

Opción A. Stack ASP.NET MVC + SQL SERVER:

3.1.1 Deberá trabajar sobre visual studio y Sql Server para el ejercicio.

3.1.2 Desarrolle una aplicación WEB en C# y SQL Server, deberá seguir el patrón MVC (ASP.NET MVC).

3.1.3 Implementar Token con JWT Authentication para ASP.NET.

3.1.4 Deberá utilizar por lo menos alguna funcionalidad con javascript en alguna de las pantallas del CRUD.

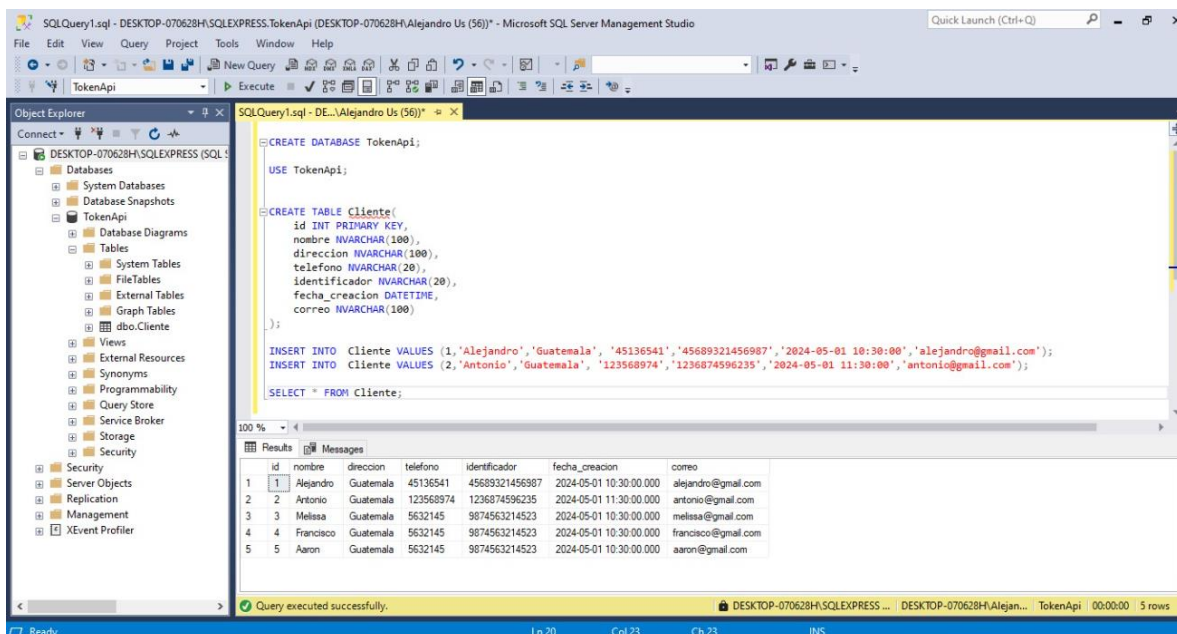
3.1.5 Implementar por lo menos alguna versión de bootstrap, material design o cualquier framework de css.

3.1.6 Crear un repositorio git público, con el proveedor que más se le facilite (GitHub, GitLab, BitBucket, etc). y subir el proyecto a realizar junto con los screenshots y el script de la base de datos.

3.2 Requisitos funcionales:

3.2.1 Crear una base de datos y en ella una tabla que permita almacenar datos de contacto de clientes con por lo menos los siguientes campos:

* Nombre de Cliente, Dirección, Telefono, Identificador (Nit o DPI), Fecha de creación, Correo Electrónico.



```
CREATE DATABASE TokenApi;

USE TokenApi;

CREATE TABLE Cliente(
    id INT PRIMARY KEY,
    nombre NVARCHAR(100),
    direccion NVARCHAR(100),
    telefono NVARCHAR(20),
    identificador NVARCHAR(20),
    fecha_creacion DATETIME,
    correo NVARCHAR(100)
);

INSERT INTO Cliente VALUES (1, 'Alejandro', 'Guatemala', '45136541', '45689321456987', '2024-05-01 10:30:00', 'alejandrogmail.com');
INSERT INTO Cliente VALUES (2, 'Antonio', 'Guatemala', '123568974', '1236874596235', '2024-05-01 11:30:00', 'antonio@gmail.com');

SELECT * FROM Cliente;
```

