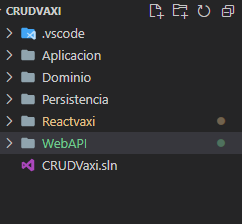
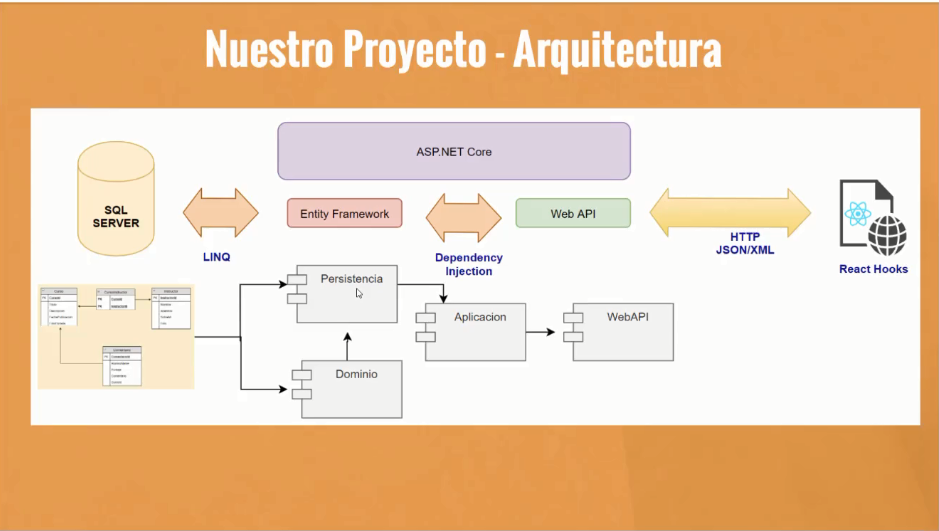
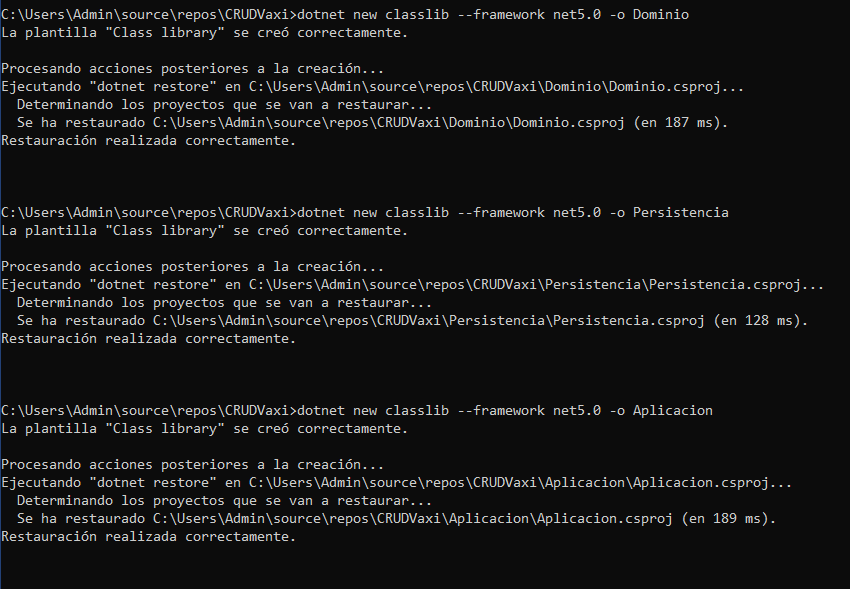
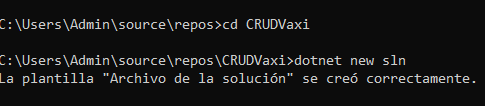
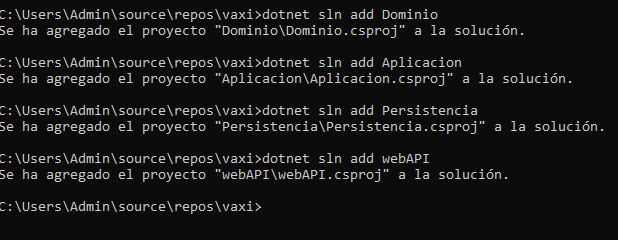
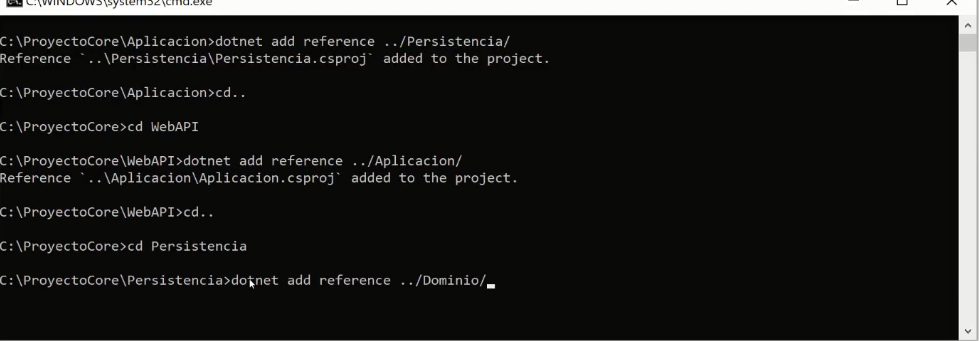
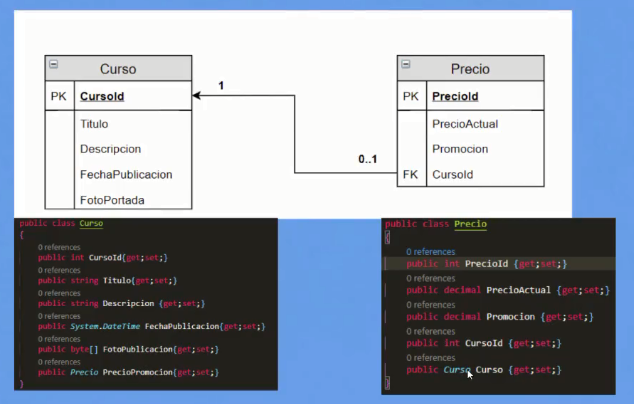
[Tutorial: Crear una API web con ASP.NET Core | Aprendizaje de Microsoft](https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-6.0&tabs=visual-studio-code)[Referencia de herramientas de EF Core (.NET CLI) - EF Core | Aprendizaje de Microsoft](https://learn.microsoft.com/en-us/ef/core/cli/dotnet) 

