

Prueba Técnica - Desarrollador Full Stack (Node.js, Angular, SQL Server)

Objetivo

Evaluar las habilidades del candidato en el desarrollo de aplicaciones full stack, incluyendo backend en Node.js (Sails.js), frontend en Angular e integración con SQL Server.

Secciones de la Prueba

1. Diseño de Base de Datos (SQL Server)
2. Desarrollo Backend (Node.js / Sails.js)
3. Desarrollo Frontend (Angular)

Diseño de Base de Datos (SQL Server)

Tarea:

Diseñar un esquema de base de datos para una aplicación de gestión de pedidos con los siguientes requerimientos:

- Usuarios pueden registrarse, iniciar sesión y administrar su información.
- Productos con nombre, precio y stock.
- Pedidos que pueden contener múltiples productos.
- Cada pedido tiene un estado (Pendiente, Enviado, Entregado).
- Cada usuario puede realizar múltiples pedidos.

Criterios de Evaluación

- ✓ Uso correcto de Primary Keys (PK) y Foreign Keys (FK)
- ✓ Normalización adecuada (3FN mínima)
- ✓ Uso correcto de tipos de datos y restricciones
- ✓ Creación de Stored Procedures para insertar un pedido

Desarrollo Backend (Node.js / Sails.js)

Tarea:

Implementar una API REST usando Sails.js para manejar pedidos con los siguientes endpoints:

- POST /api/register → Crear un usuario
- POST /api/login → Autenticación de usuario (JWT)
- GET /api/products → Listar productos
- POST /api/orders → Crear un pedido
- GET /api/orders/:id → Consultar detalles de un pedido

Requisitos:

- Usar JWT para autenticación.
- Conectar la API con SQL Server usando Sequelize u otro ORM.
- Manejo de middleware para proteger rutas privadas.

Criterios de Evaluación

- ✓ Uso de Sails.js y arquitectura MVC
- ✓ Correcta implementación de JWT
- ✓ Conexión eficiente a SQL Server
- ✓ Código estructurado y mantenible

Desarrollo Frontend (Angular)

Tarea:

- Desarrollar una SPA con Angular que permita a un usuario:
- Iniciar sesión con usuario y contraseña.
- Ver la lista de productos disponibles.
- Crear un pedido seleccionando productos.
- Consultar sus pedidos realizados.

Requisitos:

- Usar Angular Material o Bootstrap.
- Manejo de servicios HTTP para consumir la API.
- Implementar guards para proteger rutas privadas.

Criterios de Evaluación:

- ✓ Correcta estructuración del proyecto Angular
- ✓ Uso de servicios y guards
- ✓ Consumo eficiente de API
- ✓ Diseño responsive

Entrega de la Prueba

- Repositorio en GitHub o GitLab con código fuente.
- Archivo README.md con instrucciones para ejecutar la app.
- Scripts de SQL Server para crear la base de datos.

Criterios de Evaluación Globales

- ✓ Código limpio y estructurado
- ✓ Uso eficiente de Git (commits bien documentados)
- ✓ Seguridad en autenticación y base de datos
- ✓ Implementación de mejores prácticas