

Projeto 2º Bimestre – Pikachu

NextEvents

Disciplina: Web Mobile

Danilo Abude Gigliotti – 10443431

Gabriel Palacios - 10365109 Gabriel

Branco Urbinati – 1044376

Visão Geral

O projeto é uma aplicação Next.js que usa:

- **Context API do React** para compartilhar dados dos eventos entre componentes.
- **Componentes funcionais React** com hooks (useState, useEffect).
- **API Ticketmaster** para buscar eventos no Brasil.
- **Roteamento dinâmico** para página de evento específico.
- **Estilos CSS modulares** para estilizar componentes isoladamente

Tipo de Website Escolhido: Plataforma de Busca de Eventos no Brasil

O projeto desenvolvido é um **website para descoberta de eventos pelo Brasil**, utilizando dados da API oficial da Ticketmaster, uma das maiores plataformas de venda de ingressos do mundo. O objetivo principal é oferecer ao usuário uma **experiência prática e interativa** para:

- **Navegar por eventos culturais, esportivos, musicais, entre outros**, disponíveis no país;
- **Filtrar eventos por cidade**, com apenas alguns cliques;
- **Visualizar informações importantes** de cada evento como nome, imagem, data e local;
- **Acessar detalhes** do evento em uma página dedicada (com uso de slugs dinâmicos).

Esse tipo de aplicação é extremamente útil em tempos de **grande demanda por entretenimento local**, turismo e planejamento de lazer. Além disso, é uma excelente forma de exercitar conceitos modernos de **desenvolvimento web com React e Next.js**, como:

- Componentes reutilizáveis;
- Hooks (useState, useEffect, useContext);
- Context API;
- Consumo de APIs externas;
- Rotas dinâmicas e navegação otimizada.

Estrutura do WebSite

O projeto contém:

- Um **contexto global** para gerenciar eventos.
- Um **menu lateral** para navegar e buscar eventos por cidade.
- Um **componente card** que exibe detalhes do evento.
- Uma **página home** que lista os eventos, atualizada conforme a cidade selecionada.
- Layout responsivo e estilizado.

Contexto para eventos: EventosContext

Esse componente cria um contexto que armazena a lista de eventos e a cidade selecionada, além de uma função para buscar eventos da API.

```
1  "use client";
2
3  import { createContext, useState, useContext } from "react";
4
5  const EventosContext = createContext();
6
7  export const EventosProvider = ({ children }) => {
8    const [eventos, setEventos] = useState([]);
9    const [cidadeSelecionada, setCidadeSelecionada] = useState(null);
10
11    const buscarEventos = async (cidade = null) => {
12      try {
13        setCidadeSelecionada(cidade);
14        const base = "https://app.ticketmaster.com/discovery/v2/events.json";
15        const params = new URLSearchParams({
16          apiKey: "dwzDtpPdrpnAALSuOP28vm1poN38P95Y",
17          countryCode: "BR",
18          size: "100",
19        });
20
21        if (cidade) params.append("city", cidade);
22
23        const url = `${base}?${params.toString()}`;
24        const res = await fetch(url);
25        const data = await res.json();
26
27        const eventosRecebidos = data._embedded?.events || [];
28        setEventos(eventosRecebidos);
29      } catch (err) {
30        console.error("Erro na requisição:", err);
31      }
32    };
33
34    return (
35      <EventosContext.Provider value={{ eventos, buscarEventos, cidadeSelecionada }}>
36        {children}
37      </EventosContext.Provider>
38    );
39  };
40
41  export const useEventos = () => useContext(EventosContext);
42
```

- EventosProvider envolve toda a aplicação para fornecer o estado global.
- buscarEventos faz a chamada para a API e atualiza os estados.
- useEventos é um hook para usar o contexto mais facilmente.

Componente do menu lateral: Menu.jsx

Este componente exibe o menu lateral com links fixos e uma lista de cidades que possuem eventos.

```

1  "use client";
2
3  import { useState, useEffect } from "react";
4  import { usePathname } from "next/navigation";
5  import style from "../Menu.module.css";
6  import { useEventos } from "../EventosContext";
7
8  export default function Menu() {
9    const [menuAberto, setMenuAberto] = useState(false);
10   const [cidadesComEvento, setCidadesComEvento] = useState([]);
11   const { buscarEventos } = useEventos();
12   const pathname = usePathname();
13
14   const estaNaPaginaDeEvento = pathname.startsWith("/evento/");
15   const toggleMenu = () => setMenuAberto(!menuAberto);
16
17   useEffect(() => {
18     async function carregarCidadesComEvento() {
19       const res = await fetch(
20         "https://app.ticketmaster.com/discovery/v2/events.json?apikey=dwzDtpDrpnAALSuOP28vm1poN38P95Y&countryCode=BR&size=200"
21       );
22       const data = await res.json();
23       const eventos = data._embedded?.events || [];
24
25       const cidades = new Set(
26         eventos
27           .map((e) => e._embedded?.venues?.[0]?.city?.name)
28           .filter(Boolean)
29       );
30
31       setCidadesComEvento(Array.from(cidades).sort());
32     }
33     carregarCidadesComEvento();
34   }, []);
35
36   useEffect(() => {
37     if (estaNaPaginaDeEvento) setMenuAberto(false);
38   }, [pathname]);