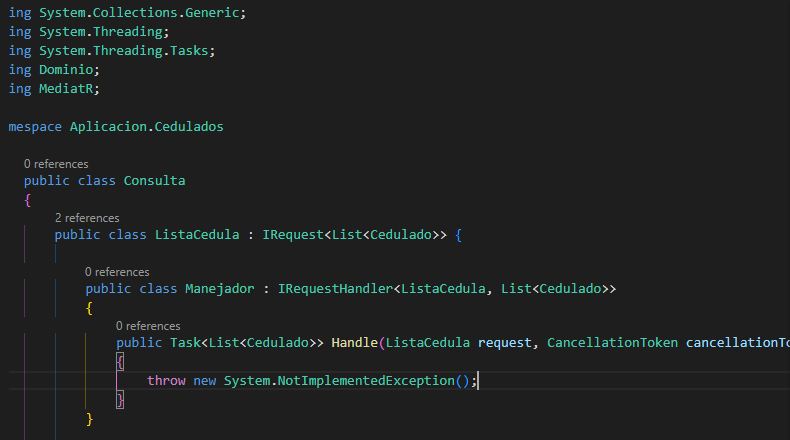


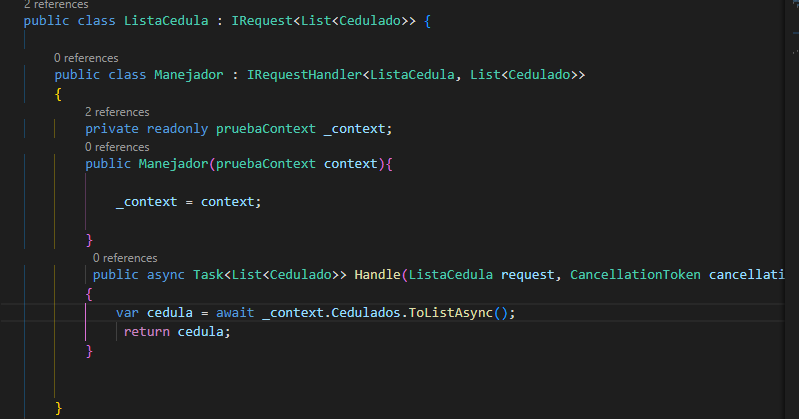




dotnet ef dbcontext scaffold "Server=DESKTOP-VJH4EMM\SQLEXPRESS;Database=prueba;Trusted\_Connection=True;" Microsoft.EntityFrameworkCore.SqlServer -o Models

Consulta



using System.Collections.Generic;

using System.Threading;

using System.Threading.Tasks;

using Dominio;

using MediatR;

using  Microsoft.EntityFrameworkCore;

using Persistencia;

namespace Aplicacion.Cedulados

{

    public class Consulta

    {

        public class ListaCedula : IRequest<List<Cedulado>> {

            public class Manejador : IRequestHandler<ListaCedula, List<Cedulado>>

            {

                private readonly pruebaContext \_context;

                public Manejador(pruebaContext context)

