Registro Diario de Avances – Sprint 1 SISTEC

**Fecha:01/10/2025**

**Autor:Raurau Quispe, Juan Boris; Munares Chamorro, Juan Carlos**

**Versión:** 1.0

# 1. Introducción

Este documento detalla el seguimiento diario del Sprint 1 del proyecto SISTEC, centrado en el desarrollo de funcionalidades básicas. Se incluyen actividades diarias, tareas completadas, evidencias de código, un burndown chart y un análisis del rendimiento del equipo.

# 2. Lineamientos de Diseño

Desarrollar una primera versión funcional de la aplicación SISTEC que permita a los usuarios:

* Registrarse e iniciar sesión.
* Crear nuevas tareas.
* Marcar tareas como completadas.
* Almacenar los datos de manera persistente con localStorage.

# 3. Historias de Usuario y Tareas

| Backlog ID | Historia de Usuario | Tiempo Estimado | Fecha de Inicio | Fecha de Finalización |
| --- | --- | --- | --- | --- |
| HU-1.1 | Registrar Entradas y Salidas Automáticas | 3.5hrs | 23/09/2025 | 23/09/2025 |
| HU-1.2 | Consultar Historial de Movimientos | 3 hrs | 24/09/2025 | 24/09/2025 |
| HU-1.3 | Registro de equipos | 5 hrs | 25/09/2025 | 25/09/2025 |
| HU-2.1 | Detectar Movimientos No Autorizados | 5 hrs | 26/09/2025 | 26/09/2025 |
| HU-2.2 | Validación de Equipos No Autorizados | 5 hrs | 27/09/2025 | 27/09/2025 |
| HU-2.3 | Reportes de Movimientos Sospechosos | 6 hrs | 28/09/2025 | 28/09/2025 |
| HU-3.1 | Generar Reportes de Inventario | 3 hrs | 29/09/2025 | 29/09/2025 |
| HU-3.2 | Reportes de Alertas por Turno | 4 hrs | 30/09/2025 | 30/09/2025 |
| HU-3.3 | Reportes Automáticos de Inventario | 3 hrs | 01/10/2025 | 01/10/2025 |

# 4. Evidencias de Código

## 1.1 Historia de Usuario 1 — Registrar Entradas y Salidas Automáticas de sesión

Archivo:

CREATE TABLE `alertas` (

`id` int(11) NOT NULL,

`mensaje` text NOT NULL,

`ubicacion` varchar(100) DEFAULT NULL,

`nivel` enum('peligro','cuidado') DEFAULT 'cuidado',

`fecha` datetime DEFAULT current\_timestamp(),

`numero\_tarjeta\_rfid` varchar(100) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

Archivo: alertas/registrar.php

<?php

require\_once '../db.php';

header('Content-Type: application/json');

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

$rfid = $\_POST['numero\_tarjeta\_rfid'] ?? '';

if (empty($rfid)) {

http\_response\_code(400);

echo json\_encode(["success" => false, "error" => "❌ número de tarjeta RFID no proporcionado."]);

exit;

}

$stmt = $conn->prepare("SELECT nombre, ubicacion FROM equipos WHERE numero\_tarjeta\_rfid = ?");

$stmt->bind\_param("s", $rfid);

$stmt->execute();

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

if ($equipo = $result->fetch\_assoc()) {

$nombre = $equipo['nombre'];

$ubicacion = $equipo['ubicacion'];

$mensaje = "📟 {$nombre} ha salido del laboratorio";

$nivel = 'peligro'; // ← EQUIPO REGISTRADO

$insert = $conn->prepare("INSERT INTO alertas (mensaje, nivel, ubicacion) VALUES (?, ?, ?)");

$insert->bind\_param("sss", $mensaje, $nivel, $ubicacion);

} else {

$mensaje = "⚠️ Tarjeta no registrada (UID: $rfid) detectada.";

$nivel = 'cuidado'; // ← NO REGISTRADO

$ubicacion = "Desconocido";

$insert = $conn->prepare("INSERT INTO alertas (mensaje, nivel, ubicacion) VALUES (?, ?, ?)");

$insert->bind\_param("sss", $mensaje, $nivel, $ubicacion);

}

if ($insert->execute()) {

echo json\_encode(["success" => true, "message" => $mensaje]);

} else {

echo json\_encode(["success" => false, "error" => "❌ Error al registrar la alerta."]);