```

```

40   return (
41     <nav className={style.navMenu}>
42       <ul className={style.lista}>
43         <li><a href="/" className={style.botaoVerde}>Home</a></li>
44         <li><a href="/about" className={style.botaoVerde}>Sobre</a></li>
45         <li><a href="/contact" className={style.botaoVerde}>Contato</a></li>
46
47         {!estaNaPaginaDeEvento && (
48           <li>
49             <button onClick={toggleMenu} className={style.botaoVerde}>
50               Buscar Eventos por Cidade
51             </button>
52             {menuAberto && (
53               <div>
54                 {cidadesComEvento.map((cidade) => (
55                   <button
56                     key={cidade}
57                     className={style.botaoVerde}
58                     onClick={() => buscarEventos(cidade)}
59                   >
60                     {cidade}
61                   </button>
62                 ))}
63               </div>
64             )}
65           </li>
66         )}
67       </ul>
68     </nav>
69   );
70 }
71

```

- Exibe links fixos para "Home", "Sobre" e "Contato".
- Ao clicar no botão "Buscar Eventos por Cidade", abre uma lista de cidades obtidas da API.
- Ao clicar numa cidade, chama buscarEventos(cidade) para filtrar eventos.
- Fecha o menu automaticamente ao navegar para a página de evento.

Componente do card de evento: EventoCard.jsx

Exibe o card com a imagem, nome, data e local do evento, e faz o card inteiro um link para a página do evento.

```

EventoCard.jsx > ...
1  "use client";
2  import styles from "../eventoCard.module.css";
3  import Link from "next/link";
4
5  export default function EventoCard({ evento }) {
6    if (!evento) return null;
7
8    const image = evento.images?.[0]?.url;
9    const data = evento.dates?.start?.localDate;
10   const local = evento._embedded?.venues?.[0]?.name;
11
12   const slug = `${evento.name.toLowerCase()
13     .replace(/\s+/g, "-")
14     .replace(/^[^w-]+/g, "")}-${evento.id}`;
15
16   return (
17     <Link href={`../evento/${slug}`} className={styles.linkLimpo}>
18       <div className={styles.card}>
19         {image && <img src={image} alt={evento.name} className={styles.imagem} />}
20         <h3 className={styles.titulo}>{evento.name}</h3>
21         <p className={styles.texto}>{data}</p>
22         <p className={styles.texto}>{local}</p>
23       </div>
24     </Link>
25   );
26 }
27

```

CSS do card (eventoCard.module.css)

```

# eventoCard.module.css > ...
1  .card {
2    border: 1px solid #ccc;
3    border-radius: 8px;
4    width: 250px;
5    padding: 1rem;
6    background-color: #white;
7    box-shadow: 0 2px 6px #rgba(0, 0, 0, 0.1);
8    text-align: center;
9    transition: transform 0.2s ease;
10  }
11
12  .card:hover {
13    transform: scale(1.03);
14  }
15
16  .imagem {
17    width: 100%;
18    height: auto;
19    border-radius: 4px;
20    margin-bottom: 1rem;
21  }
22
23  .titulo {
24    font-size: 1.1rem;
25    margin-bottom: 0.5rem;
26  }
27
28  .texto {
29    font-size: 0.9rem;
30    color: #555;
31  }
32
33  .linklimpo {
34    text-decoration: none;
35    color: inherit;
36  }
37