                {

                    \_context = context;

                }

                 public async Task<List<Cedulado>> Handle(ListaCedula request, CancellationToken cancellationToken)

                {

                    var cedula = await \_context.Cedulados.ToListAsync();

                     return cedula;

                }

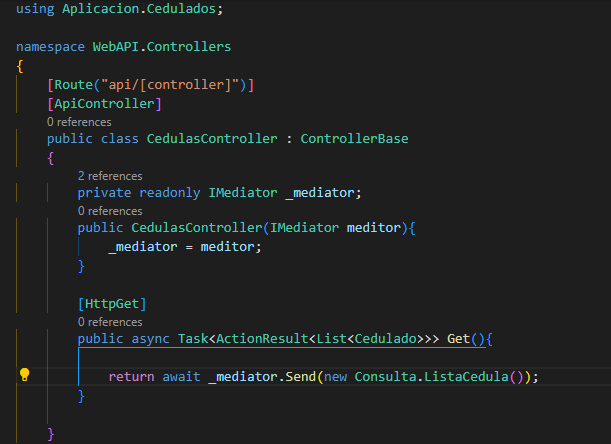
            }

        }

    }

}

Consulta controller webapi



using MediatR;

using System.Threading.Tasks;

using System.Collections.Generic;

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using Microsoft.EntityFrameworkCore;

using Dominio;

using Aplicacion.Cedulados;

namespace WebAPI.Controllers

{

    [Route("api/[controller]")]

    [ApiController]

    public class CedulasController : ControllerBase

    {

        private readonly IMediator \_mediator;

        public CedulasController(IMediator meditor){

            \_mediator = meditor;

        }

        [HttpGet]

        public async Task<ActionResult<List<Cedulado>>> Get(){

            return await \_mediator.Send(new Consulta.ListaCedula());

        }

    [HttpGet("{id}")]

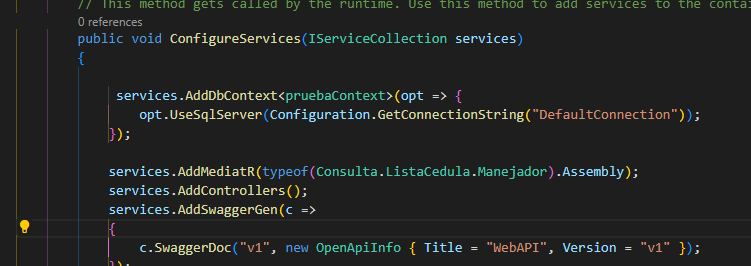
    public async Task<ActionResult<Cedulado>> Detalle(int id){

        return await \_mediator.Send(new ConsultaId.CedulaUnica{Id = id});

