

## Book Corner

Presentado por:

Andrey Felipe Leguizamo Céspedes - 2201998

Juan Felipe Rojas De La Hoz - 2220070

Danna Valentina Prada Briceño- 2211334

Dylan Francisco Jimenez Sandoval - 2202049

Grupo:B1

UNIVERSIDAD INDUSTRIAL DE SANTANDER

PROGRAMACIÓN EN LA WEB

DOMINGO, 4 DE JUNIO 2023

usuarios del proyecto:

Sign Up

✕

Email:

vale@gmail.com

Valep12>

🔒

Sign up

Sign Up

✕

Email:

andres@gmail.com

Valep12>

🔒

Sign up

## OBJETIVOS GENERALES

Como objetivos generales se presenta el desarrollo final del frontend y backend para el proyecto llamado Book Corner, donde se emplean todo lo aprendido en el semestre, donde se busca mejorar lo anteriormente desarrollado en la Fase 1 con respecto al frontend, teniendo presente lo que no se pudo implementar, y agregando nuevas funciones que ayuden al fácil uso de la biblioteca “Book Corner”.

Se busca poder registrar un usuario y poder loguearse (Iniciar Sesión) con el usuario registrado siempre y cuando este sea permitido por el servidor, además de implementar un sistema de favoritos, donde dependiendo del tipo de libro que el usuario agregue a favoritos, la página empezará a recomendar libros parecidos.

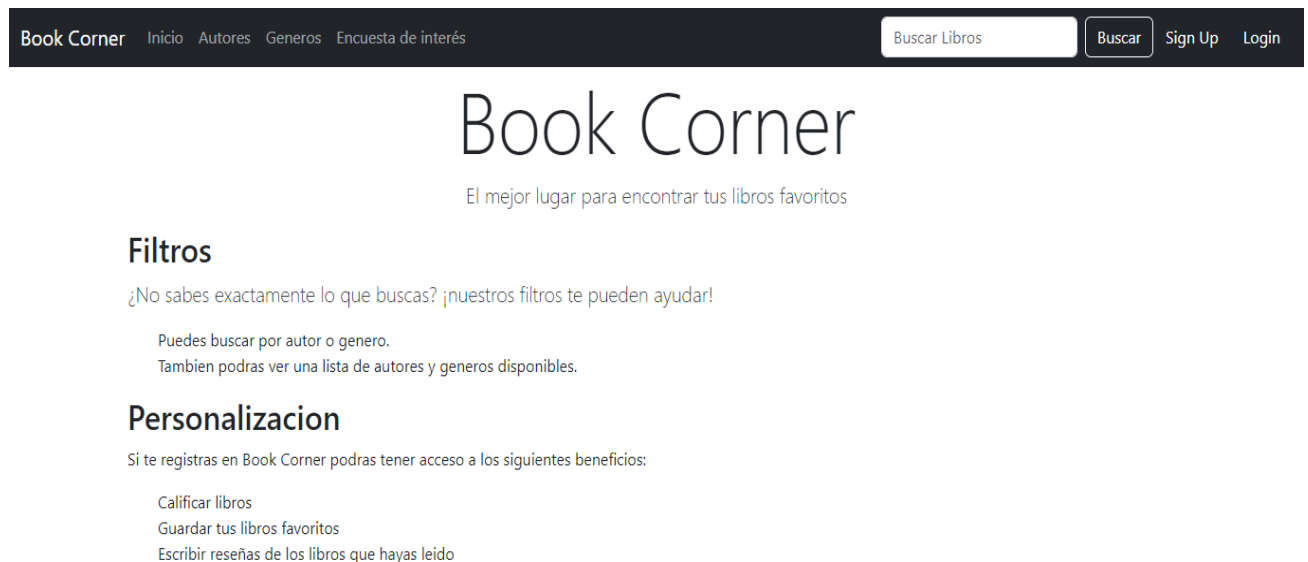
Lograr que la página “Book Corner” mantenga la esencia que tenía en su primera fase, pero con agregados estéticos que sean visualmente favorables para todo tipos de usuarios, con colores suaves a la vista y un diseño digno de una librería.

