



SOUNDSEAT

# SOUNDSEAT

## FASE 2



# Índice

Introducción  
Metodología kanban  
Funcionalidades Implementadas  
Resultados  
Conclusión

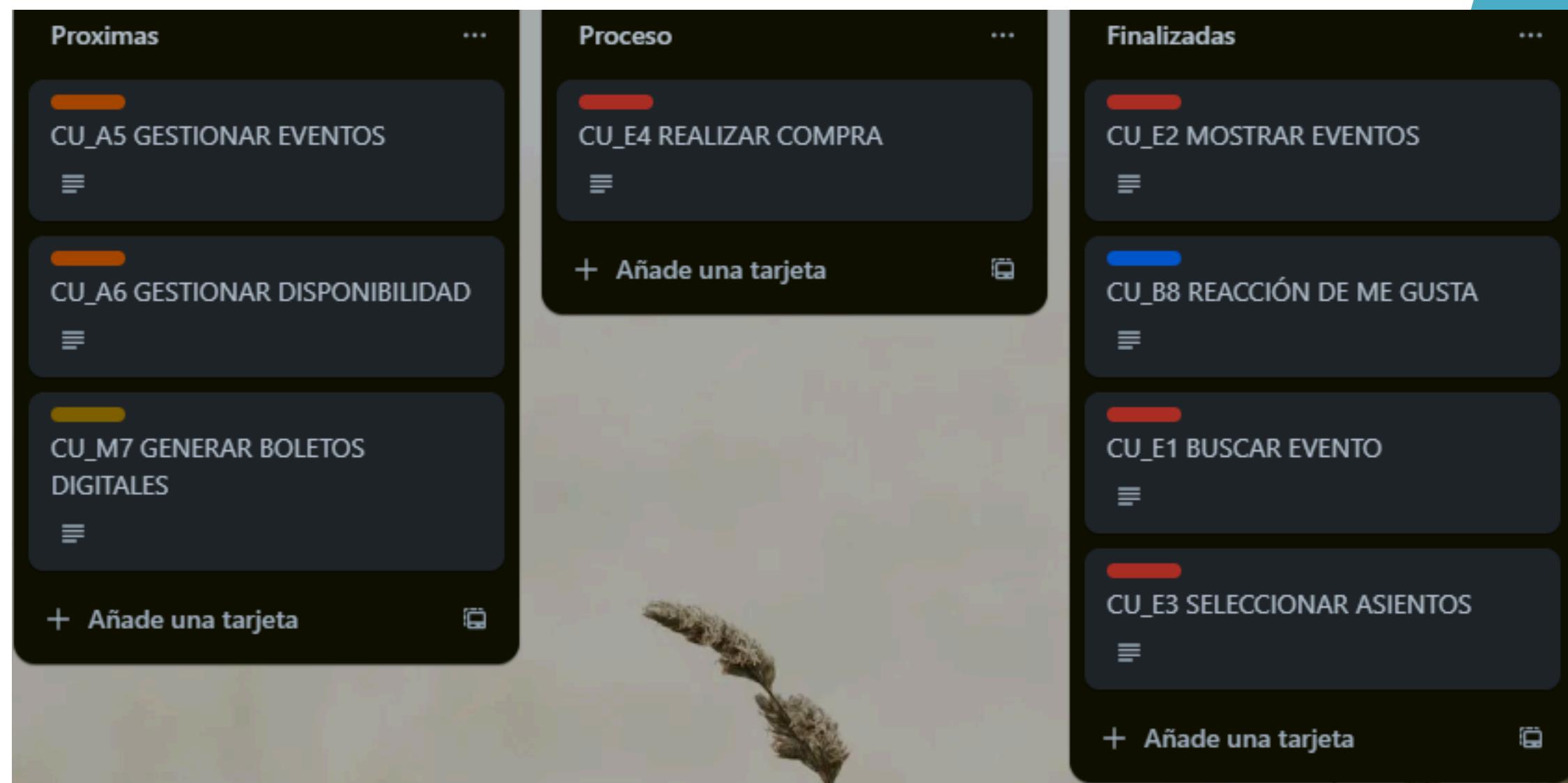
# Introducción

En esta Fase 2, hemos pasado de la planificación al desarrollo práctico de las principales funcionalidades del sistema.

- Implementar un sistema intuitivo para búsqueda y compra de boletos.
- Proveer una experiencia de usuario mejorada a través de un diseño interactivo.
- Establecer una conexión sólida y segura en el backend usando herramientas y servicios que Firebase pone a disposición para proyectos web, así como React para la parte del diseño.