    }

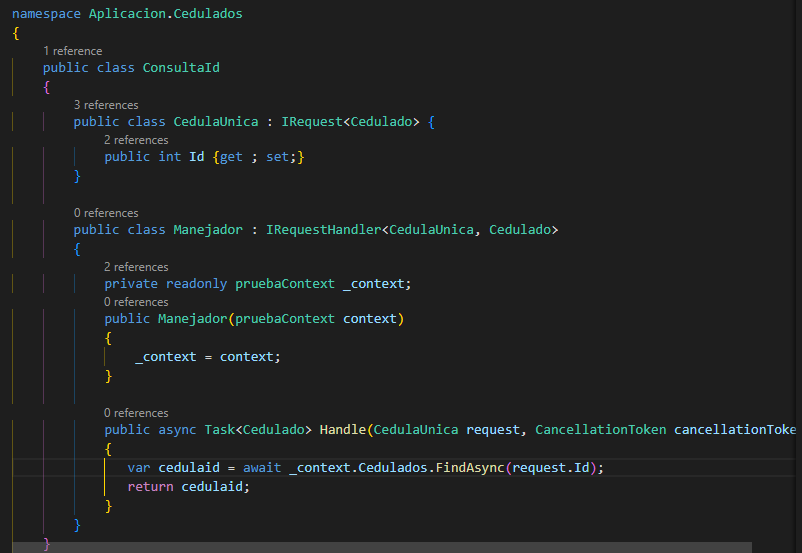
    }

}

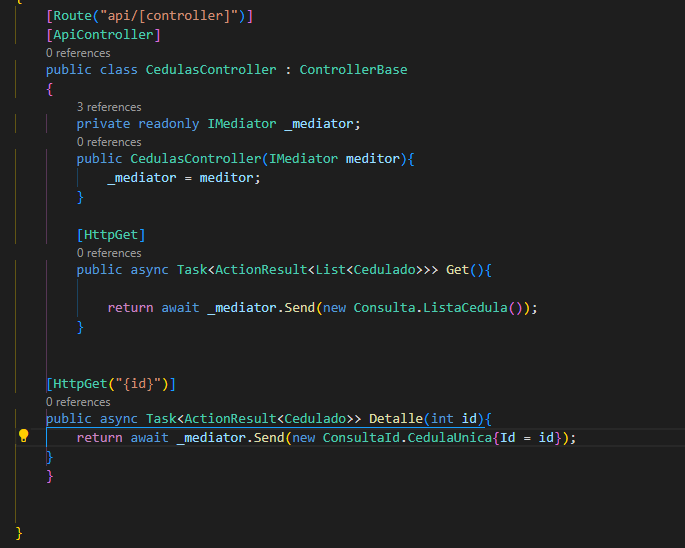
Configuracion de startup

CONSULTA POR ID