## OBJETIVOS ESPECÍFICOS

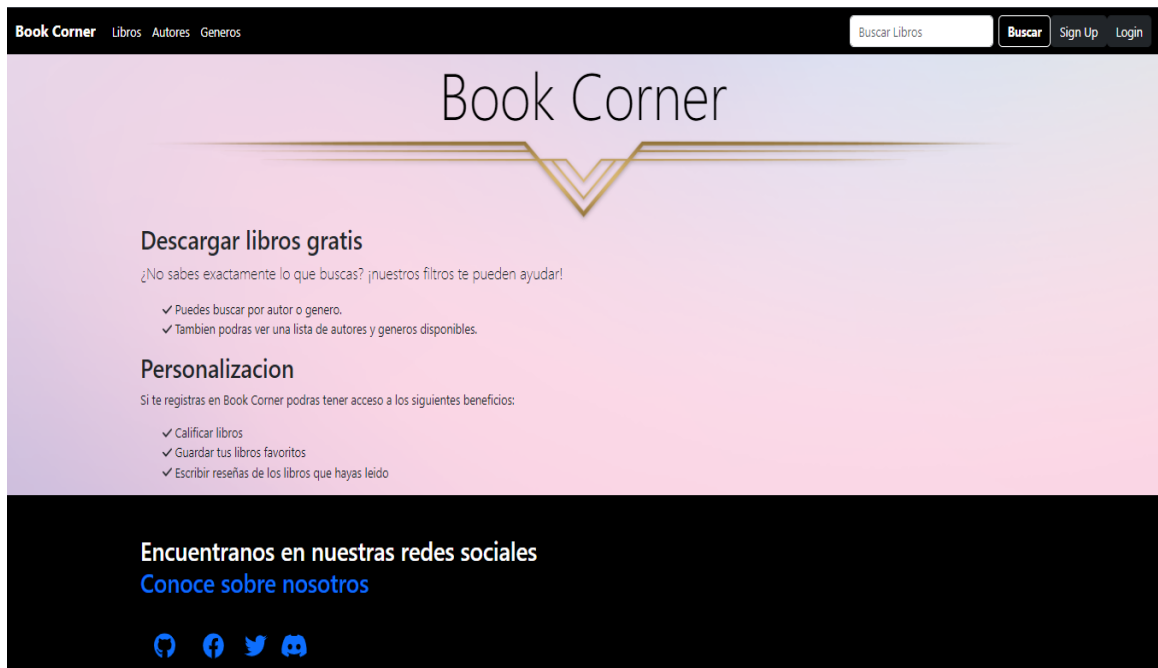
El objetivo es la creación de una plataforma en línea utilizando los diferentes conocimientos obtenidos durante el transcurso de esta materia, donde los usuarios pueden buscar, descubrir y leer libros de forma gratuita, mostrando un diseño atractivo con diferentes géneros y biografías de distintos autores, estableciendo una estructura de base de datos para almacenar y gestionar la información de los libros.

