

'nombre' => $nombre,

'apellido' => $apellido,

'correo' => $correo,

'telefono' => $telefono,

'pais' => $pais,

'fecha\_nacimiento' => $fecha\_nacimiento

]);

// Redirigir después de crear el cliente

header('Location: index.php');

exit;

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<title>Registrar Cliente</title>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">

<style>

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Source Sans Pro', 'Helvetica Neue', Arial, sans-serif;

background: #ecf0f5;

color: #333;

min-height: 100vh;

display: flex;

align-items: center;

justify-content: center;

padding: 20px;

}

.container {

width: 100%;

max-width: 600px;

}

.card {

background: #fff;

border-radius: 3px;

box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);

margin-bottom: 20px;

}

.card-header {

background: #3c8dbc;

color: #fff;

padding: 15px;

border-radius: 3px 3px 0 0;

border-bottom: 0;

}

.card-header h1 {

font-size: 24px;

font-weight: 400;

margin: 0;

}

.card-body {

padding: 25px;

}

.form-group {

margin-bottom: 20px;

}

.form-group label {

display: block;

margin-bottom: 8px;

color: #555;

font-weight: 600;

font-size: 14px;

}

.form-group label i {

margin-right: 8px;

color: #3c8dbc;

width: 16px;

text-align: center;

}

.form-control {

width: 100%;

padding: 12px 15px;

border: 1px solid #d2d6de;

border-radius: 3px;

font-size: 14px;

transition: border-color 0.15s ease-in-out, box-shadow 0.15s ease-in-out;

background: #fff;

color: #555;

}

.form-control:focus {

border-color: #3c8dbc;

outline: 0;

box-shadow: 0 0 0 0.2rem rgba(60, 141, 188, 0.25);

}

.form-control::placeholder {

color: #999;

}

.btn {

display: inline-block;

padding: 10px 20px;

font-size: 14px;

font-weight: 400;

text-align: center;

white-space: nowrap;

vertical-align: middle;

cursor: pointer;

border: 1px solid transparent;

border-radius: 3px;

transition: all 0.15s ease-in-out;

text-decoration: none;

}

.btn-primary {

background: #3c8dbc;

color: #fff;

border-color: #367fa9;

width: 100%;

font-size: 16px;

padding: 12px;

margin-top: 10px;

}

.btn-primary:hover {

background: #367fa9;

border-color: #204d74;

transform: translateY(-1px);

box-shadow: 0 2px 5px rgba(0,0,0,0.15);

}

.btn-primary:active {

transform: translateY(0);

box-shadow: 0 1px 3px rgba(0,0,0,0.12);

}

.btn-primary i {

margin-right: 8px;

}

.btn-secondary {

background: #fff;

color: #333;

border: 1px solid #d2d6de;

padding: 10px 20px;

margin-top: 10px;

width: 100%;

}

.btn-secondary:hover {

background: #f4f4f4;

border-color: #8c8c8c;

}

.row {

display: flex;

gap: 15px;

flex-wrap: wrap;

}

.col-6 {

flex: 1;

min-width: 0;

}

@media (max-width: 576px) {

.col-6 {

flex: 100%;

}

}

.alert {

padding: 15px;

margin-bottom: 20px;

border: 1px solid transparent;

border-radius: 3px;

}

.alert-success {

background: #d4edda;

border-color: #c3e6cb;

color: #155724;

}

/\* Input icons \*/

.input-group {

position: relative;

display: flex;

align-items: stretch;

width: 100%;

}

.input-group-icon {

position: absolute;

left: 12px;

top: 50%;

transform: translateY(-50%);

color: #999;

pointer-events: none;

z-index: 1;

}

.input-group .form-control {

padding-left: 38px;

}

</style>

</head>

<body>

<div class="container">

<div class="card">

<div class="card-header">

<h1><i class="fas fa-user-plus"></i> Registrar Nuevo Cliente</h1>

</div>

<div class="card-body">

<form method="POST" action="registro\_cliente.php">

<div class="row">

<div class="col-6">

<div class="form-group">

<label><i class="fas fa-user"></i> Nombre</label>

<div class="input-group">

<i class="fas fa-user input-group-icon"></i>

<input type="text" name="nombre" class="form-control" placeholder="Ingrese el nombre" required>

