

Desafío de Desarrollo Web Full Stack

Objetivo:

Construir un sistema para la gestión de productos de un negocio, utilizando Angular/React para la interfaz de usuario, C# para el backend API y SQL Server para la base de datos.

Requisitos:

Backend API:

- Crear un proyecto de API RESTful utilizando C# y .NET Core.
- Implementar endpoints para:
 - Obtener todos los productos.
 - Obtener un producto por su ID.
 - Agregar un nuevo producto.
 - Actualizar un producto existente.
 - Eliminar un producto por su ID.
- Implementar paginación y búsqueda en la lista de productos.
- Implementar autenticación y autorización utilizando JWT.
- Crear endpoints para la gestión de usuarios (registro, login, obtener perfil).

Base de Datos (SQL Server):

- Diseñar una base de datos en SQL Server para almacenar la información de los productos y usuarios.
 - Tabla de productos: ID, nombre, descripción, precio, stock, fecha de creación, fecha de última actualización.
 - Tabla de usuarios: ID, nombre de usuario, contraseña hasheada, email, rol (admin/usuario).

Frontend:

- Crear una interfaz de usuario para gestionar los productos y usuarios.
- Implementar las siguientes funcionalidades:

- Mostrar una lista de todos los productos con paginación y búsqueda.
- Permitir agregar un nuevo producto con nombre, descripción, precio y stock.
- Permitir editar un producto existente.
- Permitir eliminar un producto.
- Mostrar mensajes de éxito/error al realizar operaciones CRUD.
- Implementar autenticación y autorización en el frontend.
- Permitir el registro y login de usuarios.
- Mostrar y permitir la gestión de usuarios solo a administradores.

Recomendaciones:

- Utilizar TypeScript tanto en el frontend.
- Utilizar EntityFramework para el manejo de los objetos de la base de datos.
- Implementar validaciones de datos tanto en el frontend como en el backend.
- Utilizar servicios para la comunicación entre el frontend y el backend.
- Aplicar buenas prácticas de diseño y programación.

Entregables:

- Código fuente del backend API en C#/.NET Core.
- Código fuente del frontend en Angular/React.
- Script de creación de la base de datos en SQL Server.
- Incluir README detallado del proyecto, incluyendo instrucciones de instalación, configuración y uso.
- Hacer uso de Docker para contenerizar la aplicación (backend, frontend y base de datos).
- Incluir documentación de la API utilizando Swagger.
- Proveer un conjunto de datos de prueba.
- Subir a repositorio de Github

Tiempo de entrega: 3 días.