## Imagen de la primera página del sitio

### Antigua:

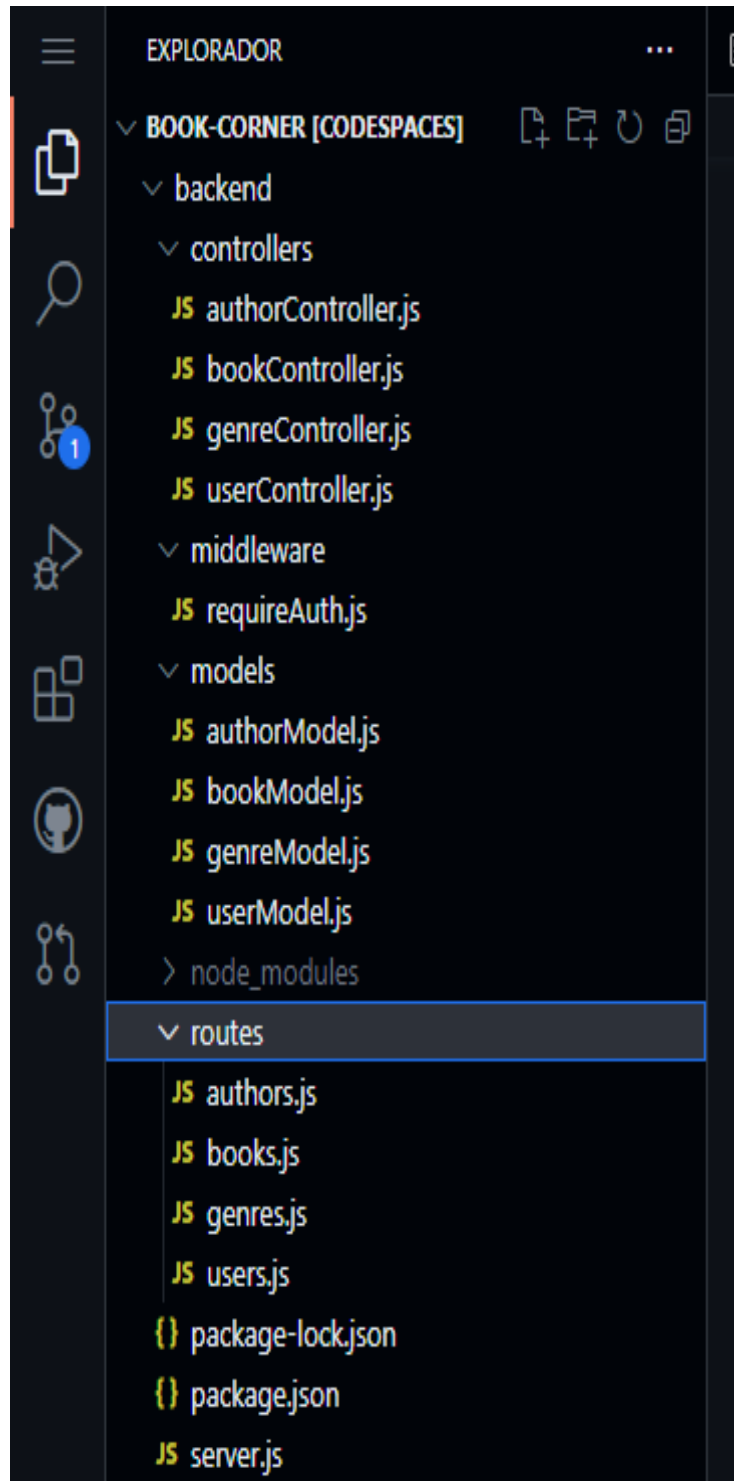


### Nueva:

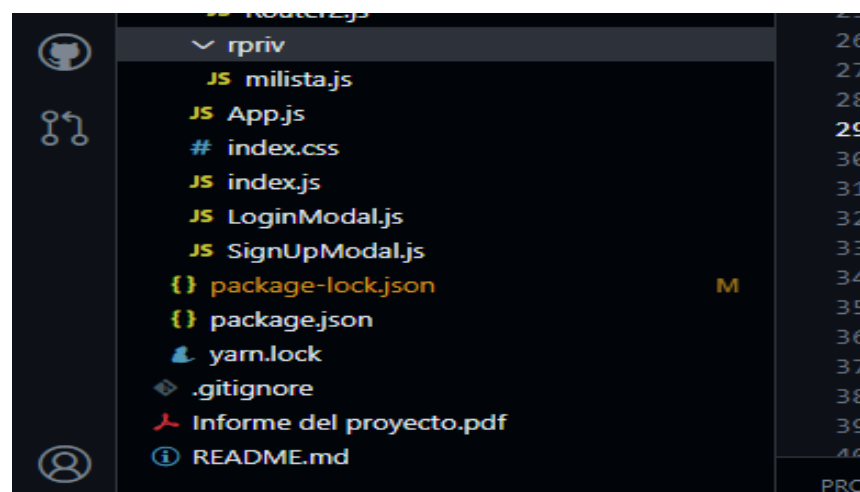