}

} else {

http\_response\_code(405);

echo json\_encode(["success" => false, "error" => "❌ Método no permitido. Usa POST."]);

}

Archivo: alertas/listar.php

<?php

require\_once '../db.php';

header('Content-Type: application/json');

$response = [

'success' => false,

'data' => [],

'error' => null

];

$sql = "SELECT id, mensaje, nivel, ubicacion, fecha FROM alertas ORDER BY fecha DESC";

$result = $conn->query($sql);

if ($result) {

while ($row = $result->fetch\_assoc()) {

$response['data'][] = $row;

}

$response['success'] = true;

} else {

$response['error'] = "Error al obtener alertas: " . $conn->error;

}

echo json\_encode($response, JSON\_UNESCAPED\_UNICODE);

## 1.2 Historia de Usuario 1 — Consultar Historial de Movimientos

Archivo: views/alertas.php

<div class="container">

<h2>📡 Alertas de Salida de Equipos</h2>

<a href="dashboard.php" class="btn-volver">← Volver</a>

<p>Esta lista se actualiza automáticamente con las alertas enviadas por el módulo RFID/NFC.</p>

<table id="tabla-alertas">

<thead>

<tr>

<th>ID</th>

<th>Mensaje</th>

<th>Laboratorio</th>

<th>Nivel</th>

<th>Fecha</th>

</tr>

</thead>

<tbody>

<!-- Se llenará con JavaScript -->

</tbody>

</table>

</div>

<script>

function cargarAlertas() {

fetch('../php/alertas/listar.php')

.then(response => response.json())

.then(data => {

const tbody = document.querySelector('#tabla-alertas tbody');

tbody.innerHTML = ''; // Limpiar contenido actual

if (data.success) {

data.data.forEach(alerta => {

const fila = document.createElement('tr');

// Normalizar nivel (puede venir en mayúsculas)

const nivel = (alerta.nivel || 'info').toLowerCase();

const claseNivel = ['peligro', 'cuidado', 'info'].includes(nivel) ? nivel : 'info';

const mensaje = alerta.mensaje || 'Sin mensaje';

const ubicacion = alerta.ubicacion || 'No especificado';

const fecha = alerta.fecha || 'Desconocida';

fila.className = claseNivel;

fila.innerHTML = `

<td>${alerta.id}</td>

<td>${mensaje}</td>

<td>${ubicacion}</td>

<td>${nivel.charAt(0).toUpperCase() + nivel.slice(1)}</td>

<td>${fecha}</td>

`;

tbody.appendChild(fila);

});

} else {

console.error('Error al cargar alertas:', data.error);

}

})

.catch(error => console.error('Error al cargar alertas:', error));

}

cargarAlertas();

setInterval(cargarAlertas, 5000);

</script>

<?php include '../includes/footer.php'; ?>

## 1.3 Historia de Usuario 1 — Registro de equipos

Archivo:

CREATE TABLE `equipos` (

`id` int(11) NOT NULL,

`nombre` varchar(100) NOT NULL,

`ubicacion` varchar(100) NOT NULL,

`estado` enum('Autorizado','No autorizado') NOT NULL DEFAULT 'Autorizado',

`fecha\_registro` timestamp NOT NULL DEFAULT current\_timestamp(),

`numero\_tarjeta\_rfid` varchar(100) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_general\_ci;

Archivo: equipos/actualizar.php

<?php

include '../db.php';

header('Content-Type: application/json');

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

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

$nombre = strtoupper(trim($\_POST['nombre']));

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

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

$numeroTarjeta = strtoupper(trim($\_POST['numero\_tarjeta\_rfid']));

if (!preg\_match('/^[A-Z0-9\s]+$/', $nombre)) {

echo json\_encode([

'success' => false,

'error' => '❌ El nombre solo puede contener letras, números y espacios.'

]);

exit();

}

if (!preg\_match('/^[A-Z0-9]+$/', $numeroTarjeta)) {

echo json\_encode([

'success' => false,

'error' => '❌ El número de tarjeta RFID solo puede contener letras y números.'

]);

exit();

}

// Verificar duplicado excluyendo el mismo ID

$verificar = $conn->prepare("SELECT id FROM equipos WHERE numero\_tarjeta\_rfid = ? AND id != ?");

$verificar->bind\_param("si", $numeroTarjeta, $id);

$verificar->execute();

$verificar->store\_result();

if ($verificar->num\_rows > 0) {

echo json\_encode([

'success' => false,

'error' => '⚠️ Otro equipo ya tiene esa tarjeta RFID.'