CONSULTAID

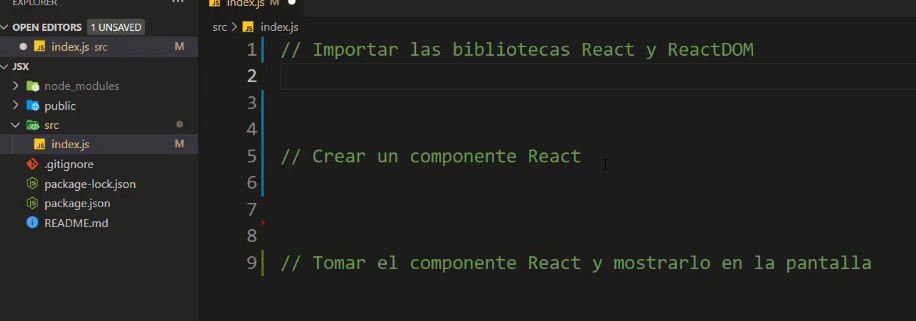


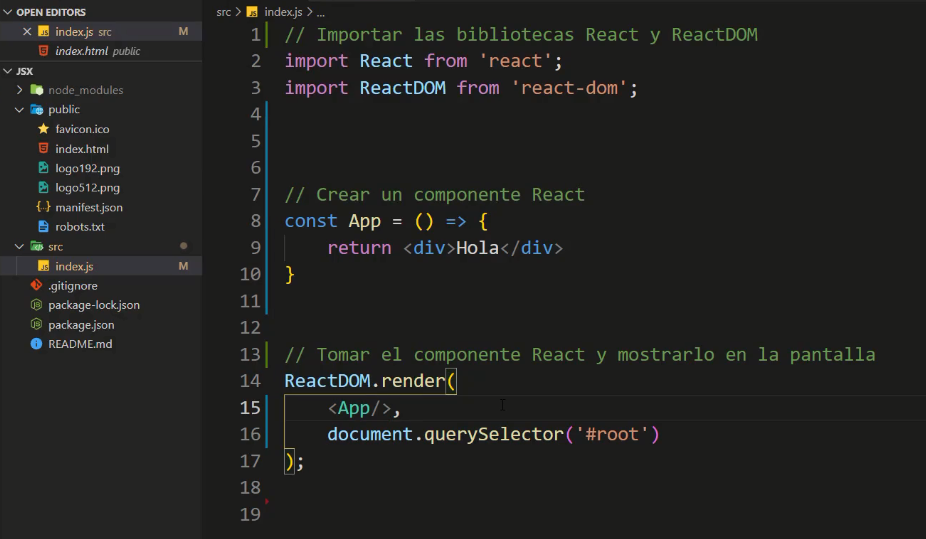
CONSULTA ID CONTROLLER

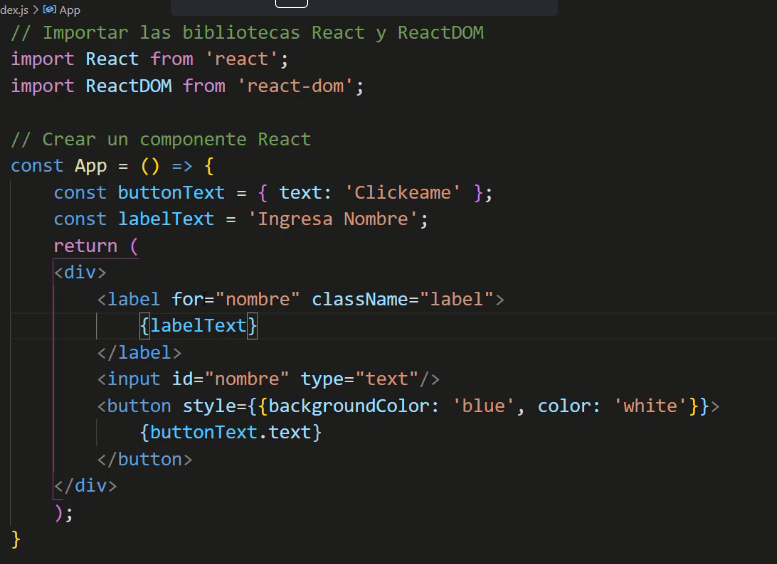