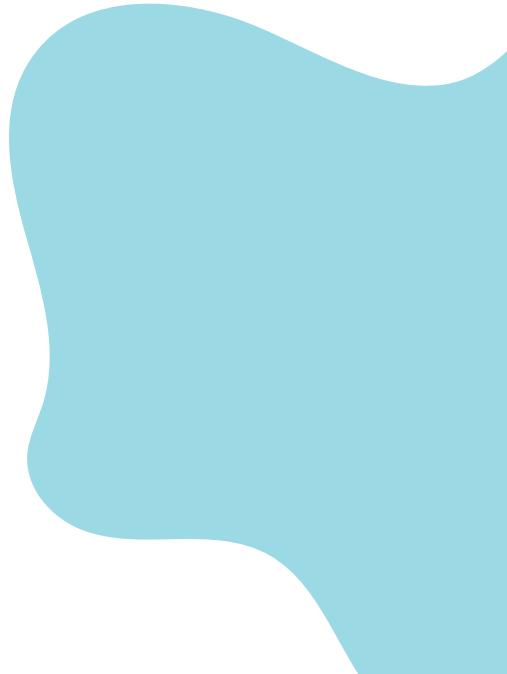
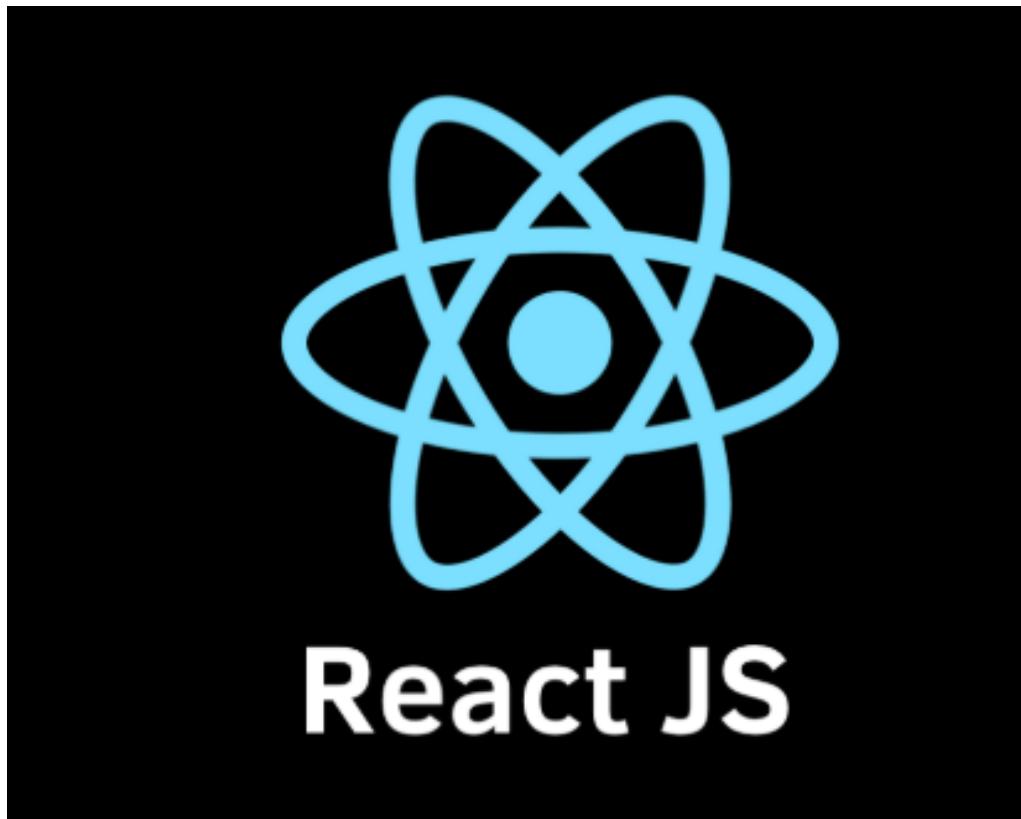
## 2. Metodología Kanban

Esta metodología ha permitido seguir organizando y priorizando nuestras tareas a lo largo de la Fase 2. Este enfoque nos ha proporcionado una visión clara del progreso, identificando rápidamente los avances y áreas de mejora así como presentar avances concretos en cada etapa del proyecto.



### 3. Funcionalidades Implementadas

El enfoque principal fue convertir las ideas planteadas en funcionalidades tangibles que mejoran la experiencia del usuario y garantizan un sistema sólido y eficiente. Estas funcionalidades se dividen en dos componentes clave: Frontend y Backend.