]);

exit();

}

$stmt = $conn->prepare("UPDATE equipos SET nombre = ?, ubicacion = ?, estado = ?, numero\_tarjeta\_rfid = ? WHERE id = ?");

$stmt->bind\_param("ssssi", $nombre, $ubicacion, $estado, $numeroTarjeta, $id);

if ($stmt->execute()) {

echo json\_encode([

'success' => true,

'message' => '✅ Equipo actualizado correctamente.'

]);

} else {

echo json\_encode([

'success' => false,

'error' => '❌ Error al actualizar el equipo: ' . $stmt->error

]);

}

}

?>

Archivo: equipos/agregar.php

<?php

require\_once '../db.php';

if (!isset($\_POST['numero\_tarjeta\_rfid'], $\_POST['nombre'], $\_POST['ubicacion'], $\_POST['estado'])) {

echo "❌ Faltan campos obligatorios.";

exit();

}

$numero\_tarjeta\_rfid = trim($\_POST['numero\_tarjeta\_rfid']);

$nombre = strtoupper(trim($\_POST['nombre']));

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

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

// Verificar si ya existe un equipo con ese número de tarjeta RFID

$verificar = $conn->prepare("SELECT id FROM equipos WHERE numero\_tarjeta\_rfid = ?");

$verificar->bind\_param("s", $numero\_tarjeta\_rfid);

$verificar->execute();

$verificar->store\_result();

if ($verificar->num\_rows > 0) {

echo "⚠️ Ya existe un equipo con esa tarjeta RFID.";

exit();

}

// Insertar nuevo equipo

$stmt = $conn->prepare("INSERT INTO equipos (numero\_tarjeta\_rfid, nombre, ubicacion, estado, fecha\_registro) VALUES (?, ?, ?, ?, NOW())");

$stmt->bind\_param("ssss", $numero\_tarjeta\_rfid, $nombre, $ubicacion, $estado);

if ($stmt->execute()) {

echo "✅ Equipo agregado exitosamente.";

} else {

echo "❌ Error al guardar el equipo: " . $stmt->error;

}

Archivo: equipos/eliminar.php

<?php

include '../db.php';

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

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

$stmt = $conn->prepare("DELETE FROM equipos WHERE id = ?");

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

if ($stmt->execute()) {

header("Location: ../../views/equipos.php?mensaje=eliminacion\_exitosa");

exit();

} else {

echo "❌ Error al eliminar equipo: " . $stmt->error;

}

}

?>

A<?php

include '../db.php';

header('Content-Type: application/json');

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

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

$stmt = $conn->prepare("DELETE FROM equipos WHERE id = ?");

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

if ($stmt->execute()) {

echo json\_encode([

'success' => true,

'message' => '✅ Equipo eliminado correctamente.'

]);

} else {

echo json\_encode([

'success' => false,

'error' => '❌ Error al eliminar el equipo: ' . $stmt->error

]);

}

} else {

echo json\_encode([

'success' => false,

'error' => '❌ Método no permitido.'

]);

}

?>

Archivo: equipos/listar.php

<?php

include '../db.php';

header('Content-Type: application/json');

$sql = "SELECT id, numero\_tarjeta\_rfid, nombre, ubicacion, estado, fecha\_registro FROM equipos ORDER BY id DESC";

$resultado = $conn->query($sql);

$equipos = [];

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

$equipos[] = $fila;

}

echo json\_encode($equipos);

?>

## 2.1 Historia de Usuario 2 — Detectar Movimientos No Autorizados

Archivo: alertas/alertas.php

<?php

require\_once '../db.php';

header('Content-Type: application/json');

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

$rfid = $\_POST['numero\_tarjeta\_rfid'] ?? '';

if (empty($rfid)) {

http\_response\_code(400);

echo json\_encode(["success" => false, "error" => "❌ número de tarjeta RFID no proporcionado."]);

exit;

}

$stmt = $conn->prepare("SELECT nombre, ubicacion FROM equipos WHERE numero\_tarjeta\_rfid = ?");

$stmt->bind\_param("s", $rfid);

$stmt->execute();

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

if ($equipo = $result->fetch\_assoc()) {

$nombre = $equipo['nombre'];

