

Universidad Autónoma De Sinaloa

Facultad De Informática Culiacán

Nombre: Javier Esaú Zaragoza Cervantes

Grupo: 2-3

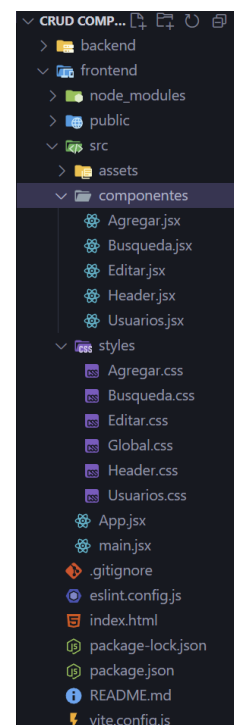
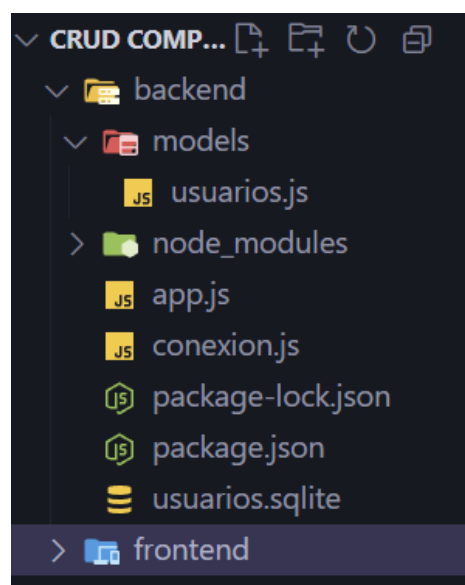
Materia: Desarrollo web del lado del servidor

Actividad: Crear una aplicación web considerando un CRUD completo (operaciones de crear, leer, actualizar y eliminar).

Este proyecto incluye todo el código y los recursos utilizados para desarrollar y operar el backend del CRUD de usuarios. Siguiendo las especificaciones, ha sido implementado con Node.js, Express y Sequelize, utilizando DB Browser for SQLite como base de datos.

El backend expone una API REST, permitiendo gestionar registros de usuarios mediante operaciones CRUD: Crear, Leer, Actualizar y Eliminar.

Diseñado para ser escalable y eficiente, este sistema proporciona una conexión estable entre el frontend y la lógica de persistencia de datos, asegurando una comunicación fluida entre la interfaz de usuario y la base de datos.



Es un backend en Node.js para manejar usuarios con CRUD y SQLite.

models/usuarios.js: define el modelo de usuario.