```

Página Home - exibindo lista de eventos

Arquivo: page.jsx ou Home.jsx

Exibe os eventos atuais, usa o contexto para pegar os dados.

```

8 page.jsx > ...
1 "use client";
2 import { useEffect } from "react";
3 import { useEventos } from "../components/EventosContext";
4 import EventoCard from "../components/EventoCard";
5
6 export default function Home() {
7   const { eventos, buscarEventos, cidadeSelecionada } = useEventos();
8
9   useEffect(() => {
10     if (!cidadeSelecionada) {
11       buscarEventos();
12     }
13   }, [cidadeSelecionada]);
14
15   return (
16     <div style={{ padding: "2rem" }}>
17       <h2 style={{ marginBottom: "1.5rem" }}>
18         {cidadeSelecionada ? "Eventos em ${cidadeSelecionada}" : "Eventos no Brasil"}
19       </h2>
20
21       {eventos.length === 0 ? (
22         <p style={{ fontStyle: "italic", color: "#555" }}>
23           Nenhum evento encontrado no momento.
24         </p>
25       ) : (
26         <div style={{ display: "flex", flexWrap: "wrap", gap: "1.5rem" }}>
27           {eventos.map(evento) => (
28             <EventoCard key={evento.id} evento={evento} />
29           )}
30         </div>
31       )}
32     </div>
33   );
34 }
35

```

Layout geral da aplicação

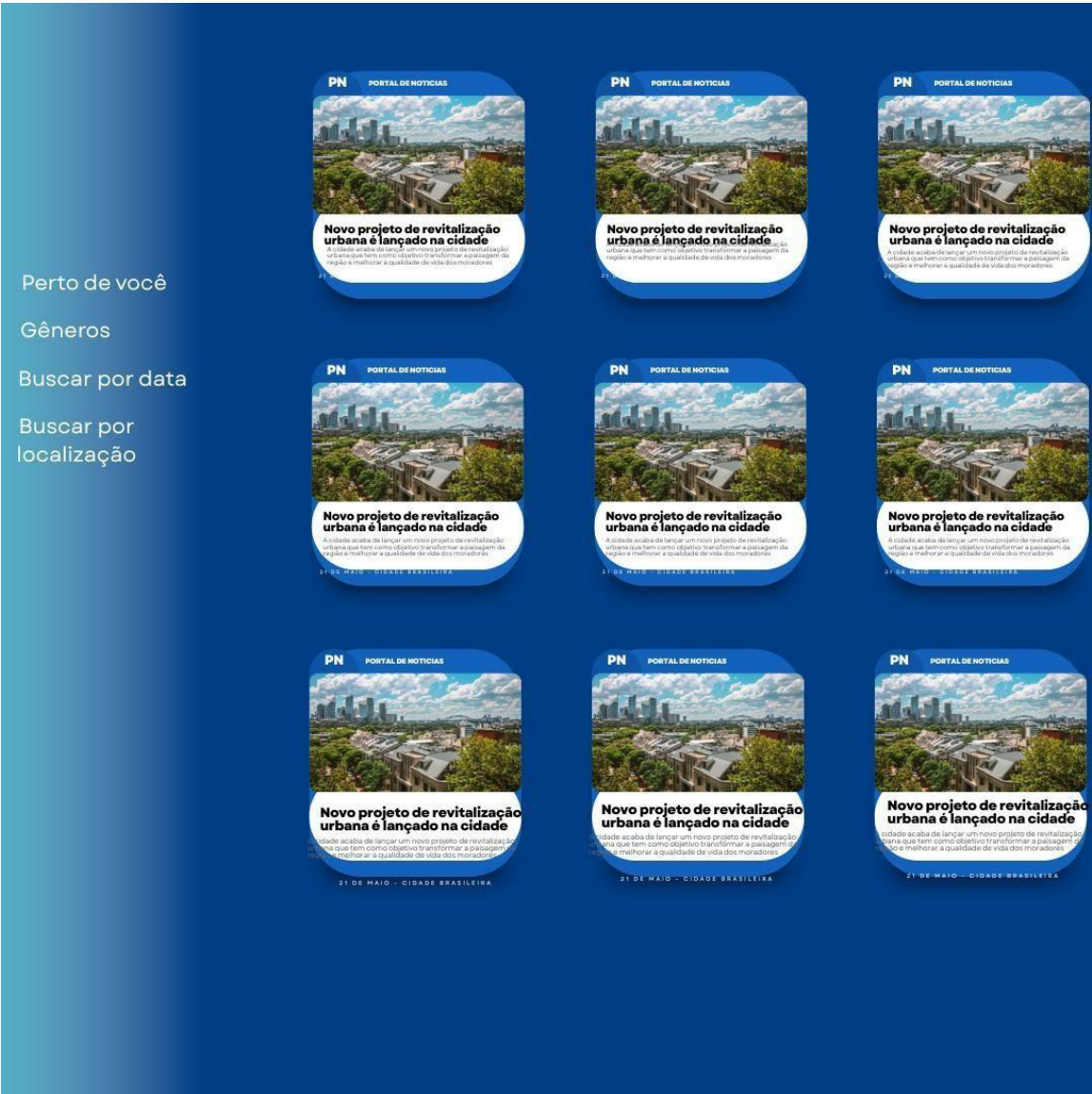
Define o layout padrão, com o menu lateral e o conteúdo principal.

```

RootLayout.jsx > ...
1 import "../globals.css";
2 import Menu from "../components/Menu";
3 import { EventosProvider } from "../components/EventosContext";
4
5 export const metadata = {
6   title: "NextEvents",
7   description: "Plataforma de eventos no Brasil",
8 };
9
10 export default function RootLayout({ children }) {
11   return (
12     <html lang="pt-BR">
13       <body style={{ margin: 0 }}>
14         <EventosProvider>
15           <section style={{ display: "flex", height: "100vh", overflow: "hidden" }}>
16             <nav style={{ width: "250px", backgroundColor: "#0071ce", overflowY: "auto" }}>
17               <Menu />
18             </nav>
19             <main style={{ flex: 1, overflowY: "auto", height: "100vh", padding: "2rem" }}>
20               {children}
21             </main>
22           </section>
23         </EventosProvider>
24       </body>
25     </html>
26   );
27 }
28