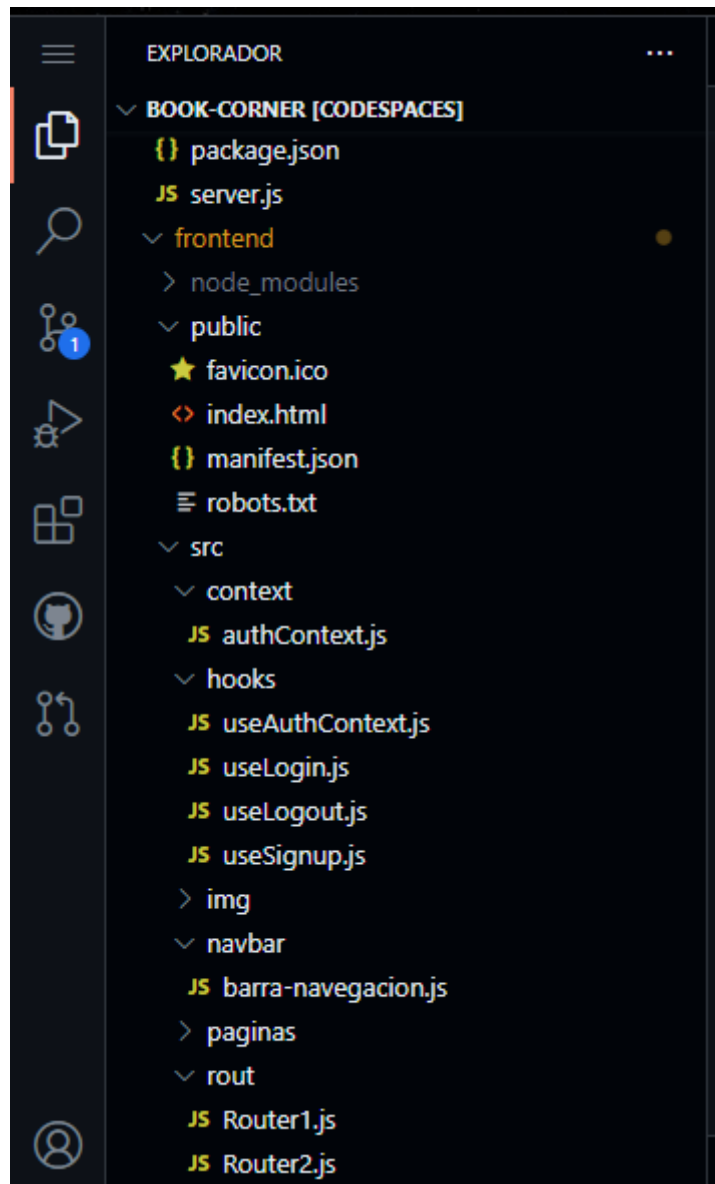


## Estructura jerárquica del sitio

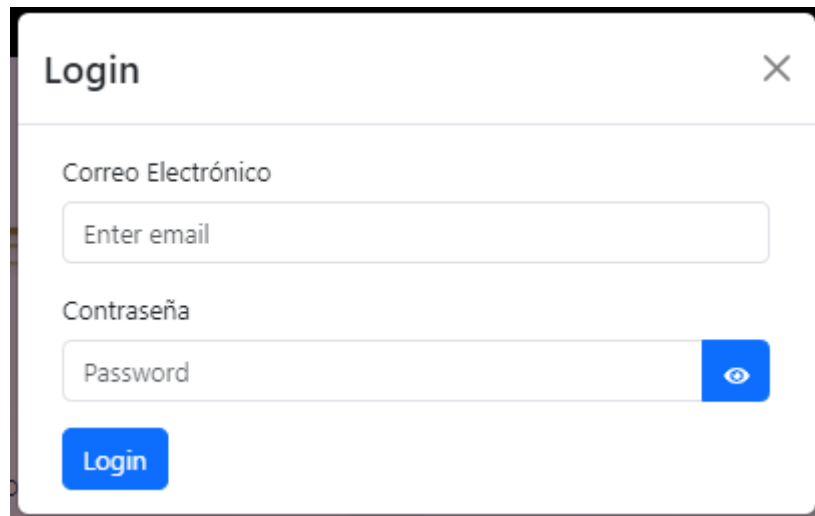
### Backend:



Fronted:

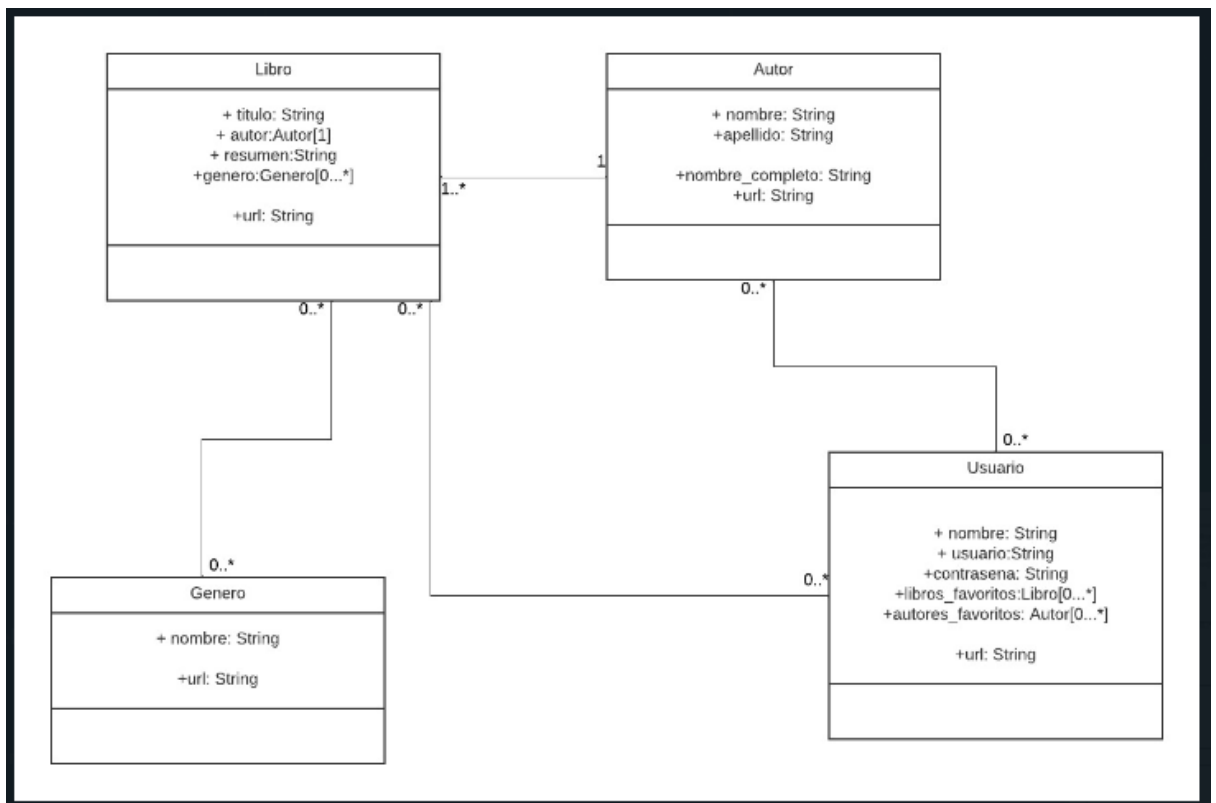


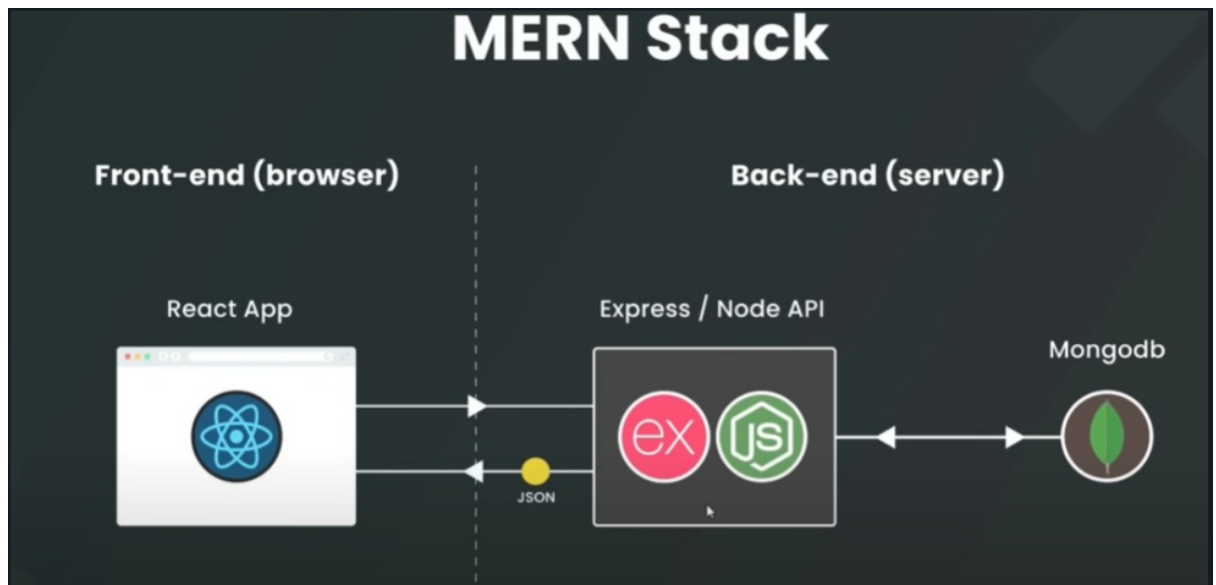
**Usuarios y credenciales para acceder al sitio mediante "login"**  
**según el siguiente formato: usuario | password**



A login form titled "Login" with a close button (X) in the top right corner. It contains two input fields: "Correo Electrónico" (Email) with a placeholder "Enter email", and "Contraseña" (Password) with a placeholder "Password" and a blue eye icon for toggling visibility. A blue "Login" button is positioned below the password field.

**Modelo relacional de la BD. ( gráfico )**





## Listas con Explicación de todas las funcionalidades:

- i. **lista1: funcionalidades del appWeb ( mencionar todas)**
- ii. **lista2: Nuevas funcionalidades implementadas en FASE-II (mencionarlas y explicar su funcionalidad)**
- iii. **lista3: funcionalidades solicitadas pero no implementadas ( explicar porque no se pudo implementar o por cual se reemplaza)**

## Lista1:

Inicio:

`import React from 'react':` Importa el módulo React desde la biblioteca 'react'. Esto es necesario para poder utilizar la sintaxis de JSX y crear componentes de React.

`import { BarraNavegacion } from`  
`'../navbar/barra-navegacion':` Importa el componente BarraNavegacion desde el archivo `'../navbar/barra-navegacion'`. Sin embargo, este componente no se utiliza en el código que has proporcionado.



`import Accordion from 'react-bootstrap/Accordion':` Importa el componente `Accordion` desde el archivo `'react-bootstrap/Accordion'`. Esto indica que estás utilizando el componente de acordeón proporcionado por la biblioteca `react-bootstrap`.

`function BasicExample() { ... }:` Define un componente de función llamado `BasicExample`. Este componente renderiza el acordeón con los elementos dentro de su retorno de llamada.

`<Accordion defaultActiveKey="0"> ... </Accordion>:` El componente `Accordion` envuelve los elementos del acordeón y establece `defaultActiveKey` en `"0"`, lo que indica que el primer elemento del acordeón estará abierto por defecto.

`<Accordion.Item eventKey="0"> ... </Accordion.Item>:` Este es el primer elemento del acordeón. `eventKey` se establece en `"0"` para identificar este elemento.

`<Accordion.Header>Gabriel García Márquez</Accordion.Header>:` El componente `Accordion.Header` define el encabezado del primer elemento del acordeón, que muestra el nombre `"Gabriel García Márquez"`.