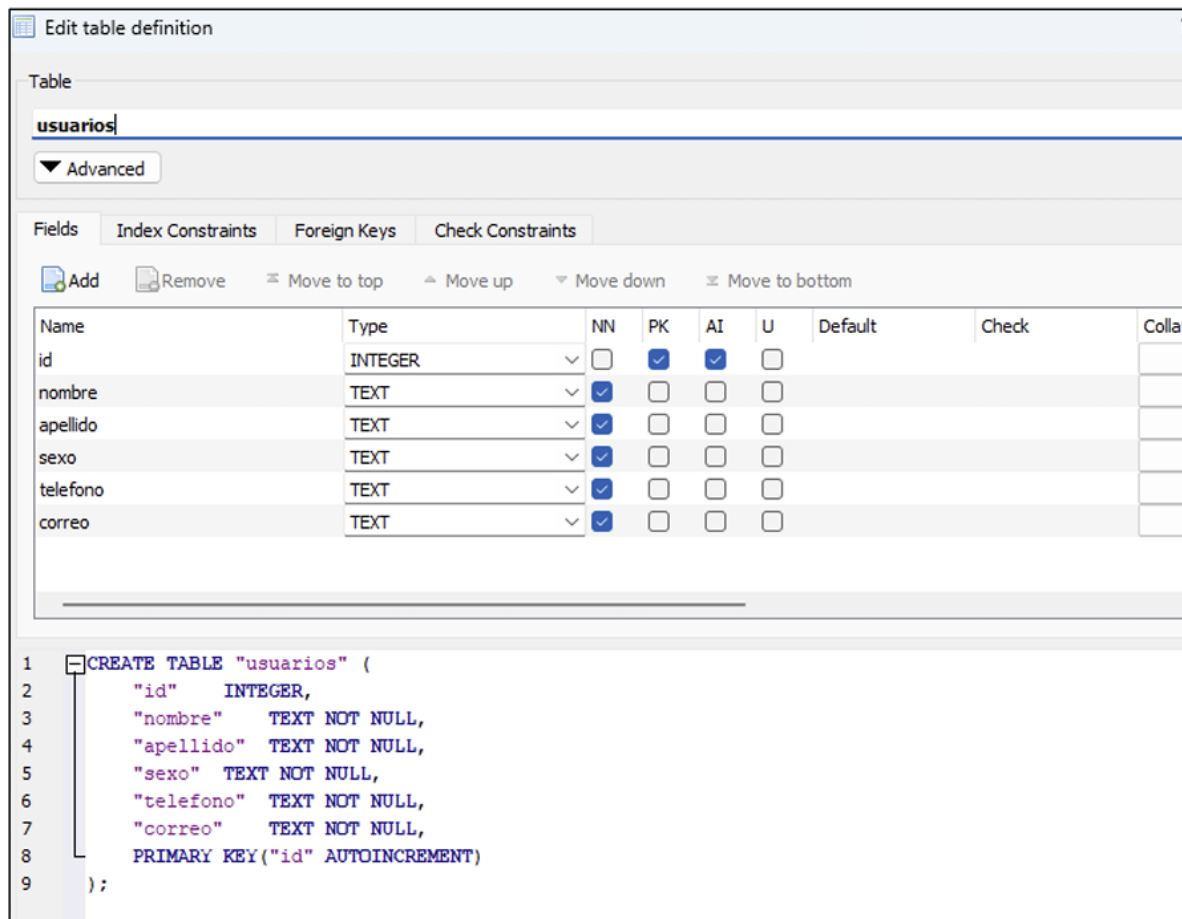
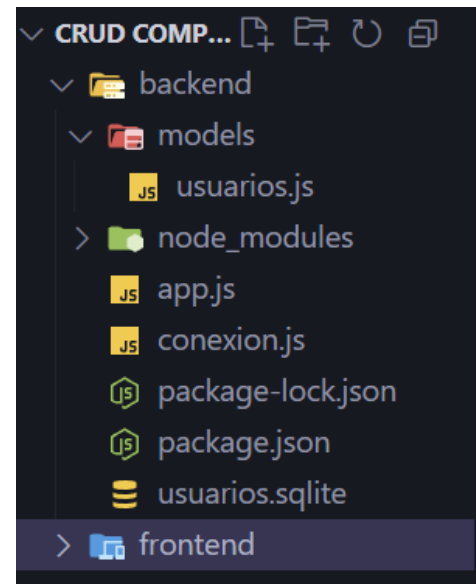
conexion.js: conecta con la base de datos SQLite.

app.js: configura y lanza el servidor Express.

usuarios.sqlite: archivo de base de datos.

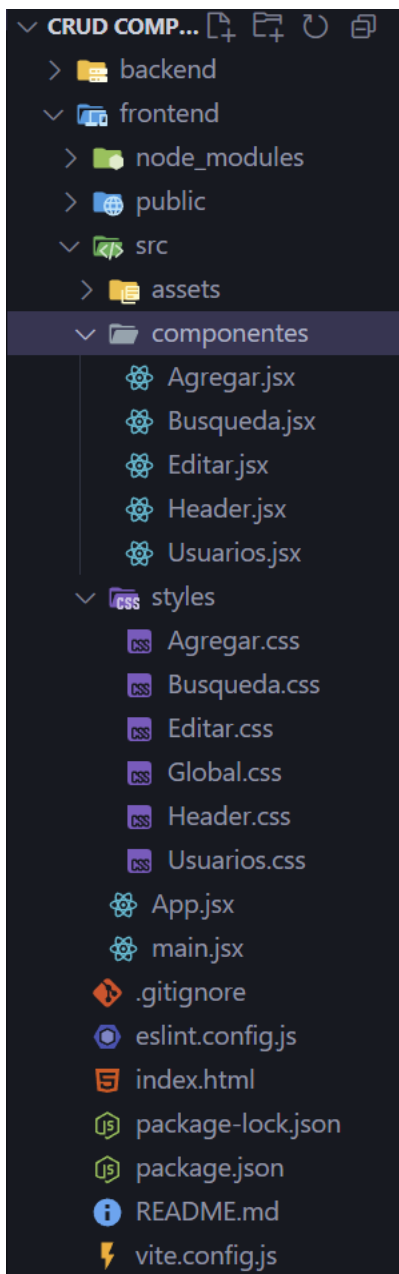
package.json: contiene configuración y dependencias.

node_modules/: librerías instaladas.



El sistema cumple una función clave al gestionar y almacenar los datos de los usuarios registrados en el **FRONTEND**, garantizando la persistencia de la información sin necesidad de depender de bases de datos externas como **MySQL** o **PostgreSQL**.

