

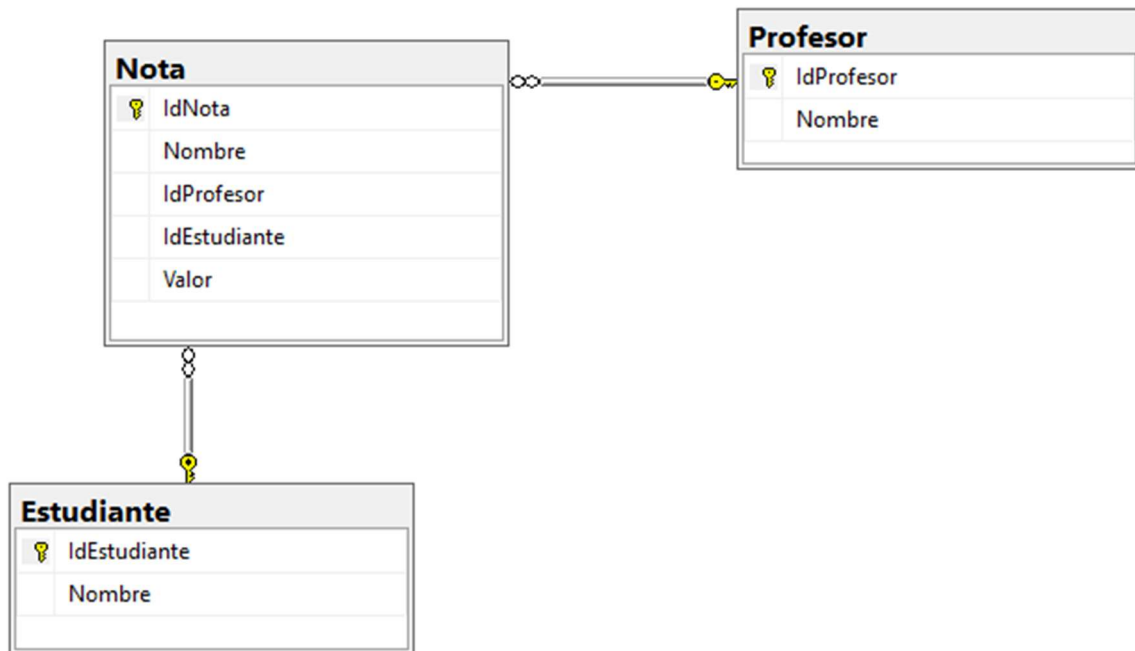
Prueba Técnica

En este documento adjunto imágenes de las vistas realizadas para la prueba técnica, en esta ocasión hice uso de **Angular 16** para el Frontend, el Backend esta construido con **.NET Framework**. Para la base de datos se usó **SQL Server**.

BASE DE DATOS

Evidencia 1. Modelo MER

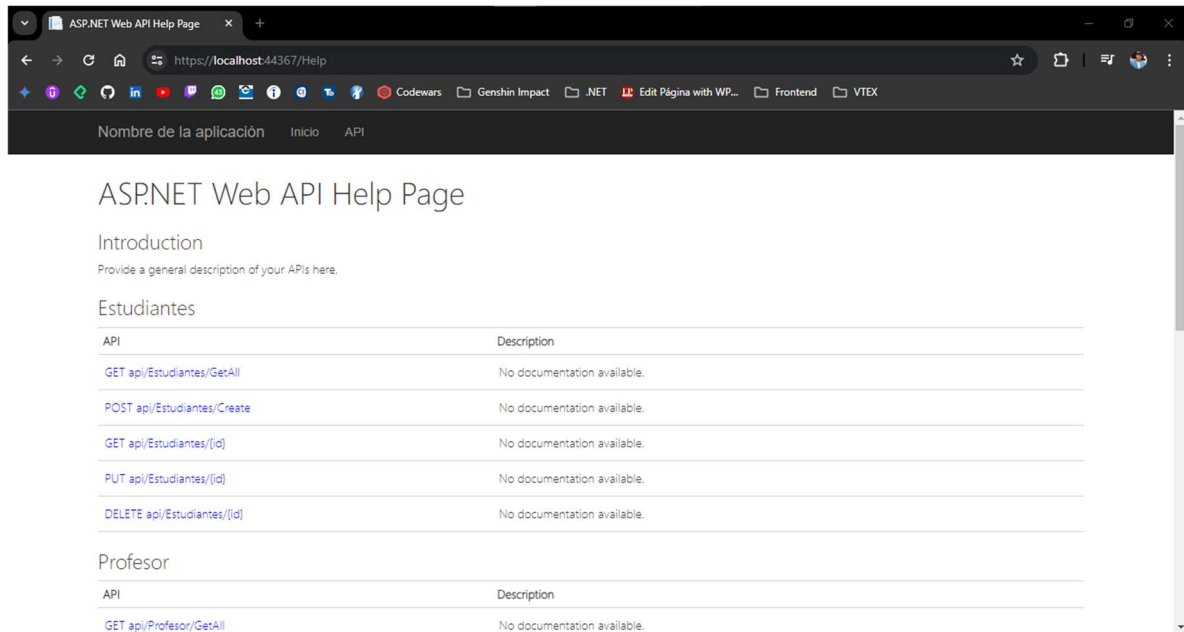
Evidencia de las tablas creadas, con los campos y llaves requeridas.