`<Accordion.Body> ... </Accordion.Body>:` El componente `Accordion.Body` define el cuerpo del primer elemento del acordeón. Contiene información sobre `Gabriel García`

Márquez, como su biografía y detalles sobre su vida y obra.

`<Accordion.Item eventKey="1"> ... </Accordion.Item>`: Este es el segundo elemento del acordeón. `eventKey` se establece en "1" para identificar este elemento.

`<Accordion.Header>Miguel de Cervantes Saavedra</Accordion.Header>`: El componente `Accordion.Header` define el encabezado del segundo elemento del acordeón, que muestra el nombre "Miguel de Cervantes Saavedra".

`<Accordion.Body> ... </Accordion.Body>`: El componente `Accordion.Body` define el cuerpo del segundo elemento del acordeón. Contiene información sobre Miguel de Cervantes Saavedra, como su biografía y detalles sobre su vida y obra.

`export default BasicExample;`: Exporta el componente `BasicExample` para que pueda ser utilizado en otros archivos.

Autores:

`import React from 'react'`: Importa el módulo `React` desde la biblioteca `'react'`. Esto es necesario para poder utilizar la sintaxis de JSX y crear componentes de `React`.

`import { BarraNavegacion } from`  
`'../navbar/barra-navegacion';`: Importa el componente BarraNavegacion desde el archivo `'../navbar/barra-navegacion'`. Este componente representa la barra de navegación de la página y se utilizará en el componente Inicio.

`import { Container, Row, Col } from 'react-bootstrap';`  
Importa los componentes Container, Row y Col desde la biblioteca 'react-bootstrap'. Estos componentes se utilizan para crear una estructura de diseño en filas y columnas en la página.

`export default function Inicio() { ... }`: Exporta el componente de función Inicio como el componente principal de este archivo.

`<BarraNavegacion />`: Renderiza el componente BarraNavegacion, que representa la barra de navegación de la página. Este componente se muestra en la parte superior de la página.

`<Container> ... </Container>`: El componente Container proporcionado por react-bootstrap envuelve el contenido principal de la página. Ayuda a establecer un ancho máximo y centrar el contenido.

`<Row> ... </Row>`: El componente Row crea una fila horizontal para contener las columnas. Las filas se utilizan para organizar el contenido en la página.

`<Col> ... </Col>`: El componente Col crea una columna dentro de una fila. Se utiliza para dividir el contenido en columnas y establecer la distribución del diseño.

`<h1 className='display-1 text-center'>Book Corner</h1>`: Renderiza un encabezado `<h1>` con el texto "Book Corner" en tamaño de visualización 1 y centrado horizontalmente en la página.

`<p className='lead text-center'>El mejor lugar para encontrar tus libros favoritos</p>`: Renderiza un párrafo con la clase 'lead' que muestra un mensaje de descripción centrado horizontalmente en la página.

`<h2>Filtros</h2>`: Renderiza un encabezado `<h2>` que muestra el título "Filtros".

`<p className='lead'>¿No sabes exactamente lo que buscas? ¡nuestros filtros te pueden ayudar!</p>`: Renderiza un párrafo con la clase 'lead' que proporciona información sobre cómo los filtros pueden ayudar a los usuarios a encontrar lo que están buscando.

`<ul style={{listStyle: "none"}}>`: Inicia una lista desordenada y establece el estilo de lista en "none" para que no haya viñetas.

21-22. `<li> ... </li>`: Renderiza elementos de lista `<li>` que describen las opciones de filtrado disponibles para los usuarios.

`<h2>Personalizacion</h2>`: Renderiza un encabezado `<h2>` que muestra el título "Personalización".