</div>

</div>

</div>

<div class="col-6">

<div class="form-group">

<label><i class="fas fa-user"></i> Apellido</label>

<div class="input-group">

<i class="fas fa-user input-group-icon"></i>

<input type="text" name="apellido" class="form-control" placeholder="Ingrese el apellido" required>

</div>

</div>

</div>

</div>

<div class="form-group">

<label><i class="fas fa-envelope"></i> Correo Electrónico</label>

<div class="input-group">

<i class="fas fa-envelope input-group-icon"></i>

<input type="email" name="correo" class="form-control" placeholder="ejemplo@correo.com" required>

</div>

</div>

<div class="form-group">

<label><i class="fas fa-phone"></i> Teléfono</label>

<div class="input-group">

<i class="fas fa-phone input-group-icon"></i>

<input type="tel" name="telefono" class="form-control" placeholder="+51 999 999 999" required>

</div>

</div>

<div class="row">

<div class="col-6">

<div class="form-group">

<label><i class="fas fa-globe"></i> País</label>

<div class="input-group">

<i class="fas fa-globe input-group-icon"></i>

<input type="text" name="pais" class="form-control" placeholder="Perú" required>

</div>

</div>

</div>

<div class="col-6">

<div class="form-group">

<label><i class="fas fa-calendar"></i> Fecha de Nacimiento</label>

<div class="input-group">

<i class="fas fa-calendar input-group-icon"></i>

<input type="date" name="fecha\_nacimiento" class="form-control" required>

</div>

</div>

</div>

</div>

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

<i class="fas fa-save"></i> Registrar Cliente

</button>

<a href="index.php" class="btn btn-secondary">

<i class="fas fa-arrow-left"></i> Volver

</a>

</form>

</div>

</div>

</div>

</body>

</html>

Archivo: TALLER\_DE\_PROYECTOS\_2/PHP RAP TRAVEL/views

/roles\_usuarios.php

<?php

session\_start();

require\_once('../config/db\_config.php');

// Verificar si el usuario tiene el rol de gerente (rol\_id = 1)

if (!isset($\_SESSION['user\_id']) || $\_SESSION['role\_id'] != 1) {

header('Location: index.php'); // Si no es gerente, redirigir

exit;

}

// Obtener los roles

$rolesStmt = $pdo->query("SELECT \* FROM roles");

$roles = $rolesStmt->fetchAll(PDO::FETCH\_ASSOC);

// Procesar la creación de un nuevo rol

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['create\_role'])) {

$role\_name = $\_POST['role\_name'];

$role\_description = $\_POST['role\_description'];

// Insertar nuevo rol

$stmt = $pdo->prepare("INSERT INTO roles (nombre, descripcion) VALUES (:nombre, :descripcion)");

$stmt->execute(['nombre' => $role\_name, 'descripcion' => $role\_description]);

// Redirigir para evitar que el formulario se envíe varias veces

header('Location: roles\_usuarios.php');

exit;

}

// Procesar la creación de un nuevo usuario

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['create\_user'])) {

$user\_name = $\_POST['user\_name'];

$user\_lastname = $\_POST['user\_lastname'];

$user\_email = $\_POST['user\_email'];

$user\_password = password\_hash($\_POST['user\_password'], PASSWORD\_DEFAULT);

$user\_role\_id = $\_POST['user\_role\_id'];

// Insertar nuevo usuario

$stmt = $pdo->prepare("INSERT INTO usuarios (nombre, apellido, correo, contrasena\_hash, rol\_id)

VALUES (:nombre, :apellido, :correo, :contrasena\_hash, :rol\_id)");

$stmt->execute([

'nombre' => $user\_name,

'apellido' => $user\_lastname,

'correo' => $user\_email,

'contrasena\_hash' => $user\_password,

'rol\_id' => $user\_role\_id

]);

// Redirigir después de crear el usuario

header('Location: roles\_usuarios.php');

exit;

}

?>

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<title>Gestionar Roles y Usuarios</title>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">