BACKEND

Evidencia 1. API.

En esta imagen podemos ver la construcción de la API en donde para cada entidad (Estudiante, Profesor, Nota) se realizaron los servicios **GET, GET BY ID, POST, PUT, DELETE**.



FRONTEND

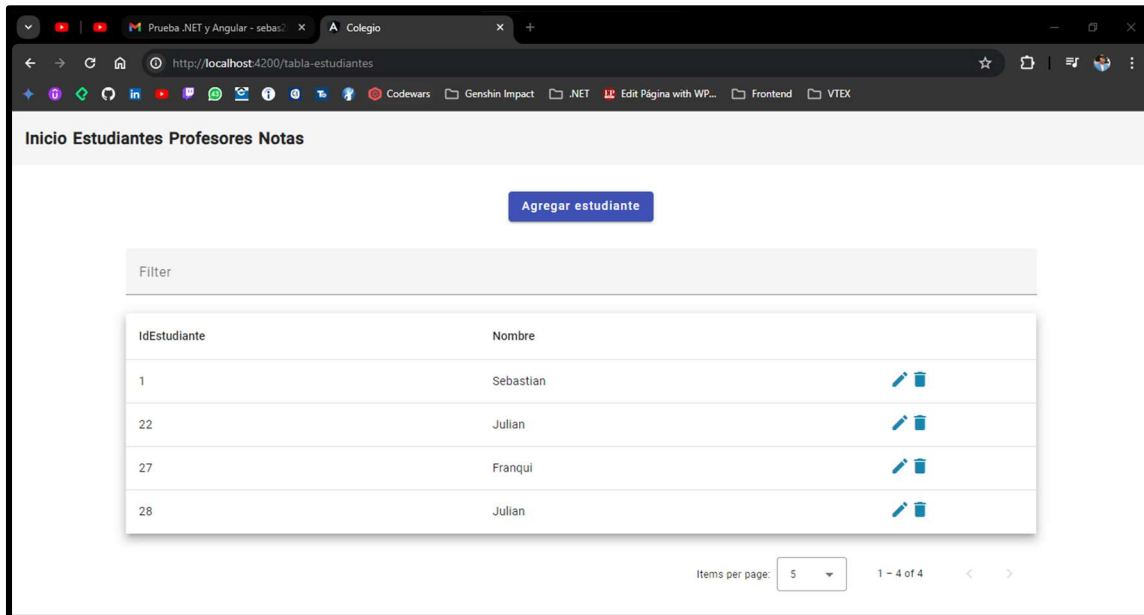
Vista 1. Home.

Aquí podemos visualizar lo que sería una pantalla de inicio, al igual que una barra de navegación en donde se tendrá acceso a los **CRUDS** de **Estudiantes**, **Profesores** y **Notas**.

Vista 2. Tabla Estudiantes.

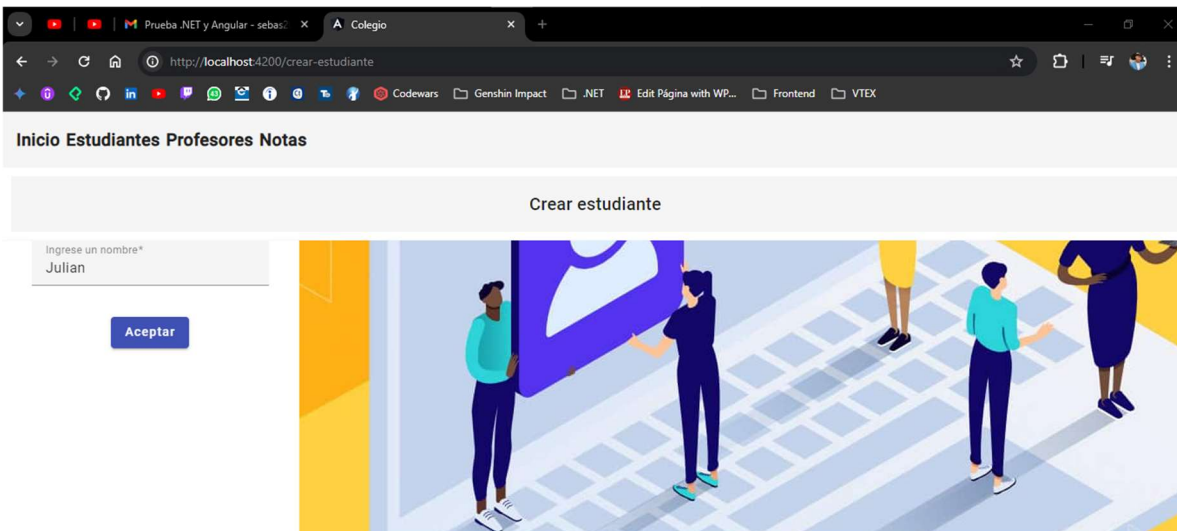
Como primera opción tenemos un botón para **agregar** un nuevo estudiantes, este redireccionara a la ventana de 'crear-estudiante'.