# Backend

El backend, desarrollado con Firebase, asegura la gestión eficiente de datos en tiempo real, mejorando la confiabilidad y seguridad del sistema.

```
>>

#####
##   ##   ##   ##   ##   ##   ##   ##   ##
##   ##   ##   ##   ##   ##   ##   ##   ##
#####   ##   #####   ##   #####   ##   #####
##   ##   ##   ##   ##   ##   ##   ##   ##   ##
##   ######   ##   #####   ##   #####   ##   #####
##   ##   ##   ##   ##   ##   ##   ##   ##   ##

You're about to initialize a Firebase project in this directory:

C:\Users\rpu03\OneDrive\Escritorio\APP\mi-react-app

? Are you ready to proceed? Yes

*** Project Setup
```

## ¡Conexión a Firebase exitosa!

### Autenticación con Firebase

Correo electrónico		Contraseña	<input type="button" value="Registrarse"/>	<input type="button" value="Iniciar Sesión"/>
--------------------	--	------------	--	---

### Prueba de Firebase Realtime Database

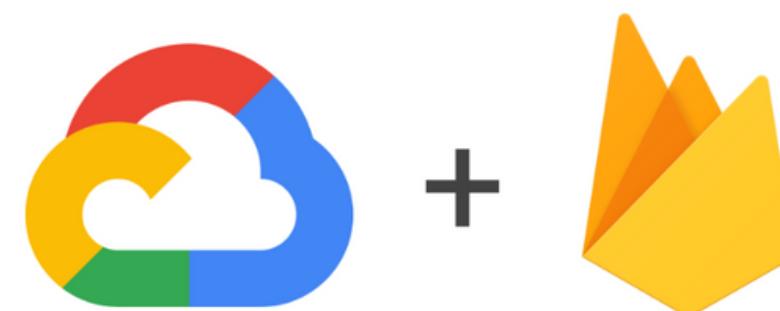
holaaaaa	<input type="button" value="Guardar en Firebase"/>	<input type="button" value="Leer desde Firebase"/>
----------	--	--

Datos obtenidos: holaaaaa

### Notas

Escribe una nota	<input type="button" value="Agregar Nota"/>
------------------	---

Total de notas: 0



# Backend

La propuesta para modelar la página con la arquitectura presentada incluye las siguientes características



Firestore

```
1  firebase-backend
2
3      └── functions
4          ├── routes
5              ├── events.js
6              ├── seats.js
7              └── transactions.js
8          ├── controllers
9              ├── eventsController.js
10             ├── seatsController.js
11             └── transactionsController.js
12          ├── firebaseConfig.js
13          ├── index.js
14          └── package.json
15
16      └── serviceAccountKey.json
```