<style>

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Source Sans Pro', 'Helvetica Neue', Arial, sans-serif;

background: #ecf0f5;

color: #333;

min-height: 100vh;

padding: 20px;

}

.wrapper {

max-width: 1200px;

margin: 0 auto;

}

.page-header {

background: #fff;

padding: 20px 25px;

margin-bottom: 20px;

border-radius: 3px;

box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);

border-left: 4px solid #dd4b39;

}

.page-header h1 {

font-size: 28px;

font-weight: 400;

color: #333;

margin: 0;

}

.page-header h1 i {

margin-right: 10px;

color: #dd4b39;

}

.content-wrapper {

display: grid;

grid-template-columns: 1fr 1fr;

gap: 20px;

margin-bottom: 20px;

}

.card {

background: #fff;

border-radius: 3px;

box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);

overflow: hidden;

}

.card-header {

padding: 15px 20px;

border-bottom: 1px solid #f4f4f4;

background: #fafafa;

}

.card-header.primary {

background: #3c8dbc;

color: #fff;

border-bottom: 0;

}

.card-header.success {

background: #00a65a;

color: #fff;

border-bottom: 0;

}

.card-header.warning {

background: #f39c12;

color: #fff;

border-bottom: 0;

}

.card-header h2 {

font-size: 18px;

font-weight: 500;

margin: 0;

}

.card-header h2 i {

margin-right: 8px;

}

.card-body {

padding: 20px;

}

.form-group {

margin-bottom: 18px;

}

.form-group label {

display: block;

margin-bottom: 8px;

color: #555;

font-weight: 600;

font-size: 14px;

}

.form-group label i {

margin-right: 8px;

color: #3c8dbc;

width: 16px;

text-align: center;

}

.form-control {

width: 100%;

padding: 10px 12px;

border: 1px solid #d2d6de;

border-radius: 3px;

font-size: 14px;

transition: border-color 0.15s ease-in-out, box-shadow 0.15s ease-in-out;

background: #fff;

color: #555;

font-family: inherit;

}

.form-control:focus {

border-color: #3c8dbc;

outline: 0;

box-shadow: 0 0 0 0.2rem rgba(60, 141, 188, 0.25);

}

textarea.form-control {

min-height: 80px;

resize: vertical;

}

select.form-control {

cursor: pointer;

appearance: none;

background-image: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org/2000/svg' width='12' height='12' viewBox='0 0 12 12'%3E%3Cpath fill='%23333' d='M6 9L1 4h10z'/%3E%3C/svg%3E");

background-repeat: no-repeat;

background-position: right 12px center;

padding-right: 35px;

}

.btn {

display: inline-block;

padding: 10px 20px;

font-size: 14px;

font-weight: 400;

text-align: center;

white-space: nowrap;

vertical-align: middle;

cursor: pointer;

border: 1px solid transparent;

border-radius: 3px;

transition: all 0.15s ease-in-out;

text-decoration: none;

}

.btn-primary {

background: #3c8dbc;

color: #fff;

border-color: #367fa9;

width: 100%;

font-size: 15px;

padding: 11px;

}

.btn-primary:hover {

background: #367fa9;

border-color: #204d74;

transform: translateY(-1px);

box-shadow: 0 2px 5px rgba(0,0,0,0.15);

}

.btn-success {

background: #00a65a;

color: #fff;

border-color: #008d4c;

width: 100%;

font-size: 15px;

padding: 11px;

}

.btn-success:hover {

background: #008d4c;

border-color: #007d42;

transform: translateY(-1px);

box-shadow: 0 2px 5px rgba(0,0,0,0.15);

}

.btn-warning {

background: #f39c12;

color: #fff;

border-color: #e08e0b;

width: 100%;

font-size: 15px;

padding: 11px;

}

.btn-warning:hover {

background: #e08e0b;

border-color: #c87f0a;

transform: translateY(-1px);

box-shadow: 0 2px 5px rgba(0,0,0,0.15);

}

.btn-default {

background: #fff;

color: #333;

border: 1px solid #d2d6de;

padding: 10px 20px;

}

.btn-default:hover {

background: #f4f4f4;

border-color: #8c8c8c;