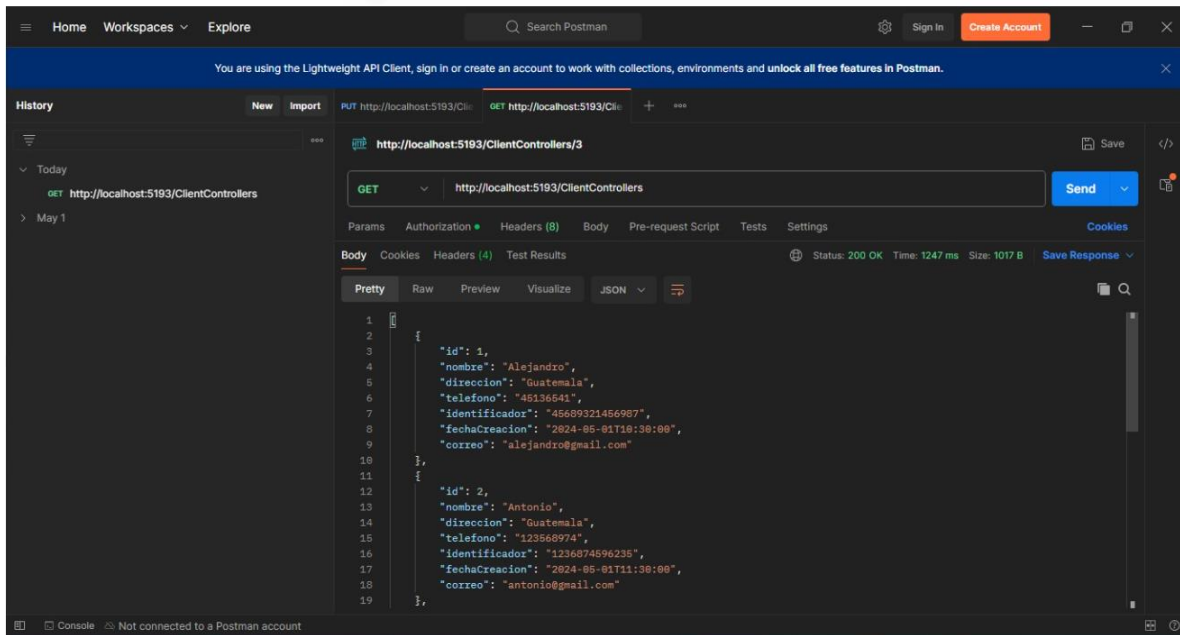
	id	nombre	direccion	telefono	identificador	fecha_creacion	correo
1	1	Alejandro	Guatemala	45136541	45689321456987	2024-05-01 10:30:00.000	alejandrogmail.com
2	2	Antonio	Guatemala	123568974	1236874596235	2024-05-01 11:30:00.000	antonio@gmail.com
3	3	Melissa	Guatemala	5632145	9874563214523	2024-05-01 10:30:00.000	melissa@gmail.com
4	4	Francisco	Guatemala	5632145	9874563214523	2024-05-01 10:30:00.000	francisco@gmail.com
5	5	Aaron	Guatemala	5632145	9874563214523	2024-05-01 10:30:00.000	aaron@gmail.com

Query executed successfully.

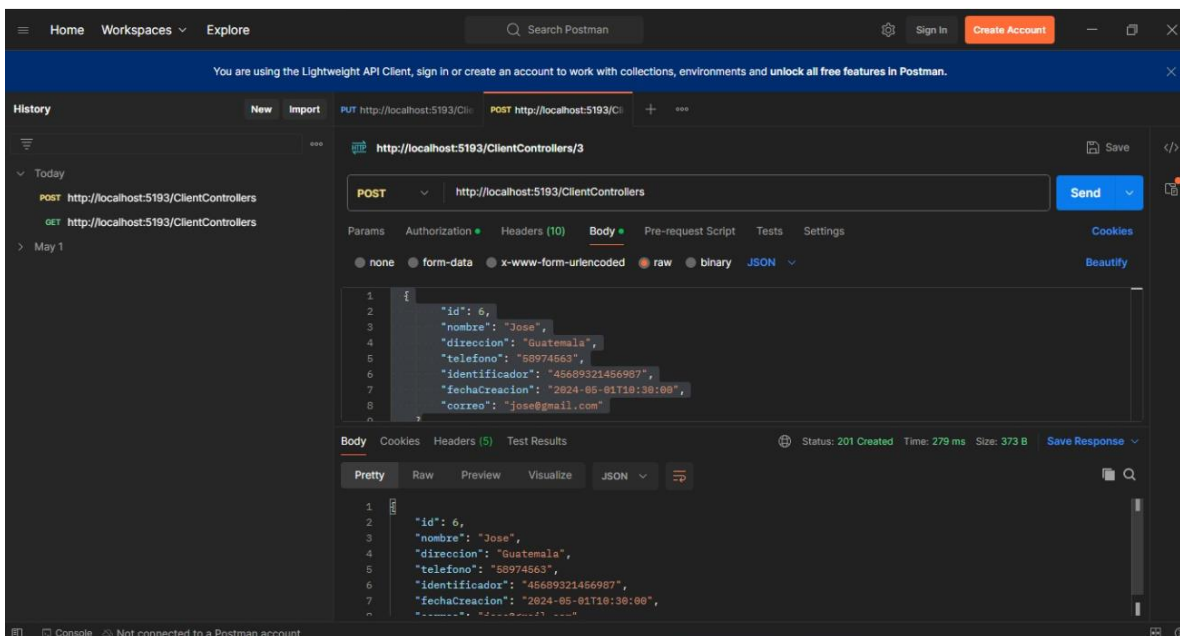
3.2.2 Desarrolle el backend y fronted para que la aplicación pueda realizar el CRUD de la tabla [clientes] de la base de datos del inciso anterior.

