Krugger Challengue

Contenido

[Back End 1](#_Toc186803428)

[FrontEND 2](#_Toc186803429)

[Faltantes: 3](#_Toc186803430)

# Back End

Instalo dependencias:

Npgsql.EntityFrameworkCore.PostgreSQL recomendado 8.0.8

Microsoft Entity FrameworkCore Tools

Microsoft.EntityFrameworkCore.Design

Microsoft ApsNetCore Authentication JwtBearer

Cree la conexión a la BDD: appsetting.json

"ConnectionStrings": {

"CadenaSQL": "Host=localhost;Port=5432;Username=postgres;Password=admin;Database=SisPaseCortesia;"

},

"JWT": {

"key": "7a6d81cd-3c1c-4660-ba71-68e585a00400"

}

Cree la clase KruggerDbContext

Creo la clase para Entidad

Tabla Cliente

Creo el modelo para CrearActualizar

En Dbcontext pongo la función

public DbSet<Cliente> Cliente { get; set; }

Establezco la conexion a la BDD

var connectionString = builder.Configuration.GetConnectionString("CadenaSQL");

builder.Services.AddDbContext<KruggerDbContext>(op =>

{

op.UseNpgsql(connectionString);

});

Habilito el CORS

app.UseCors("AllowAll");

Creo y ejecuto las migraciones

dotnet --version

dotnet ef migrations add Inicial

dotnet ef database update

\*\*\*\*\*\*\*\*\*\*Se ha creado con éxito la tabla de datos estar pilas con el DBcontext

Voy a crear una clase genérica para manejar respuestas personalizadas

Response.cs

Voy a crear un controlador

Inyecto clase dbcontext usando el constructor principal

Creo el EndPoint del Listado

Creo el EndPoint para crear Registros

Creo el objeto de tipo cliente para enviar al dbcontext

Creo el EndPoint para Actualizar

Se crea un método privado para obtener id reutilizando consulta

Creo el EndPoint para Eliminar

Revisar el actualizar porque sin existir me actualiza

# FrontEND

ng new AppKruggerChallengue --routing --skiptests --style=css

ng dev

Empiezo a instalar complementos

ng add @angular/material

npm I ngx-toastr

voy a crear las carpetas del proyecto

Crear el servicio con el cli

ng g s services/clients

Voy a proveer y configurar toast

Crear componentes reutilizables y paginas de rutas

ng g c components/navbar

ng g c components/spinner

ng g c components/cliente-card

Creo las paginas

ng g c pages/home

admin

configure las rutas

Vamos a listar los registros

Creo el eliminar

Creo el ver detalle

Creo el actualizar

SECTORES

He configurado el Api para los servicios de sectores

Listar Sectores listo

Crear, editar y eliminar realizado

Login para Clientes Realizado con autenticación JWT

### Faltantes:

* Falta que al ingresar el usuario muestre sus datos y su sector (revisar método getClient By id)
* Falta implementar mapa en cliente domicilio y en sectores Y para el Login del Cliente donde lea longitud
* La Api del mapa Google es pagada
* Utilizar Leaflet pero no es compatible para mi versión. (opcional bajar de versión?)