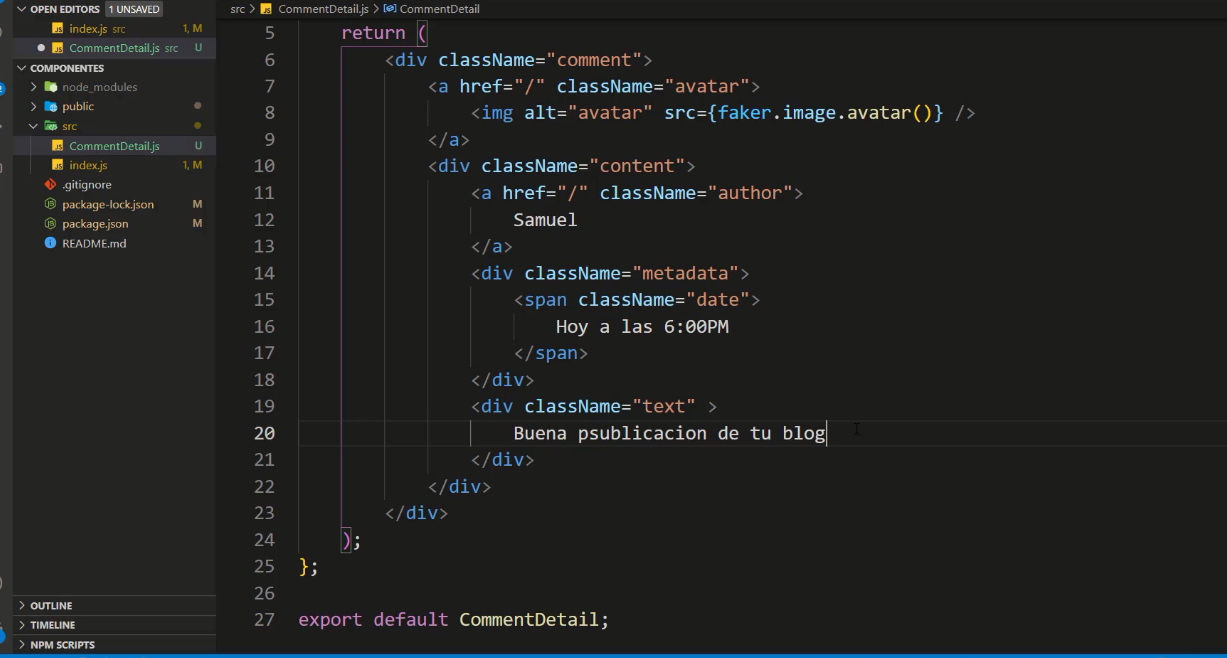
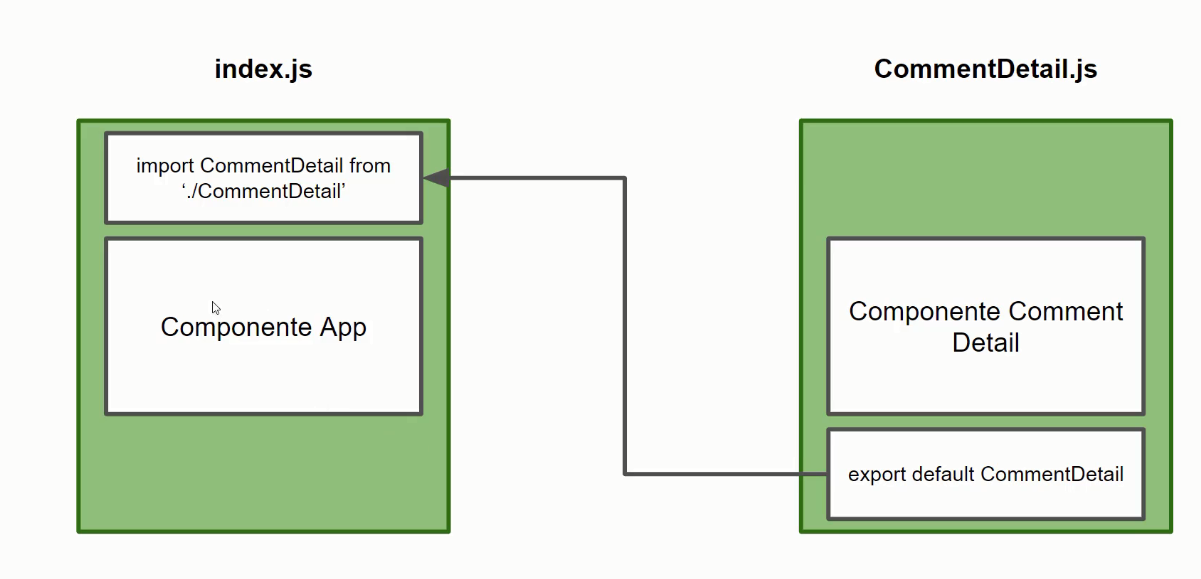
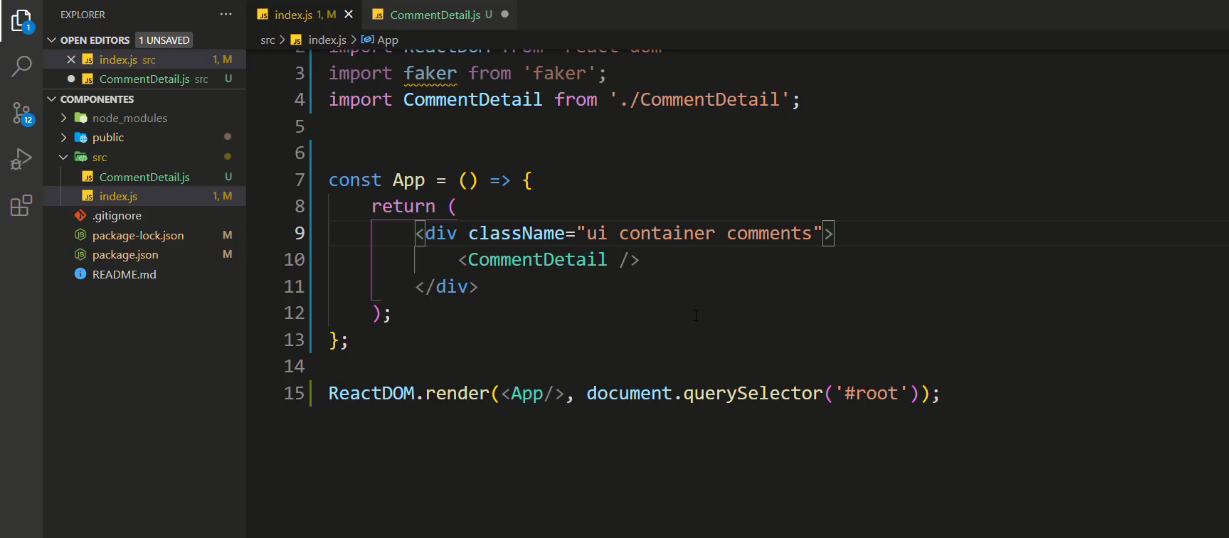