}

.btn i {

margin-right: 8px;

}

/\* Lista de roles \*/

.roles-list {

list-style: none;

padding: 0;

margin: 0;

}

.role-item {

padding: 15px;

border-bottom: 1px solid #f4f4f4;

transition: background 0.2s;

}

.role-item:last-child {

border-bottom: 0;

}

.role-item:hover {

background: #f9f9f9;

}

.role-name {

font-weight: 600;

color: #333;

font-size: 15px;

margin-bottom: 5px;

display: flex;

align-items: center;

}

.role-name i {

margin-right: 10px;

color: #3c8dbc;

font-size: 16px;

}

.role-description {

color: #666;

font-size: 13px;

padding-left: 26px;

}

.empty-state {

text-align: center;

padding: 30px;

color: #999;

}

.empty-state i {

font-size: 48px;

margin-bottom: 15px;

opacity: 0.5;

}

.back-link {

background: #fff;

padding: 15px 20px;

border-radius: 3px;

box-shadow: 0 1px 3px rgba(0,0,0,0.12), 0 1px 2px rgba(0,0,0,0.24);

display: inline-block;

text-decoration: none;

color: #333;

transition: all 0.2s;

}

.back-link:hover {

background: #f4f4f4;

transform: translateX(-3px);

}

.back-link i {

margin-right: 8px;

}

.full-width {

grid-column: 1 / -1;

}

@media (max-width: 992px) {

.content-wrapper {

grid-template-columns: 1fr;

}

}

@media (max-width: 576px) {

body {

padding: 10px;

}

.page-header {

padding: 15px;

}

.page-header h1 {

font-size: 22px;

}

.card-body {

padding: 15px;

}

}

/\* Animaciones \*/

@keyframes slideIn {

from {

opacity: 0;

transform: translateY(10px);

}

to {

opacity: 1;

transform: translateY(0);

}

}

.card {

animation: slideIn 0.3s ease-out;

}

</style>

</head>

<body>

<div class="wrapper">

<div class="page-header">

<h1><i class="fas fa-users-cog"></i> Gestionar Roles y Usuarios</h1>

</div>

<div class="content-wrapper">

<!-- Formulario para crear un nuevo rol -->

<div class="card">

<div class="card-header primary">

<h2><i class="fas fa-user-tag"></i> Crear Nuevo Rol</h2>

</div>

<div class="card-body">

<form method="POST" action="roles\_usuarios.php">

<div class="form-group">

<label><i class="fas fa-tag"></i> Nombre del Rol</label>

<input type="text" name="role\_name" class="form-control" placeholder="Ej: Vendedor, Supervisor, etc." required>

</div>

<div class="form-group">

<label><i class="fas fa-align-left"></i> Descripción del Rol</label>

<textarea name="role\_description" class="form-control" placeholder="Describe las responsabilidades de este rol..." required></textarea>

</div>

<button type="submit" name="create\_role" class="btn btn-primary">

<i class="fas fa-plus-circle"></i> Crear Rol

</button>

</form>

</div>

</div>

<!-- Lista de Roles Existentes -->

<div class="card">

<div class="card-header primary">

<h2><i class="fas fa-list"></i> Roles Existentes</h2>

</div>

<div class="card-body" style="padding: 0;">

<?php if (count($roles) > 0): ?>

<ul class="roles-list">

<?php foreach ($roles as $role): ?>

<li class="role-item">

<div class="role-name">

<i class="fas fa-shield-alt"></i>

<?php echo htmlspecialchars($role['nombre']); ?>

</div>

<div class="role-description">

<?php echo htmlspecialchars($role['descripcion']); ?>

</div>

</li>

<?php endforeach; ?>

</ul>

<?php else: ?>

<div class="empty-state">

<i class="fas fa-folder-open"></i>

<p>No hay roles registrados</p>

</div>

<?php endif; ?>

</div>

</div>

<!-- Formulario para crear un nuevo usuario -->

<div class="card full-width">

<div class="card-header success">

<h2><i class="fas fa-user-plus"></i> Crear Nuevo Usuario</h2>

</div>

<div class="card-body">

<form method="POST" action="roles\_usuarios.php">

<div style="display: grid; grid-template-columns: repeat(auto-fit, minmax(250px, 1fr)); gap: 15px;">