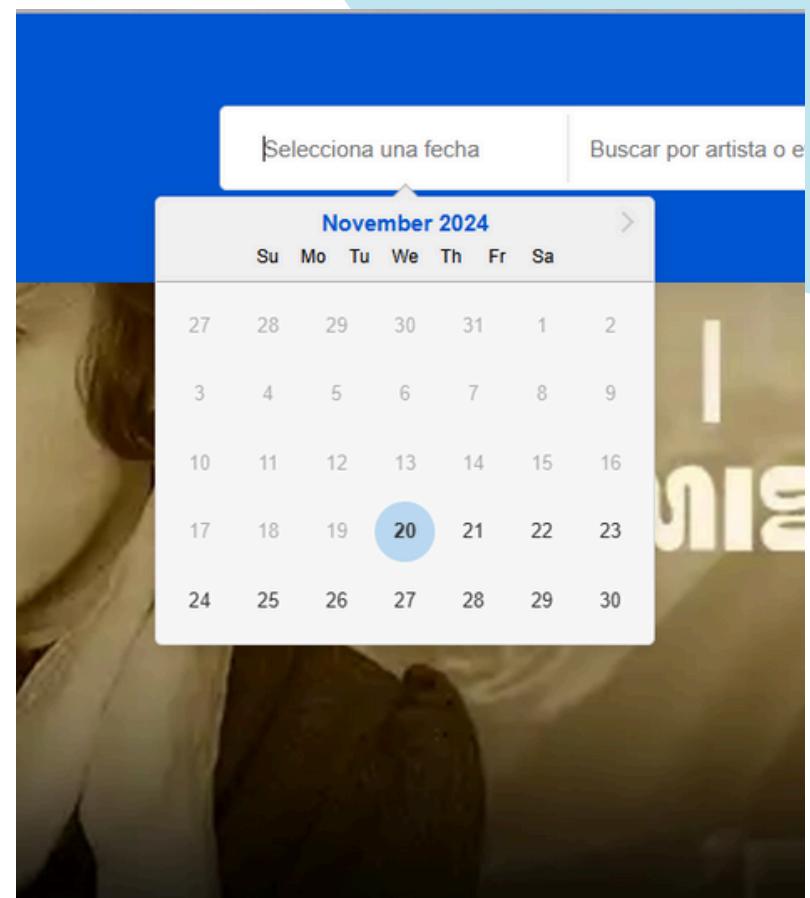
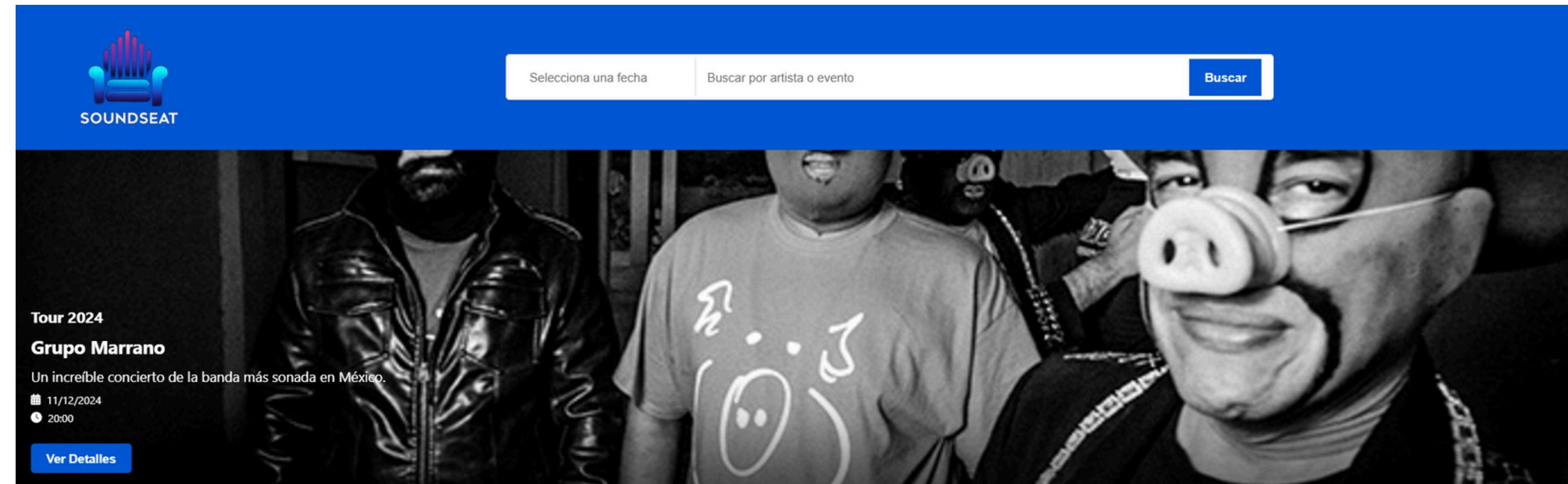
The screenshot shows a code editor interface with several tabs at the top: EXPLORER, ... (Ellipsis), JS eventsController.js, JS seatsController.js, JS transactionsController.js X, JS index.js, {} package.json, and JS firebaseConfig.js. The main area displays the contents of the transactionsController.js file:

```
1  const { db } = require('../firebaseConfig');
2
3  const createTransaction = async (req, res) => {
4      try {
5          const { eventId, seats, total } = req.body;
6          await db.collection('transactions').add({ eventId, seats, total });
7          res.status(201).send('Transacción creada correctamente');
8      } catch (error) {
9          res.status(500).send(error.message);
10     }
11 };
12
13 const getTransactions = async (req, res) => {
14     try {
15         const snapshot = await db.collection('transactions').get();
16         const transactions = snapshot.docs.map(doc => ({ id: doc.id, ...doc.data() }));
17     }
18 }
```

At the bottom of the editor, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS.