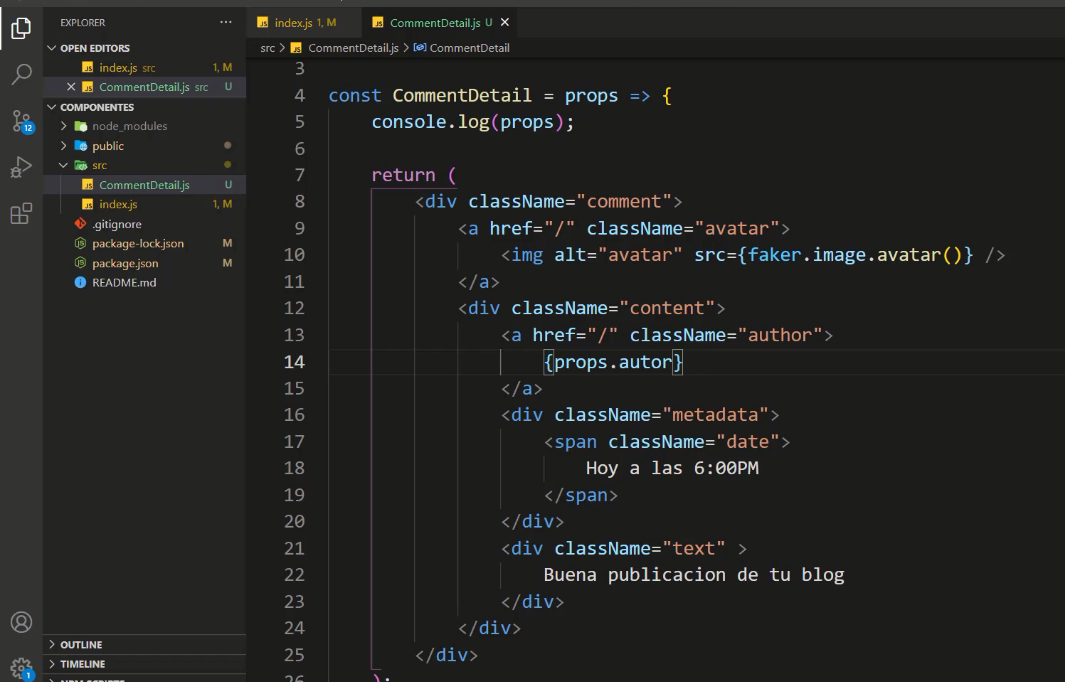
REACT

npx create-react-app my-app





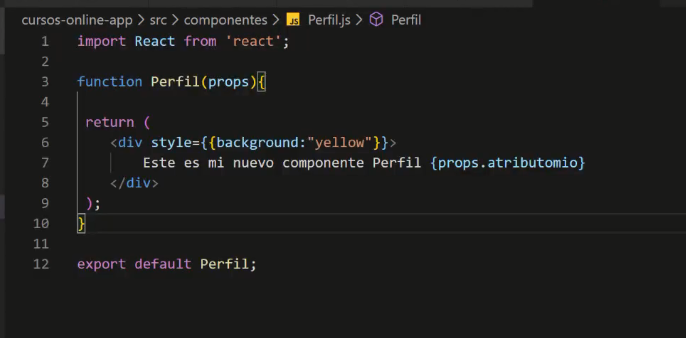
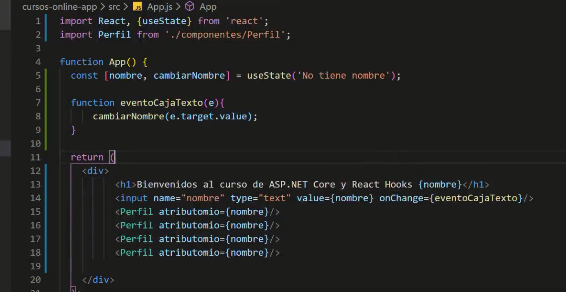
+ 

npm install axios

npm i cors

npm install @mui/material @emotion/react @emotion/styled

npm install @ material - ui / styles



import React, { useEffect, useState } from "react";

import 'bootstrap/dist/css/bootstrap.min.css'

import axios from 'axios';

import {Modal,ModalBody,ModalFooter,ModalHeader} from "reactstrap";

function FrontendT(){

     const baseUrl = "https://localhost:5001/api/Cedula";

     const [data, setdata]=useState([]);

     const [modalInsertar, setModalInsertar]= useState(false);

     const [modalEditar, setModalEditar]= useState(false);

     const [modalEliminar, setModalEliminar]= useState(false);

     const [gestorSeleccionado,setGestorSeleccionado ]=useState({

        numeroCedula: '',

        nombre: '',

        apellidos: '',

        fechaNacimineto: '',

        nacionalidad: '',

        fkCedula: '',

     })

     const handleChange=e=>{

        const {name, value}=e.target;

        setGestorSeleccionado({

            ...gestorSeleccionado,

            [name]: value

        });

        console.log(gestorSeleccionado);

    }

    const abrirCerarModalInsertar=()=>{

        setModalInsertar(!modalInsertar);

    }

    const abrirCerarModalEliminar=()=>{

        setModalEliminar(!modalEliminar);

    }

    const abrirCerarModalEditar=()=>{

        setModalEditar(!modalEditar);

    }

     const peticionGet=async()=>{

        await axios.get(baseUrl)

        .then(response =>{

            setdata(response.data);

        }).catch(error =>{

            console.log(error);

        })

     }

     const peticionPost=async()=>{

       // delete gestorSeleccionado.id;

       //gestorSeleccionado.fkCedula=parseInt(gestorSeleccionado.fkCedula)

        await axios.post(baseUrl,gestorSeleccionado)

       .then(response =>{

           setdata(data.concat(response.data));

          abrirCerarModalInsertar();

        }).catch(error =>{

            console.log(error);

        })

     }

     const peticionPut=async()=>{

        gestorSeleccionado.numeroCedula=parseInt(gestorSeleccionado.numeroCedula);

         await axios.put(baseUrl+"/"+gestorSeleccionado.numeroCedula,gestorSeleccionado)

        .then(response =>{

            var respuesta=response.data;

            var dataAuxiliar=data;

           dataAuxiliar.map(gestor =>{

            if(gestor.numeroCedula===gestorSeleccionado.numeroCedula){

                gestor.numeroCedula=respuesta.numeroCedula;

                gestor.nombre=respuesta.nombre;

                gestor.apellidos=respuesta.apellidos;

                gestor.fechaNacimineto=respuesta.fechaNacimineto;

                gestor.nacionalidad=respuesta.nacionalidad;

                gestor.fkCedula=respuesta.fkCedula;

                }

           });

           abrirCerarModalEditar();

           peticionGet();

         }).catch(error =>{

             console.log(error);

         })

      }

      const peticionDelete=async()=>{

         await axios.delete(baseUrl+"/"+gestorSeleccionado.numeroCedula,gestorSeleccionado)

        .then(response =>{

           setdata(data.filter(gestor=>gestor.numeroCedula!==response.data));

           abrirCerarModalEliminar();

           peticionGet();

