**Descripción del Proyecto**

Esta es una API de librería que permite a los usuarios gestionar un catálogo de libros. Los usuarios pueden agregar, editar, eliminar y obtener información sobre los libros que se encuentran almacenados en un objeto json remoto.

**Requisitos Previos**

Antes de comenzar, asegúrate de tener instalados estas son las versiones que yo tengo por si no te funcionan otras:

* [Node.js](https://nodejs.org/) (v22.9.0)
* [npm](https://www.npmjs.com/)(10.8.3)
* Opcional una herramienta para probar el api como postman o thunder client

**Instalación**

1. **Clona el repositorio**:

<https://github.com/DaniGarcia4207/Libreria>

cd Libreria

1. **Instala las dependencias**:

npm install

**Ejecución**

Para iniciar el servidor del back, utiliza los siguientes comandos:

cd server

node index.js

Esto iniciará la API en http://localhost:3001.

Para iniciar el servidor del front, utiliza el siguiente comando:

npm start

Esto iniciará el proyecto en http://localhost:3000.

**Uso de la API**

**Agregar un Libro**

**Endpoint**: POST /registro-libro

**Descripción**: Agrega un nuevo libro al archivo json.

**Campos requeridos**:

* title: (string) Título del libro.
* author: (string) Autor del libro.
* yearPublished: (number) Año de publicación.
* genre: (string) Género literario.

**Editar un Libro**

**Endpoint**: PUT /actualizar-libro/:id

**Descripción**: Edita un libro existente.

**Parámetros**:

* id: (string) ID del libro a editar.

**Campos requeridos**:

* title: (string) Título del libro.
* author: (string) Autor del libro.
* yearPublished: (number) Año de publicación.
* genre: (string) Género literario.

**Eliminar un Libro**

**Endpoint**: DELETE /eliminar-libro/:id

**Descripción**: Elimina un libro existente.

**Parámetros**:

* id: (string) ID del libro a eliminar.

**Obtener Libros**

**Endpoint**: GET /libros

**Descripción**: Obtiene una lista de todos los libros del json.

Retroalimentación: La prueba estaba muy chévere definitivamente todos los días se aprenden cosas nuevas, casi todas las cosas que realizo aquí son con conocimientos previos aunque si me toco investigar muchas cosas por mi cuenta ya que me complique un poco con lo del json, pensé que una base de datos para una sola tabla ya era mucho pero definitivamente hubiera sido más sencillo , de lo que si no tenía conocimiento fue sobre las pruebas automatizadas estuve leyendo e investigando, pero la verdad no las pude implementar por el tiempo ya que en este momento tengo muchas obligaciones en mi centro de formación y apenas pude sacar tiempo hoy para hacer la prueba por eso la hora de entrega creo que fue lo único que me causo inconvenientes de resto todo muy bien y aprendí mucho.

Las pruebas que yo siempre realizo son estas con Thunder Client Ejemplo:



