

Trabajo Práctico: "Agenda de Citas"

Objetivo

Crear una API RESTful para gestionar una agenda de citas utilizando **TypeORM** con una base de datos **MySQL**. La API debe permitir las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para los recursos necesarios.

Requerimientos

1. **Crear una base de datos llamada agenda_citas.**
2. **Definir dos tablas:**
 - clientes (almacena la información de los clientes)
 - citas (almacena la información de las citas programadas para los clientes)

Modelo de Base de Datos

- **Tabla clientes:**
 - id: entero, clave primaria, auto-incremental
 - nombre: varchar(100), no nulo
 - apellido: varchar(100), no nulo
 - email: varchar(150), no nulo, único
 - telefono: varchar(20), opcional
- **Tabla citas:**
 - id: entero, clave primaria, auto-incremental
 - fecha: datetime, no nulo
 - motivo: varchar(255), no nulo
 - clienteId: entero, clave foránea que referencia a la tabla clientes

Requisitos Funcionales

1. **Crear clientes:**
 - Crear un endpoint para añadir nuevos clientes.
2. **Listar clientes:**
 - Crear un endpoint para listar todos los clientes.
3. **Crear citas:**
 - Crear un endpoint para añadir nuevas citas para los clientes existentes.
4. **Listar citas:**
 - Crear un endpoint para listar todas las citas programadas.
 - Crear un endpoint para listar las citas de un cliente específico.
5. **Actualizar citas:**
 - Crear un endpoint para actualizar la fecha o el motivo de una cita existente.
6. **Eliminar citas:**
 - Crear un endpoint para eliminar una cita.

Validaciones

- Verificar que los datos del cliente (nombre, apellido, email) sean válidos antes de crearlo.
- Verificar que no se cree una cita en el pasado.
- No permitir la creación de citas duplicadas (mismo cliente y misma fecha).

Desafíos Adicionales

- **Implementar Paginación:** para los listados de clientes y citas.
- **Filtrado por Fecha:** permitir filtrar citas por un rango de fechas.
- **Autenticación:** agregar autenticación básica para proteger los endpoints de creación, actualización y eliminación de citas.

Consideraciones

- **Uso de TypeORM:** Definir las entidades correspondientes en TypeORM y manejar la conexión con MySQL.
- **Manejo de Errores:** Asegurarse de que la API maneje correctamente los errores, como clientes inexistentes o citas en fechas inválidas.