$ubicacion = $equipo['ubicacion'];

$mensaje = "📟 {$nombre} ha salido del laboratorio";

$nivel = 'peligro'; // ← EQUIPO REGISTRADO

$insert = $conn->prepare("INSERT INTO alertas (mensaje, nivel, ubicacion) VALUES (?, ?, ?)");

$insert->bind\_param("sss", $mensaje, $nivel, $ubicacion);

} else {

$mensaje = "⚠️ Tarjeta no registrada (UID: $rfid) detectada.";

$nivel = 'cuidado'; // ← NO REGISTRADO

$ubicacion = "Desconocido";

$insert = $conn->prepare("INSERT INTO alertas (mensaje, nivel, ubicacion) VALUES (?, ?, ?)");

$insert->bind\_param("sss", $mensaje, $nivel, $ubicacion);

}

if ($insert->execute()) {

echo json\_encode(["success" => true, "message" => $mensaje]);

} else {

echo json\_encode(["success" => false, "error" => "❌ Error al registrar la alerta."]);

}

} else {

http\_response\_code(405);

echo json\_encode(["success" => false, "error" => "❌ Método no permitido. Usa POST."]);

}

## 2.2 Historia de Usuario 2 — Validación de Equipos No Autorizados

Archivo: views/equipos.php

<?php

require\_once '../php/db.php';

require\_once '../includes/verificar\_sesion.php';

if ($\_SESSION['rol'] !== 'admin' && $\_SESSION['rol'] !== 'tecnico') {

header('Location: ../index.php');

exit;

}

require\_once '../includes/roles.php';

include '../includes/header.php';

?>

<div class="container">

<h2>📟 Gestión de Equipos</h2>

<a href="dashboard.php" class="btn-volver">← Volver</a>

<div id="mensaje-alerta" style="display:none; padding:10px; border-radius:5px; margin-top:10px; position:relative;"></div>

<!-- FORMULARIO -->

<form id="formEquipo" method="POST" action="../php/equipos/agregar.php">

<input type="hidden" name="id" id="equipo\_id">

<label>Número de Tarjeta RFID:</label>

<input type="text" name="numero\_tarjeta\_rfid" id="numero\_tarjeta\_rfid" required pattern="[A-Za-z0-9]+" title="Solo letras y números">

<label>Nombre del equipo (nombre y número):</label>

<input type="text" name="nombre" id="nombre" required pattern="[A-Za-z0-9 ]+" title="Solo letras y números">

<label>Laboratorio:</label>

<select name="ubicacion" id="ubicacion" required>

<option value="">Seleccione...</option>

<?php

$laboratorios = ['101','102','201','202','203','204','301','302','303','304','401','402','403','404','501','502','503','504','601','602','603','604','701','702','703','704','801','802','803','804'];

foreach ($laboratorios as $lab) {

echo "<option value='Laboratorio $lab'>Laboratorio $lab</option>";

}

?>

</select>

<label>Estado:</label>

<select name="estado" id="estado" required>

<option value="Autorizado">Autorizado</option>

<option value="No autorizado">No autorizado</option>

</select>

<button type="submit" id="btnGuardar">Guardar Equipo</button>

<button type="button" onclick="cancelarEdicion()" style="display:none;" id="btnCancelar">Cancelar</button>

</form>

<!-- TABLA -->

<h3>📋 Lista de Equipos</h3>

<table>

<thead>

<tr>

<th>ID</th>

<th>Número de Tarjeta RFID</th>

<th>Nombre</th>

<th>Ubicación</th>

<th>Estado</th>

<th>Fecha</th>

<th>Acciones</th>

</tr>

</thead>

<tbody>

<?php

$resultado = $conn->query("SELECT \* FROM equipos ORDER BY id DESC");

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

echo "<tr>

<td>{$fila['id']}</td>

<td>{$fila['numero\_tarjeta\_rfid']}</td>

<td>{$fila['nombre']}</td>

<td>{$fila['ubicacion']}</td>

<td>{$fila['estado']}</td>

<td>{$fila['fecha\_registro']}</td>

<td>

<button onclick='editarEquipo(" . json\_encode($fila) . ")'>✏️</button>

<form method='POST' action='../php/equipos/eliminar.php' style='display:inline' onsubmit='return false;'>