# Frontend

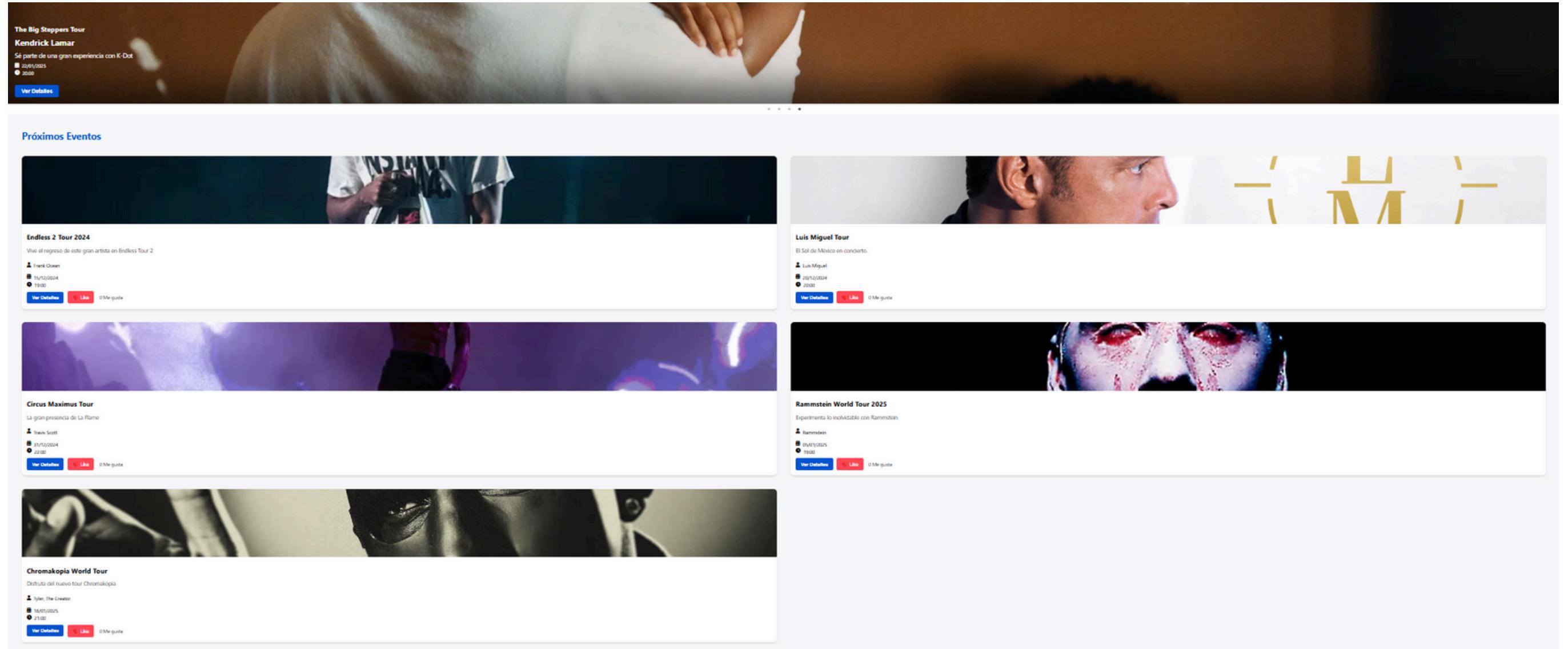
La integración de React en el desarrollo del frontend nos ha permitido crear una interfaz interactiva que optimiza el rendimiento del sistema y la satisfacción del usuario.



## Cabecera Principal de SoundSeat

Presenta un buscador con dos campos: uno para seleccionar la fecha del evento (calendario desplegable) y otro para buscar por artista o nombre del evento.

# Eventos Destacados (Carrusel):



El carrusel de eventos destacados es una herramienta visual clave que captura la atención del usuario al ingresar a la plataforma.

Presenta los eventos más relevantes, nuevos o populares, incentivando a los usuarios a explorar más detalles.

# Dedalles de Evento



## The Big Steppers Tour

Sé parte de una gran experiencia con K-Dot

### Detalles del evento

Artista:  
Kendrick Lamar

Fecha:  
22/01/2025

Hora:  
20:00

### Descripción

Kendrick Lamar, uno de los artistas más influyentes del hip-hop contemporáneo, llevará su increíble talento y sus letras poderosas a este evento único. El concierto presentará los temas más representativos de su carrera, incluyendo los de su último álbum. Además, Kendrick Lamar es conocido por su increíble capacidad para conectar con el público, lo que garantiza una experiencia musical y emocional que quedará grabada en la memoria de todos los asistentes.

### Redes Sociales

Visita las redes sociales de Kendrick Lamar

Facebook

Twitter

Instagram

[Volver a la página de inicio](#)

[Comprar boletos](#)

Esta sección representa una página de detalle de un evento específico, diseñada para proporcionar al usuario toda la información relevante del evento seleccionado.

Se implementan botones para visitar las redes sociales del artista (Facebook, Twitter e Instagram). Proporciona enlaces para que los usuarios sigan al artista y se mantengan informados de futuros eventos o noticias.

# Interfaz de selección de asientos para un evento específico en la plataforma SoundSeat.

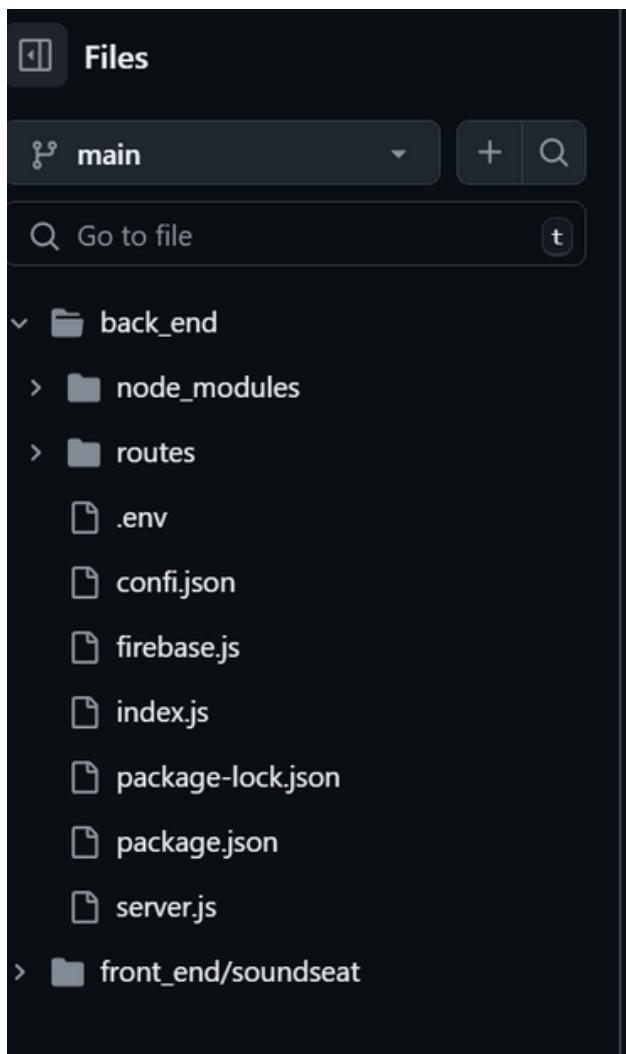
The screenshot shows a seat selection interface for a concert. At the top, there's a search bar with fields for 'Selecciona una fecha' (Select a date), 'Buscar por artista o evento' (Search by artist or event), and a 'Buscar' (Search) button. Below the search bar, the event details are displayed: 'Selecciona tus asientos para The Big Steppers Tour' (Select your seats for The Big Steppers Tour), 'Kendrick Lamar - 22/01/2025 a las 20:00' (Kendrick Lamar - January 22, 2025 at 20:00), and a dropdown menu for 'Número de asientos a seleccionar' (Number of seats to select) set to 1. The main area features a stadium seating diagram with sections labeled: Oriente, Tlaxcala Alta, Tlaxcala Baja, Poniente, Centro Oriente, Centro Poniente, Pronto Sur, and Pronto Norte. The 'Baja Central' section is highlighted in green. To the right, a summary box titled 'Resumen de Asientos' (Seat Summary) shows 'No se han seleccionado asientos.' (No seats selected) and a total price of 'Total: \$0'. Below this, another section for seat selection is shown, identical to the first one but with a different stadium diagram. It includes a 'Volver al estadio' (Return to the stadium) button, the text 'Tlaxcala Alta - \$700 por asiento' (Tlaxcala Alta - \$700 per seat), and a legend indicating seat status: TA1 (Available), TA2 (Occupied), TA3 (Selected), and TA4 (Available). A summary box shows 'Asiento: TA3, Precio: \$700' (Seat: TA3, Price: \$700), 'Eliminar Todos' (Delete All), 'Comprar' (Buy), and a total price of 'Total: \$700'.

Esta página tiene como objetivo simplificar y personalizar la experiencia del usuario durante la compra de boletos, ofreciendo un flujo claro y lógico que permite:

- Seleccionar las áreas preferidas del recinto.
- Elegir asientos específicos con control total.
- Revisar el precio total antes de confirmar la compra.

En general, esta interfaz es clave para que los usuarios puedan visualizar y gestionar sus compras de manera eficiente, asegurando que el sistema sea transparente y fácil de usar.