<div class="form-group">

<label><i class="fas fa-user"></i> Nombre</label>

<input type="text" name="user\_name" class="form-control" placeholder="Nombre del usuario" required>

</div>

<div class="form-group">

<label><i class="fas fa-user"></i> Apellido</label>

<input type="text" name="user\_lastname" class="form-control" placeholder="Apellido del usuario" required>

</div>

<div class="form-group">

<label><i class="fas fa-envelope"></i> Correo Electrónico</label>

<input type="email" name="user\_email" class="form-control" placeholder="ejemplo@correo.com" required>

</div>

<div class="form-group">

<label><i class="fas fa-lock"></i> Contraseña</label>

<input type="password" name="user\_password" class="form-control" placeholder="Mínimo 8 caracteres" required>

</div>

<div class="form-group">

<label><i class="fas fa-user-tag"></i> Rol del Usuario</label>

<select name="user\_role\_id" class="form-control" required>

<option value="">Seleccione un rol</option>

<?php foreach ($roles as $role): ?>

<option value="<?php echo $role['id']; ?>"><?php echo htmlspecialchars($role['nombre']); ?></option>

<?php endforeach; ?>

</select>

</div>

</div>

<button type="submit" name="create\_user" class="btn btn-success">

<i class="fas fa-user-check"></i> Crear Usuario

</button>

</form>

</div>

</div>

</div>

<a href="index.php" class="back-link">

<i class="fas fa-arrow-left"></i> Volver al Panel Principal

</a>

</div>

</body>

</html>

## 4.3 Historia de Usuario 3 — Prueba de Inicio de sesión y seguridad

Archivo:**TALLER\_DE\_PROYECTOS\_2/PHP RAP TRAVEL/views/login.php**

<?php

// La parte de verificación de credenciales y autenticación ya ha sido testeada.

// En el caso de una contraseña incorrecta, el sistema muestra un mensaje de error.

if (isset($error)) {

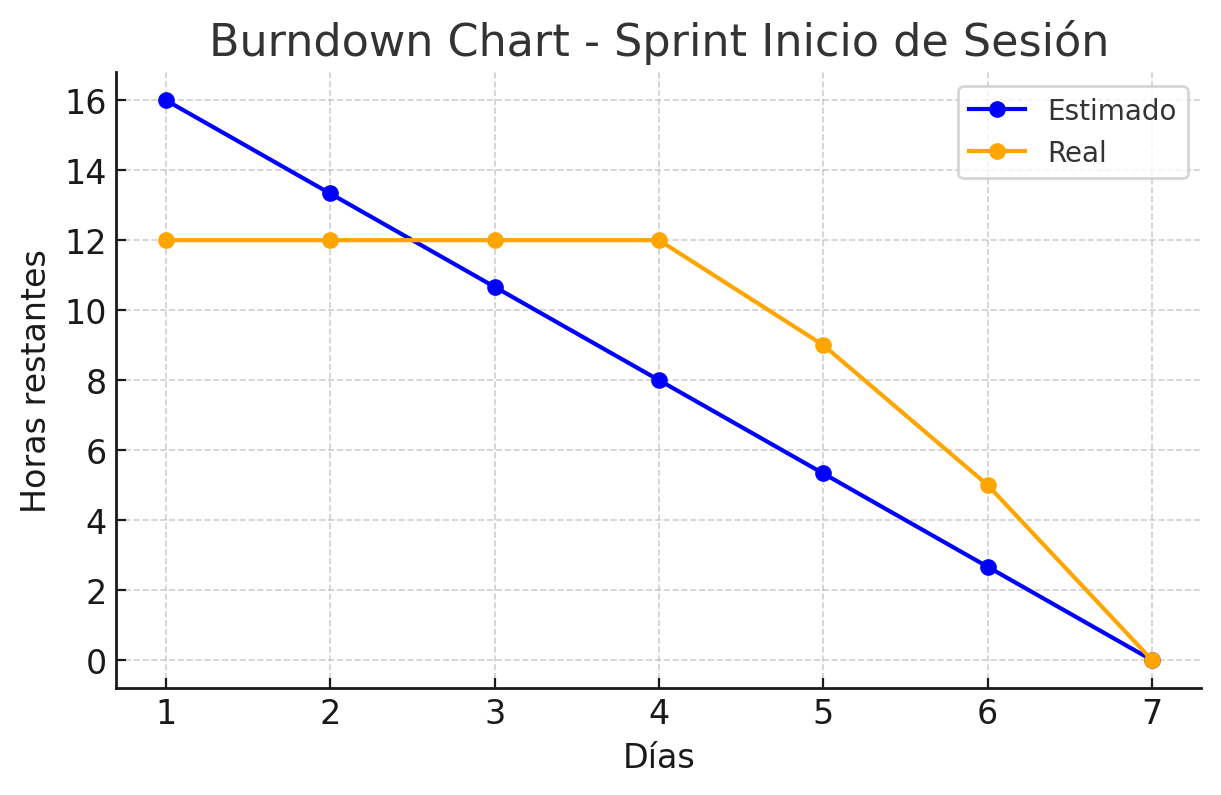
echo "<p style='color: red; font-weight: bold;'>$error</p>";