         }).catch(error =>{

             console.log(error);

         })

      }

     const seleccionarGestor=(gestor, caso)=>{

        setGestorSeleccionado(gestor);

        (caso==="Editar")?

         abrirCerarModalEditar():

         abrirCerarModalEliminar();

     }

     useEffect(()=>{

        peticionGet();

     },[])

     return(

       <div>

                <div className="App">

                 <br/>

                 <button onClick={()=>abrirCerarModalInsertar()}  className="btn btn-success">Insertar nuevo Registro</button>

                        <table className="table table-bordered">

                            <thead>

                                <tr>

                                <th >Cedula</th>

                                <th >Nombre</th>

                                <th >Apellido</th>

                                <th >Fecha de nacimiento</th>

                                <th >Nacionalidad</th>

                                <th>DEMAS</th>

                                <th>Acciones</th>

                                </tr>

                            </thead>

                            <tbody>

                                {data.map((gestor,indice) => (

                                <tr key={indice}>

                                <td>{gestor.numeroCedula}</td>

                                <td>{gestor.nombre}</td>

                                <td>{gestor.apellidos}</td>

                                <td>{gestor.fechaNacimineto}</td>

                                <td>{gestor.nacionalidad}</td>

                                <td>{gestor.fkCedula}</td>

                                <td>

                                    <button className="btn btn-primary" onClick={()=>seleccionarGestor(gestor,"Editar")}>Editar</button>

                                    <button className="btn btn-danger" onClick={()=>seleccionarGestor(gestor,"Eliminar")}>Eliminar</button>

                                </td>

                                </tr>

                            ))}

                            </tbody>

                            </table>

                            </div>

                            <Modal isOpen={modalInsertar}>

                                <ModalHeader>Insertar datos</ModalHeader>

                                <ModalBody>

                                    <div>

                                        <label>Cedula: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="numeroCedula" onChange={handleChange}/>

                                        <label>Nombre: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="nombre" onChange={handleChange}/>

                                        <label>Apellido: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="apellidos" onChange={handleChange}/>

                                        <label>Fecha de Naciomiento: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="fechaNacimineto" onChange={handleChange}/>

                                        <label>Nacionalidad: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="nacionalidad" onChange={handleChange}/>

                                        <label>De mas: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="fkCedula" onChange={handleChange}/>

                                        </div>

                                </ModalBody>

                                <ModalFooter>

                                <button className="btn btn-primary" onClick={()=>peticionPost()}>Insertar</button>

                                    <button className="btn btn-danger" onClick={()=>abrirCerarModalInsertar()}>Cancelar</button>

                                </ModalFooter>

                            </Modal>

                            <Modal isOpen={modalEditar}>

                                <ModalHeader>Editar datos</ModalHeader>

                                <ModalBody>

                                    <div>

                                        <label>Cedula: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="numeroCedula" readOnly value={gestorSeleccionado && gestorSeleccionado.numeroCedula}/>

                                        <label>Nombre: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="nombre" onChange={handleChange}value={gestorSeleccionado && gestorSeleccionado.nombre}/>

                                        <label>Apellido: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="apellidos" onChange={handleChange} value={gestorSeleccionado && gestorSeleccionado.apellidos}/>

                                        <label>Fecha de Naciomiento: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="fechaNacimineto" onChange={handleChange} value={gestorSeleccionado && gestorSeleccionado.fechaNacimineto}/>

                                        <label>Nacionalidad: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="nacionalidad" onChange={handleChange} value={gestorSeleccionado && gestorSeleccionado.nacionalidad}/>

                                        <label>De mas: </label>

                                        <br/>

                                        <input type="text" className="form-control" name="fkCedula" onChange={handleChange} value={gestorSeleccionado && gestorSeleccionado.fkCedula}/>

                                    </div>

                                </ModalBody>

                                <ModalFooter>

                                <button className="btn btn-primary" onClick={()=>peticionPut()}>Editar</button>

                                    <button className="btn btn-danger" onClick={()=>abrirCerarModalEditar()}>Cancelar</button>

                                </ModalFooter>

                            </Modal>

                            <Modal isOpen={modalEliminar}>

                                <ModalHeader>Editar datos</ModalHeader>

                                <ModalBody>

                                    <div>

                                        <h3>esta seguro que quiere eliminar el Registro</h3>

                                        <h4>{gestorSeleccionado && gestorSeleccionado.nombre}</h4>

                                   </div>

                                </ModalBody>

                                <ModalFooter>

                                <button className="btn btn-danger" onClick={()=>peticionDelete()}>si</button>

                                    <button className="btn btn-segundary" onClick={()=>abrirCerarModalEliminar()}>no</button>

                                </ModalFooter>

                            </Modal>

        </div>

     )

}

export default FrontendT;