# Plataforma de colaboración

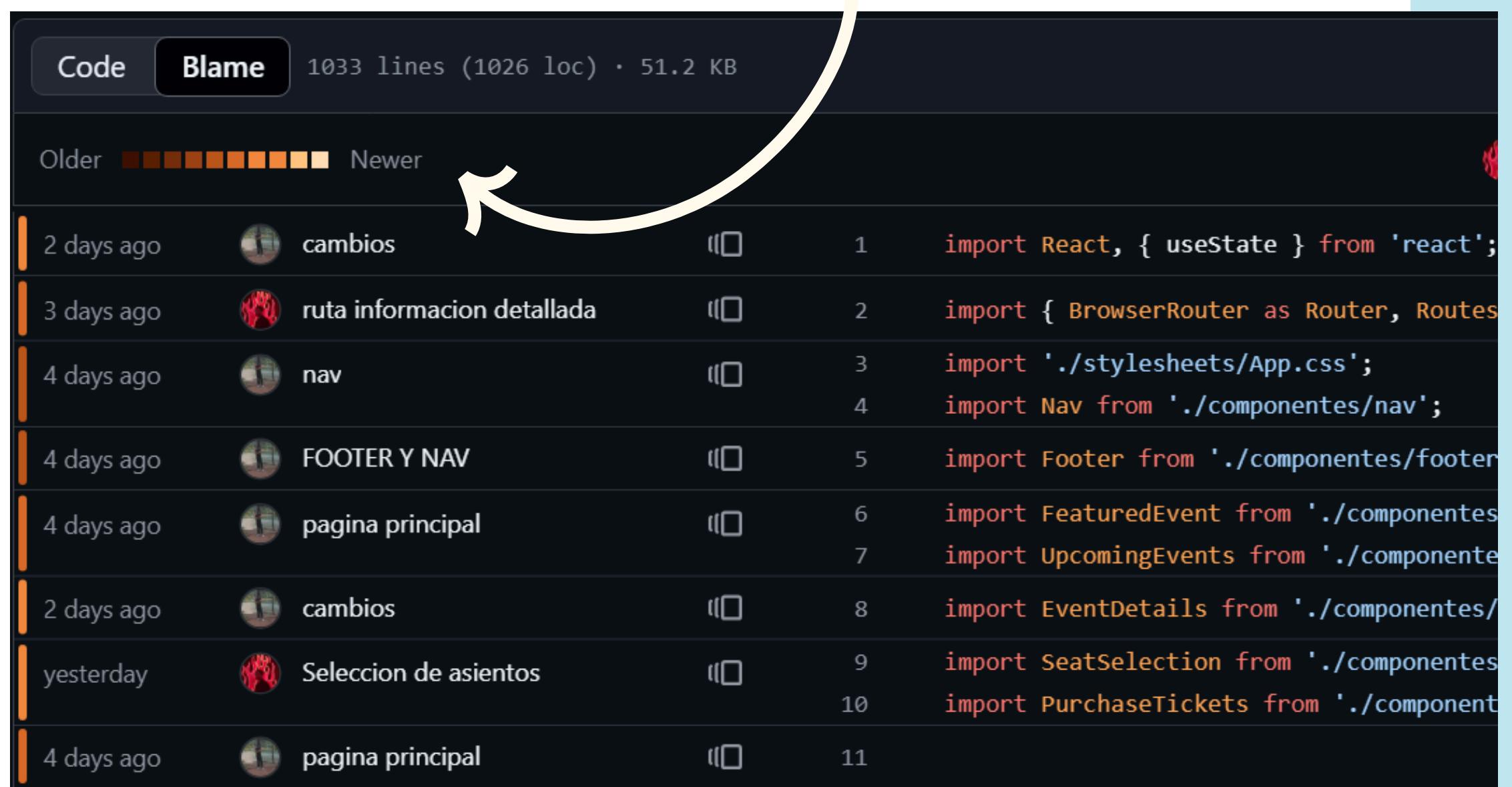


soundseat\_ads / front\_end / soundseat / src / App.js

Code Blame 1033 lines (1026 loc) · 51.2 KB

```
function App() {  
  return (  
    <Routes>  
      <Route path="/" element={<EventList />} />  
      <Route path="/event/:id" element={<EventDetails />} />  
      <Route path="/buy-tickets" element={<SeatSelection />} />  
      <Route path="/purchase" element={<PurchaseTickets />} />  
    </Routes>  
  );  
}  
;
```