}

?>

# 5. Burndown Chart

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Backlog ID | Historia de Usuario | Tiempo Estimado | 24/09/2025 | 25/09/2025 | 26/09/2025 | 27/09/2025 | 28/09/2025 | 29/09/2025 | 30/09/2025 |
| **Dia1** | **Dia2** | **Dia3** | **Dia4** | **Dia5** | **Dia6** | **Dia7** |
| HU-1 | Diseñar de la interfaz de Inicio de sesión | 4 | 4 |  |  |  |  |  |  |
| HU-2 | Gestion de Accesos y Roles | 8 |  |  |  |  | 3 | 4 | 1 |
| HU-3 | Registro de clientes de venta | 4 |  |  |  |  |  |  | 4 |
| Tiempo de trabajo | | 16 | 4 | 0 | 0 | 0 | 3 | 4 | 5 |



# 6. Análisis del Sprint

Las tareas realizadas hasta ahora incluyen el diseño y la implementación de las siguientes funcionalidades:

* **PPA001**: Diseñar formulario de inicio de sesión
* Tarea completada por Rodrigo en el plazo estimado de 4 horas.
* **PPA002**: Implementar autenticación backend
* Rodrigo completó la implementación del backend de autenticación en 8 horas.
* **PPA003**: Pruebas de inicio de sesión y seguridad
* Fabricio Realizo pruebas exhaustivas para verificar la seguridad del inicio de sesión.
* **PPA004**: Crear módulo de gestión de roles
* Carlos desarrolló el módulo de roles y asignación de permisos en 10 horas.
* **PPA005**: Diseñar interfaz para asignación de permisos
* Carlos diseñó la interfaz para asignar permisos en 6 horas.
* **PPA006**: Validar asignación y control de acceso
* Se validó que los permisos sean correctamente asignados según el rol del usuario.
* **PPA007**: Crear formulario de registro de clientes
* Pacsi desarrolló el formulario de registro de clientes en 4 horas.
* **PPA008**: Implementar validación y almacenamiento
* Pacsi completó la implementación de la validación y almacenamiento de datos de clientes.
* **PPA009**: Pruebas de registro de clientes
* Se realizaron pruebas para asegurar que los datos de clientes fueran registrados correctamente.

# 7. Conclusiones

El Sprint 1 ha sido exitoso en la implementación de las funcionalidades clave, con el avance de todas las tareas dentro de lo esperado. El equipo sigue trabajando de acuerdo a los plazos establecidos, y el progreso continuo asegura que las tareas pendientes serán completadas conforme avance el proyecto.

Es crucial resaltar que, a pesar de los desafíos iniciales, el sistema de autenticación implementado asegura la integridad y privacidad de los usuarios, lo que es fundamental para el éxito de cualquier plataforma en línea. A futuro, se recomienda mantener un enfoque similar en la planificación y evaluación de tareas, asegurando la flexibilidad necesaria para adaptarse a los imprevistos sin perder el foco en los objetivos.