

## **Prueba Técnica: Sistema de Gestión de Incidentes (Cloud Technologies)**

**Rol:** Programador Web Junior

**Fecha de Inicio:** 26 de Enero **Fecha Límite de Entrega:** Viernes 30 de Enero 12pm.

### **1. Contexto del Proyecto**

La empresa "Cloud Technologies" necesita desarrollar un módulo interno web para la gestión de incidentes técnicos. El objetivo es permitir que los empleados reporten fallas y que los administradores puedan gestionarlas.

### **2. Requerimientos Funcionales**

#### **A. Módulo de Autenticación y Roles :**

El sistema debe contar con un Login que diferencie dos tipos de usuarios:

##### **1. Rol Empleado:**

- Puede iniciar sesión.
- Puede **crear** un nuevo ticket de soporte.
- Puede **ver** la lista de tickets (solo lectura).

##### **2. Rol Administrador:**

- Tiene todos los permisos del Empleado.
- Puede **editar** el estado de los tickets (ej: cambiar de "Abierto" a "Resuelto").
- Puede **eliminar** tickets.

#### **B. Módulo de Tickets (CRUD) Cada ticket debe contener al menos:**

- Título del incidente.
- Descripción detallada.
- Prioridad (Baja, Media, Alta).
- Estado (Pendiente, En Proceso, Resuelto).

### C. Dashboard de Administrador :

Al entrar como Administrador, la pantalla principal debe mostrar un resumen visual del estado del sistema. Debe incluir al menos:

1. **Tarjetas de Conteo (KPIs):**
  - Total de tickets registrados.
  - Total de tickets "Abiertos" vs "Cerrados".
2. **Visualización:**
  - Se valorará positivamente (Bonus) si incluyes un gráfico simple (Barras o Pastel) mostrando la cantidad de tickets por "Prioridad" (ej: cuántos son de prioridad Alta).

### 3. Requerimientos Técnicos

- **Lenguaje:** Libre (puedes usar el stack con el que te sientas más cómodo: PHP, Python/Django/Flask, Java/Spring, Node.js, C#/.NET, etc.).
- **Base de Datos:** Libre (MySQL, PostgreSQL, SQL Server, MongoDB, etc.).
- **Frontend:** Libre (HTML/CSS puro, React, Vue, Angular, etc.).
- **Repositorio:** El código debe subirse a GitHub o GitLab.

### 4. Criterios de Evaluación (Lo que revisaremos)

1. **Funcionalidad:** Que el sistema cumpla con la distinción de roles (que un empleado no pueda borrar tickets).
2. **Calidad de Código:** Orden, nombres de variables claros y estructura del proyecto.
3. **Documentación (README):** Un archivo explicando cómo instalar y ejecutar tu proyecto en local.
4. **Bonus:** Validaciones de seguridad básicas
5. **Visual :** Siga normas de diseño ux/ui , microinteracciones.