<input type='hidden' name='id' value='{$fila['id']}'>

<button type='submit' onclick='confirmarEliminacion(this.closest(\"form\"))'>🗑️</button>

</form>

</td>

</tr>";

}

?>

</tbody>

</table>

</div>

<!-- MODAL -->

<div id="confirmModal" style="display:none; position:fixed; top:0; left:0; width:100%; height:100%; background-color:rgba(0,0,0,0.4); z-index:999;">

<div style="background:white; padding:30px; border-radius:10px; width:90%; max-width:400px; margin:100px auto; text-align:center;">

<h3 id="modalTitle">¿Estás seguro?</h3>

<p id="modalMessage">Esta acción no se puede deshacer.</p>

<button id="confirmBtn" style="background-color:#28a745; color:white; border:none; padding:10px 20px; margin-right:10px; border-radius:5px;">Sí</button>

<button onclick="closeModal()" style="background-color:#dc3545; color:white; border:none; padding:10px 20px; border-radius:5px;">Cancelar</button>

</div>

</div>

<script>

let currentForm = null;

function editarEquipo(equipo) {

document.getElementById('equipo\_id').value = equipo.id;

document.getElementById('numero\_tarjeta\_rfid').value = equipo.numero\_tarjeta\_rfid;

document.getElementById('nombre').value = equipo.nombre;

document.getElementById('ubicacion').value = equipo.ubicacion;

document.getElementById('estado').value = equipo.estado;

document.getElementById('formEquipo').action = "../php/equipos/actualizar.php";

document.getElementById('btnGuardar').textContent = "Actualizar Equipo";

document.getElementById('btnCancelar').style.display = "inline";

}

function cancelarEdicion() {

document.getElementById('formEquipo').reset();

document.getElementById('formEquipo').action = "../php/equipos/agregar.php";

document.getElementById('btnGuardar').textContent = "Guardar Equipo";

document.getElementById('btnCancelar').style.display = "none";

}

function closeModal() {

document.getElementById('confirmModal').style.display = 'none';

currentForm = null;

}

function showModal(message, form) {

currentForm = form;

document.getElementById('modalMessage').textContent = message;

document.getElementById('confirmModal').style.display = 'block';

}

function confirmarEliminacion(form) {

showModal('¿Estás seguro de que deseas eliminar este equipo?', form);

}

document.getElementById('confirmBtn').addEventListener('click', function () {

if (currentForm) currentForm.submit();

});

document.getElementById('formEquipo').addEventListener('submit', function(e) {

e.preventDefault();

const mensaje = document.getElementById('mensaje-alerta');

const formData = new FormData(this);

const isUpdate = this.action.includes("actualizar.php");

fetch(this.action, {

method: 'POST',

body: formData

})

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

.then(res => {

mensaje.style.display = 'block';

if (res.includes("⚠️")) {

mensaje.style.backgroundColor = '#f8d7da';

mensaje.style.color = '#721c24';

mensaje.innerHTML = `❌ ${res}` +

'<button onclick="this.parentElement.style.display=\'none\'" style="position:absolute; top:5px; right:10px; background:none; border:none; font-size:18px;">&times;</button>';

} else {

mensaje.style.backgroundColor = '#d4edda';

mensaje.style.color = '#155724';

mensaje.innerHTML = `✅ ${isUpdate ? 'Equipo actualizado correctamente.' : 'Equipo agregado exitosamente.'}` +

'<button onclick="this.parentElement.style.display=\'none\'" style="position:absolute; top:5px; right:10px; background:none; border:none; font-size:18px;">&times;</button>';

setTimeout(() => location.reload(), 1500);

}

});

});

</script>

<?php include '../includes/footer.php'; ?>

## 2.3 Historia de Usuario 2 — Reportes de Movimientos Sospechosos

Archivo: views/reportes.php

<!-- Reporte de Alertas -->

<section style="margin-top: 40px;">

<div style="display: flex; justify-content: space-between; align-items: center;">

<h3>Alertas Generadas</h3>

<button onclick="descargarAlertasPDF()">📥 Descargar PDF</button>

</div>

<table id="tabla-alertas">

<thead>

<tr>

<th>Mensaje</th>

<th>Nivel</th>

<th>Fecha</th>

<th>Laboratorio</th>

</tr>

</thead>

<tbody id="tabla-cuerpo-alertas">

<!-- Se llenará automáticamente por JavaScript -->

</tbody>

</table>

</section>

</div>

## 3.1 Historia de Usuario 3 — Generar Reportes de Inventario

Archivo: views/reportes.php

<!-- Reporte de Equipos -->

<section>

<div style="display: flex; justify-content: space-between; align-items: center;">