`<Nav.link href></Nav.link>`:

## Lista2:

### Rutas:

- `router.get("/", )`;; Esta ruta se utiliza para manejar solicitudes GET realizadas a la URL raíz de la aplicación.
- `router.get("/:id", )`;; Esta ruta se utiliza para manejar solicitudes GET realizadas a URL que tienen un parámetro de ID. Llama a la función `getAuthor` cuando se realiza una solicitud GET.
- `router.post("/", )`;; Esta ruta se utiliza para manejar solicitudes POST realizadas a la URL raíz de la aplicación.
- `router.delete("/:id", )`;; Esta ruta se utiliza para manejar solicitudes DELETE realizadas a URL que tienen un parámetro de ID..
- `router.patch("/:id", )`;; Esta ruta se utiliza para manejar solicitudes PATCH realizadas a URL que tienen un parámetro de ID.

### Controller:

- `res.status(200).json(book);`: Este código devuelve una respuesta con un libro en formato JSON. La respuesta tiene un estado de 200, lo que significa que la solicitud se ha completado correctamente. El objeto `book` se convierte en formato JSON y se envía como respuesta.

### **Lista3:**

- Modal en la información de contacto: Cuando se intentó cambiar la información de contacto de la lista al modal, llegaba a producir varios errores en el inicio y al momento de abrir el link de información, entonces se decidió dejarlo tal como estaba en la fase 1:
- Seguimiento de favoritos: Al momento de intentar implementar un sistema que indicará qué libros eran parecidos a los que se encontraban en los favoritos del usuario, generó inconvenientes que recomendaban cosas que no tenían similitud con los libros en favoritos, o a veces solo daba como referencia el mismo libro encontrado en favoritos.

## Encuesta:

Crea un componente de React llamado Encuesta que representa un formulario de encuesta con varios campos de entrada y opciones de selección. El formulario está envuelto en un contenedor flexible de Bootstrap (Container) y utiliza estilos de Bootstrap para el diseño y apariencia visual. El componente se exporta para que pueda ser utilizado en otros archivos y se puede integrar fácilmente en una aplicación de React para recolectar datos de encuesta de los usuarios.

## Bibliografía

- <https://react-bootstrap.netlify.app/docs/components/carousel/>

- <https://react-bootstrap.netlify.app/docs/components/accordion/>
- <https://react-bootstrap.netlify.app/docs/components/modal>
- [https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express\\_Nodejs](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs)
- <https://net-ninja-pro.teachable.com/p/net-ninja-pro/>

## Conclusiones

Para finalizar este proyecto llegamos a la conclusión de la importancia de la union correcta entre un frontend(Visual) y un backend(Base de datos), logrando que una página web se sienta completa con respecto a lo que se buscar ofrecer, por ende es importante que sea fácil de usar para que todo tipo de usuarios puedan sentirse a gusto con la página.

Por ende para lograr todo esto es importante el conocimiento previo y técnico sobre todos los lenguajes, además de tener la habilidad para poder trabajar con algo tan importante como las BD.

## Anexo 1

Como funcionalidad con respecto al diseño utilizamos Bootstrap, este es conocido por su capacidad de respuesta y su capacidad para la creación de diseños que se adaptan a diferentes dispositivos y tamaños de pantalla, asegurándose de la utilización de clases de diseño responsivo de Bootstrap, esto para que el fronted se vea bien en dispositivos móviles, tabletas y computadores.

## Anexo 2

Dando a conocer las funcionalidades implementadas:

- Carrusel de géneros: funcionalidad que muestra un carrusel con imágenes de libros relacionados con diferentes géneros. Esta funcionalidad permite a los usuarios explorar libros de su interés de manera visual y atractiva.
- Acordeón con los autores: Un acordeón que muestra información sobre diferentes autores. Esto brinda a los usuarios la posibilidad de obtener detalles específicos sobre los escritores y explorar su trabajo de manera organizada y fácil de navegar.
- Encuesta: La encuesta utiliza campos de formulario adecuados para cada pregunta, como campos de texto para el nombre completo, email, dirección y ciudad, etc.



- Login: Un sistema de inicio de sesión que permite a los usuarios registrarse y acceder a sus cuentas personalizadas.

Funcionalidades no implementadas:

- Presentación personal modal con imágenes.
- Reemplazar el acordeón por un modal para la sección "Nosotros".
- Mejorar el inicio de sesión y registro con la adición de un captcha