NOTA: Debido a restricciones de tiempo, no pude completar la implementación del frontend en el proyecto. Sin embargo, todos los métodos requeridos fueron desarrollados y probados utilizando POSTMAN.

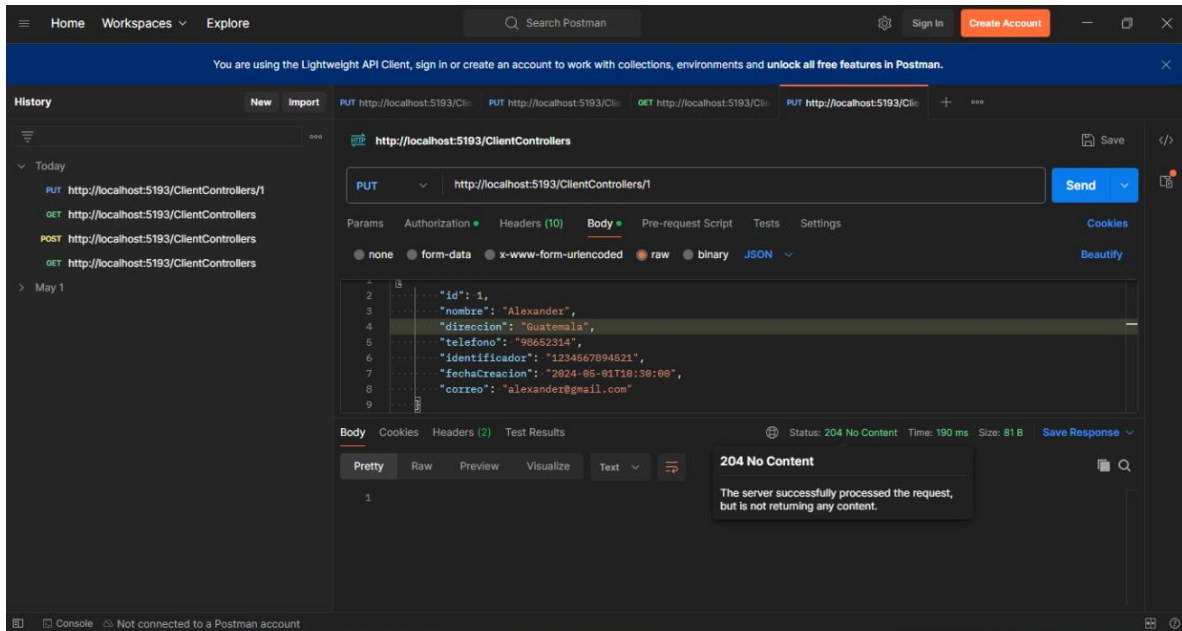
GET:



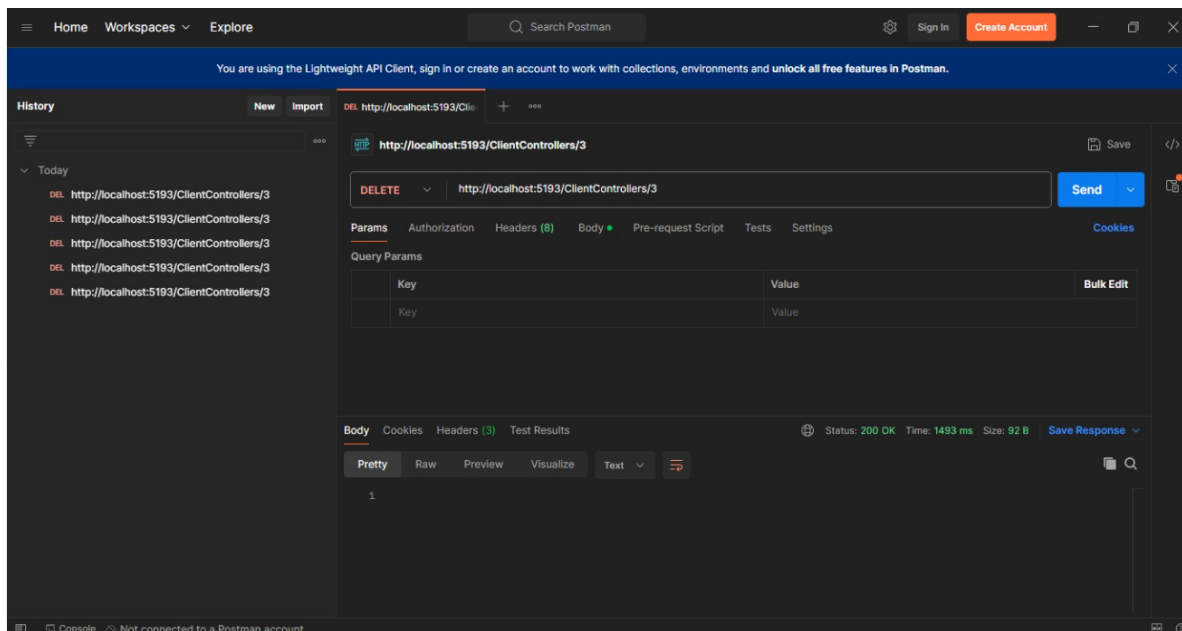
POST



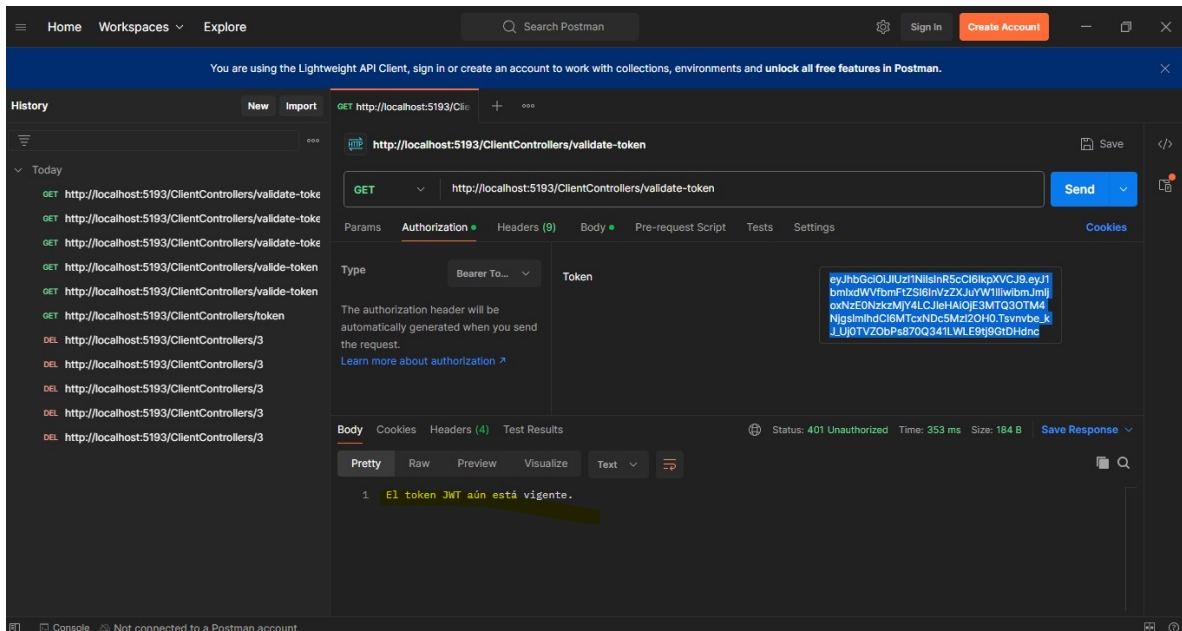
PUT



DELETE



3.2.4 Crear otro metodo get o post que valide el token generado en el punto anterior e indique si aun esta vigente.



3.2.5 Cree un método get que reciba como parámetro el ID de un cliente y devuelva en un JSON los datos del mismo.