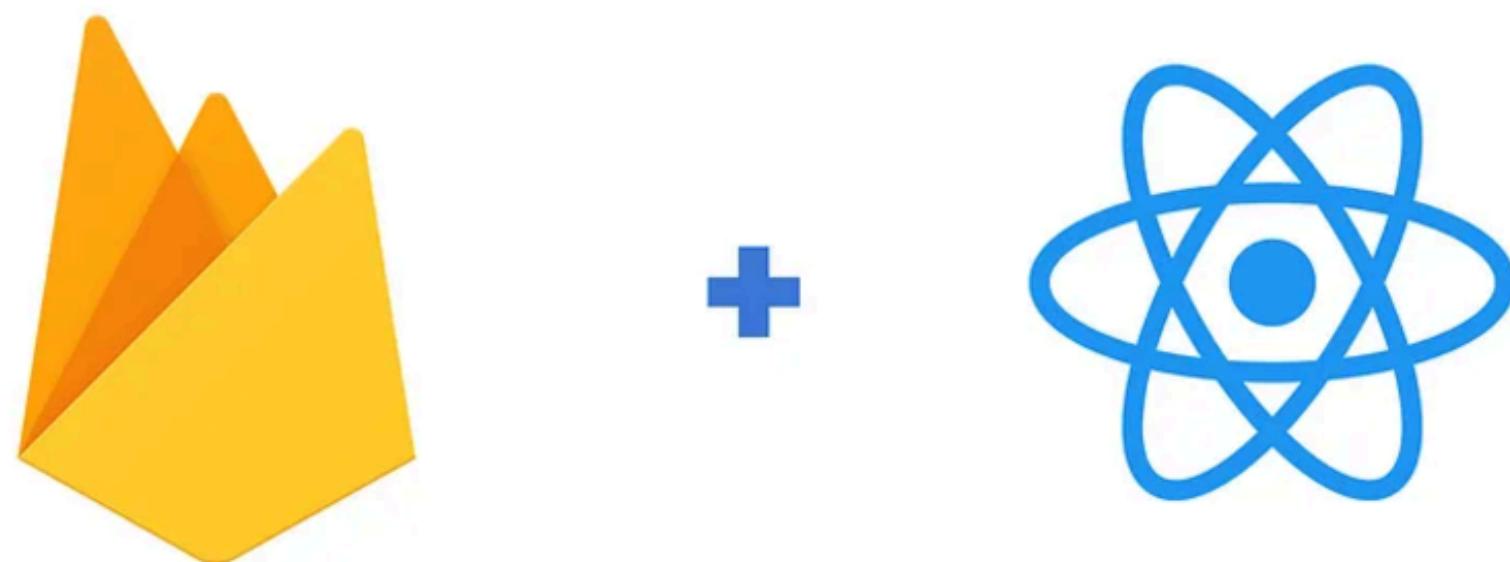
La organización jerárquica del proyecto en carpetas y archivos permite a los desarrolladores identificar rápidamente dónde realizar cambios o buscar funcionalidades específicas.



Older	Newer	Author	Date	Message	Line Number	Code Snippet
			2 days ago	cambios	1	import React, { useState } from 'react';
			3 days ago	ruta informacion detallada	2	import { BrowserRouter as Router, Routes,
			4 days ago	nav	3	import './stylesheets/App.css';
			4 days ago	FOOTER Y NAV	4	import Nav from './componentes/nav';
			4 days ago	pagina principal	5	import Footer from './componentes/footer'
			2 days ago	cambios	6	import FeaturedEvent from './componentes/
			yesterday	Seleccion de asientos	7	import UpcomingEvents from './componente'
			4 days ago	pagina principal	8	import EventDetails from './componentes/
					9	import SeatSelection from './componentes/
					10	import PurchaseTickets from './componente
					11	

## Próximos Pasos: Fase 3

- El siguiente paso clave será integrar el backend y frontend para tener una aplicación completamente funcional. En esta etapa se asegurará que los usuarios puedan interactuar con la interfaz visual (frontend) y que las solicitudes se manejen correctamente a través del backend.
- Se realicen pruebas completas para garantizar que todas las funcionalidades, desde la búsqueda de eventos hasta la compra de boletos, funcionen de manera fluida.
- Se ajuste el diseño y la funcionalidad según retroalimentación de las pruebas para optimizar la experiencia final.



# Conclusión

En esta segunda fase del proyecto SoundSeat, se desarrollaron propuestas sólidas para el backend y el frontend. El backend, basado en Firebase, está diseñado para gestionar datos en tiempo real y proporcionar una estructura modular que facilite su mantenimiento y escalabilidad. Por otro lado, el frontend en React presenta una experiencia interactiva enfocada en la visualización de eventos y la interacción del usuario. Estas propuestas establecen una base sólida para la fase final, donde se integrarán ambos componentes para lograr una aplicación completamente funcional y operativa.





GRACIAS  
A TODOS