Después podemos visualizar la tabla en donde se obtienen todos los estudiantes registrados hasta el momento, la columna final corresponde a una columna de acciones, donde se podrá **editar** y **eliminar**. Cuando el usuario seleccione el icono de basura el registro seleccionado se eliminará. Adicionalmente se agregó un filtro y un paginador para mejorar la experiencia en la página.



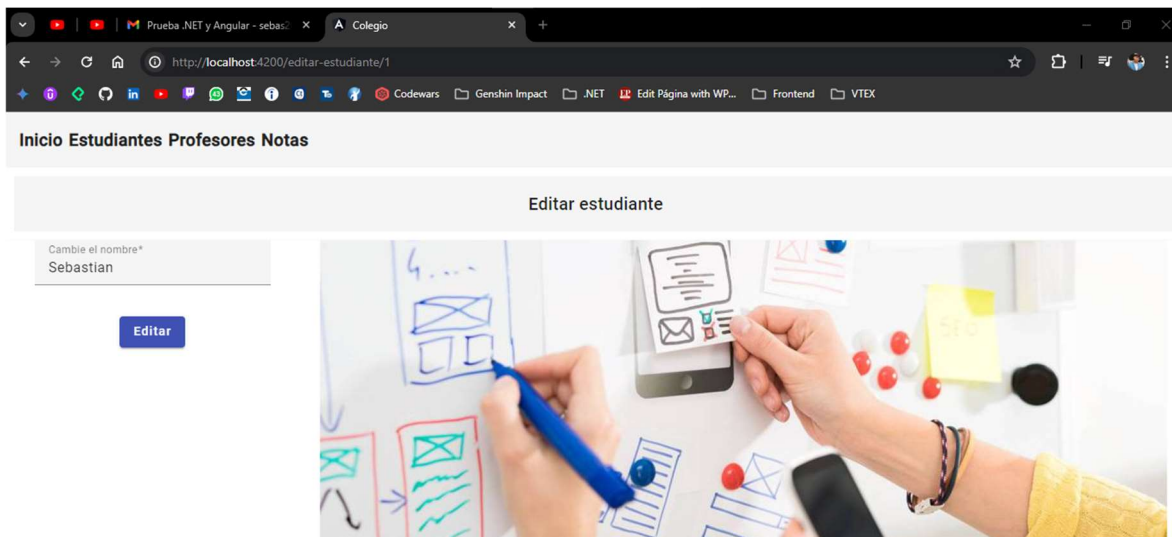
Vista 3. Crear estudiante.

En esta vista podemos ver el formulario para la **creación** de un nuevo usuario.



Vista 4. Editar estudiante.

En esta vista podemos ver el formulario para la **edición** de un usuario ya existente.

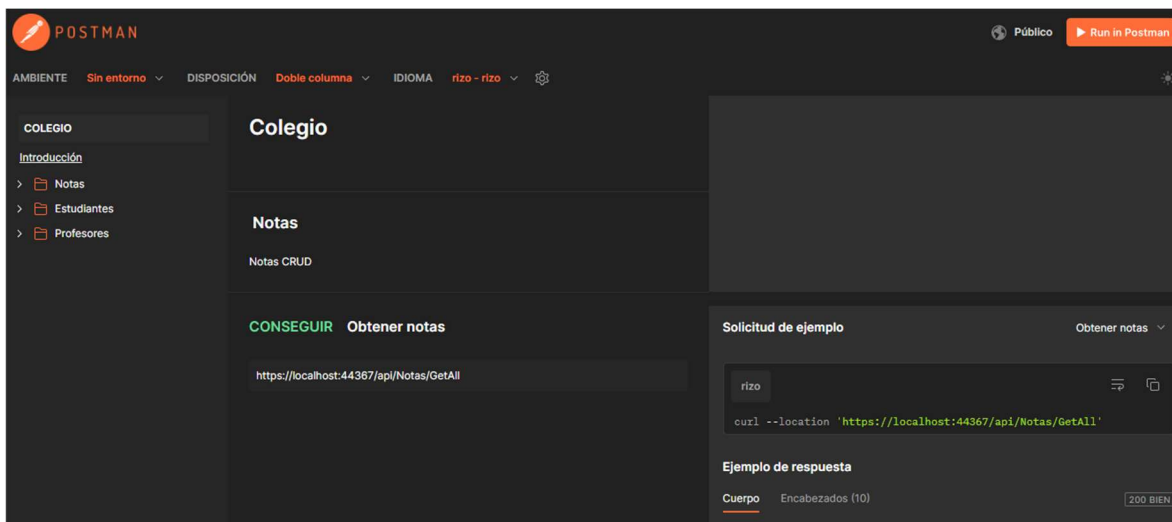


Las vistas son iguales tanto como para estudiante como para profesor y nota.

ARCHIVOS ADJUNTOS

Adjunto la documentación de la API en donde muestro como se deben consumir los servicios.

Documentación: <https://documenter.getpostman.com/view/20666214/2sA35Baj12>



Adjunto el script para la creación de la base de datos.