Gracias a su estructura ligera y autónoma, la aplicación se mantiene **portátil y fácil de desplegar**, permitiendo su ejecución en diversos entornos sin configuraciones complejas. Esto no solo optimiza la accesibilidad del sistema, sino que también reduce la carga de infraestructura, favoreciendo su implementación rápida y flexible.



Este proyecto contiene todo el código y los recursos utilizados para desarrollar la interfaz de usuario del Frontend, construido con React como biblioteca principal.

La aplicación ofrece una experiencia interactiva y dinámica, permitiendo a los usuarios:

- ✓ Registrar nuevos usuarios de manera sencilla.
- ✓ Buscar registros específicos mediante filtros.
- ✓ Editar información existente con facilidad.
- ✓ Eliminar usuarios de forma eficiente.

El sistema está diseñado a través de componentes reutilizables, asegurando una estructura modular que facilita la gestión de interacciones con el Backend mediante solicitudes HTTP, permitiendo una comunicación fluida y eficiente con la base de datos.

Pantalla de inicio

Gestión de Usuarios
Administra y organiza la información de manera eficiente

Lista De Usuarios Registrados

[Agregar](#)

Buscar: Ejemp: Nombre, Apellido, ID o Correo

ID	Nombre	Apellido	Sexo	Teléfono	Correo	Acciones
0	Isaac	Cardenas	Masculino	6673545230	isaacardenas@gmail.com	Editar Eliminar
1	Jose	Guerrero	Masculino	6673237380	joseguerrero@gmail.com	Editar Eliminar
4	Rosa	Martinez	Femenino	6612554318	rosamartinez@hotmail.com	Editar Eliminar
7	Javier	Zaragoza	Masculino	6673895210	javierzaragoza@gmail.com	Editar Eliminar

Agregar Usuario

Gestión de Usuarios

Agregar Usuario

Nombre:

Apellido:

Sexo:

Teléfono:

Correo:

[Guardar](#) [Cancelar](#)

ID	Nombre	Apellido	Sexo	Acciones
0	Isaac	Cardenas	Masculino	Editar Eliminar
1	Jose	Guerrero	Masculino	Editar Eliminar
4	Rosa	Martinez	Femenino	Editar Eliminar
7	Javier	Zaragoza	Masculino	Editar Eliminar

Usuario Agragado

The screenshot shows a web browser window with the address bar at localhost:5173. The page has a dark blue header with the title 'Gestión de Usuarios' and a subtitle 'Administra y organiza la información de manera eficiente'. Below the header is a light blue section titled 'Lista De Usuarios Registrados'. In this section, there is a purple 'Agregar' button and a search bar labeled 'Buscar' with a placeholder 'Ejemp: Nombre, Apellido, ID o Correo'. Below the search bar is a table with 7 columns: ID, Nombre, Apellido, Sexo, Teléfono, Correo, and Acciones. The table contains 5 rows of user data. Each row has 'Editar' and 'Eliminar' buttons in the 'Acciones' column.

ID	Nombre	Apellido	Sexo	Teléfono	Correo	Acciones
0	Isaac	Cardenas	Masculino	6673545230	isaacardenas@gmail.com	<button>Editar</button> <button>Eliminar</button>
1	Jose	Guerrero	Masculino	6673237380	joseguerrero@gmail.com	<button>Editar</button> <button>Eliminar</button>
4	Rosa	Martinez	Femenino	6612554318	rosamartinez@hotmail.com	<button>Editar</button> <button>Eliminar</button>
7	Javier	Zaragoza	Masculino	6673895210	javierzaragoza@gmail.com	<button>Editar</button> <button>Eliminar</button>
9	Ana	Maribel	Femenino	6672914380	anamaribel@gmail.com	<button>Editar</button> <button>Eliminar</button>

Registro en SQLite

The screenshot shows the DB Browser for SQLite application. The main window displays the 'usuarios' table with 5 rows of data. The table has 7 columns: id, nombre, apellido, sexo, telefono, correo, and acciones. The data is as follows:

id	nombre	apellido	sexo	telefono	correo	acciones
4	Rosa	Martinez	Femenino	6612554318	rosamartinez@hotmail.com	
1	Jose	Guerrero	Masculino	6673237380	joseguerrero@gmail.com	
7	Javier	Zaragoza	Masculino	6673895210	javierzaragoza@gmail.com	
0	Isaac	Cardenas	Masculino	6673545230	isaacardenas@gmail.com	
9	Ana	Maribel	Femenino	6672914380	anamaribel@gmail.com	

The application also shows a right-hand panel with an 'Edit Database Cell' window and a 'Remote' section for connecting to a database.