<h3>Equipos Registrados</h3>

<button onclick="descargarEquiposPDF()">📥 Descargar PDF</button>

</div>

<form id="filtro-form" onsubmit="event.preventDefault(); actualizarTablas();">

<label for="dia">Día:</label>

<input type="number" id="dia" name="dia" min="1" max="31" style="width: 70px;">

<label for="mes">Mes:</label>

<input type="number" id="mes" name="mes" min="1" max="12" style="width: 70px;">

<label for="anio">Año:</label>

<input type="number" id="anio" name="anio" min="2000" max="2100" style="width: 90px;">

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

</form>

<table id="tabla-equipos">

<thead>

<tr>

<th>Nombre</th>

<th>Fecha</th>

<th>Laboratorio</th>

</tr>

</thead>

<tbody id="tabla-cuerpo-equipos">

<!-- Se llenará automáticamente por JavaScript -->

</tbody>

</table>

</section>

## 3.2 Historia de Usuario 3 — Reportes de Alertas por Turno

Archivo: views/reportes.php

function descargarAlertasPDF() {

const doc = new jsPDF();

doc.text("Reporte de Alertas Generadas", 14, 15);

doc.autoTable({ html: '#tabla-alertas', startY: 20 });

doc.save('reporte\_alertas.pdf');

}

## 3.3 Historia de Usuario 3 — Reportes Automáticos de Inventario

Archivo: Archivo: views/reportes.php

function descargarEquiposPDF() {

const doc = new jsPDF();

doc.text("Reporte de Equipos Registrados", 14, 15);

doc.autoTable({ html: '#tabla-equipos', startY: 20 });

doc.save('reporte\_equipos.pdf');

}

# 5. Burndown Chart

| **Backlog ID** | **Historia de Usuario** | **Tiempo Estimado** | **23/09/2025** | **24/09/2025** | **25/09/2025** | **26/09/2025** | **27/09/2025** | **28/09/2025** | **29/09/2025** | **30/09/2025** | **01/10/2025** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dia1** | **Dia2** | **Dia3** | **Dia4** | **Dia5** | **Dia6** | **Dia7** | **Dia8** | **Dia9** |
| **HU-1.1** | Registrar Entradas y Salidas Automáticas | 3,5 | 3 |  |  |  |  |  |  |  |  |
| **HU-1.2** | Consultar Historial de Movimientos | 3 |  | 2 |  |  |  |  |  |  |  |
| **HU-1.3** | Registro de equipos | 5 |  |  | 5 |  |  |  |  |  |  |
| **HU-2.1** | Detectar Movimientos No Autorizados | 5 |  |  |  | 5 |  |  |  |  |  |
| **HU-2.2** | Validación de Equipos No Autorizados | 5 |  |  |  |  | 5 |  |  |  |  |
| **HU-2.3** | Reportes de Movimientos Sospechosos | 6 |  |  |  |  |  | 5 |  |  |  |
| **HU-3.1** | Generar Reportes de Inventario | 3 |  |  |  |  |  |  | 2 |  |  |
| **HU-3.2** | Reportes de Alertas por Turno | 4 |  |  |  |  |  |  |  | 4 |  |
| **HU-3.3** | Reportes Automáticos de Inventario | 3 |  |  |  |  |  |  |  |  | 3 |
| **Tiempo de trabajo** | | 37,5 | 3 | 2 | 5 | 5 | 5 | 5 | 2 | 4 | 3 |

# 6. Análisis del Sprint

* Se cumplió con todas las tareas estimadas.
* El progreso fue constante y sin bloqueos mayores.
* Las estimaciones fueron precisas y se logró un buen ritmo de trabajo.

# 7. Conclusiones

El Sprint 1 concluyó exitosamente con todas las funcionalidades básicas implementadas. Se generaron componentes reutilizables, se probaron los flujos de usuario, y el código quedó preparado para futuras funcionalidades como edición y múltiples listas.

El equipo demostró capacidad de entrega puntual y buena gestión de riesgos menores. Las métricas indican que la carga fue bien distribuida y los objetivos del sprint se cumplieron completamente.