```

Protótipo Inicial / Wireframe



Perto de você

Gêneros

Buscar por data

Buscar por
localização



Novo projeto de revitalização urbana é lançado na cidade

A cidade acaba de lançar um novo projeto de revitalização urbana que tem como objetivo transformar a paisagem da região e melhorar a qualidade de vida dos moradores.

Novo projeto de revitalização urbana é lançado na cidade

Exemplo de descrição detalhada:

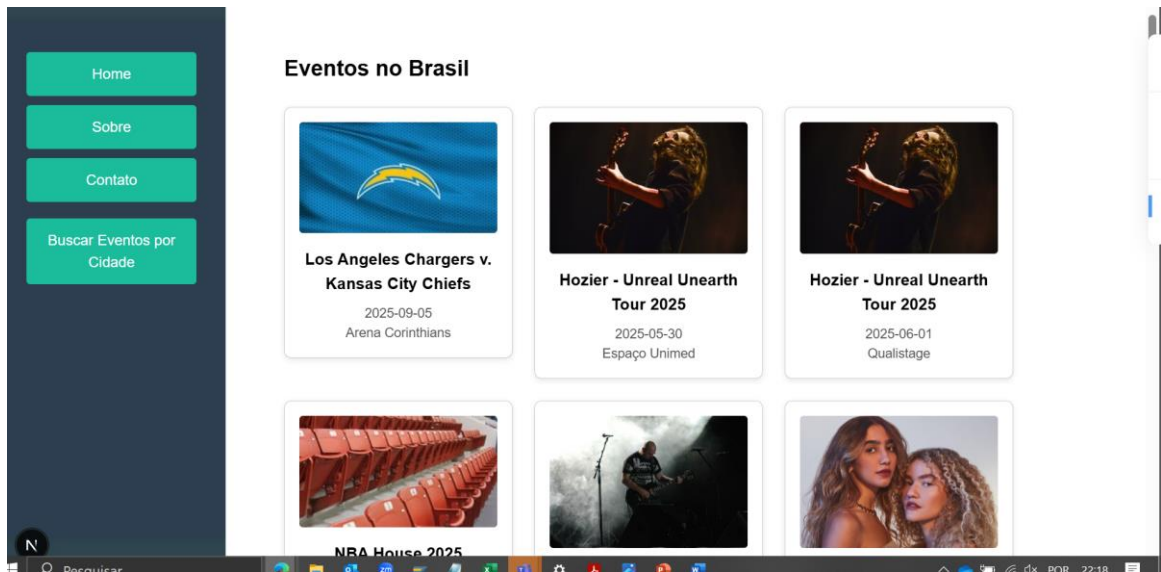
Título do Evento: Festival de Música Alternativa São Paulo 2025
Descrição: Prepare-se para viver uma experiência única no Festival de Música Alternativa São Paulo 2025, um dos eventos mais aguardados do calendário cultural da cidade! Reunindo artistas consagrados e novos talentos da cena alternativa, o festival promete dois dias intensos de música, arte e conexão.
O evento contará com mais de 30 atrações divididas em três palcos simultâneos, além de espaços gastronômicos, feira de vinhos, workshops criativos e instalações artísticas interativas. É o lugar ideal para quem busca diversidade sonora, encontros culturais e momentos inesquecíveis com amigos.
Data e horário: 📅 Dias: 15 e 16 de Novembro de 2025
🕒 Horário: das 12h00 às 23h00 (abertura dos portões às 11h)
Localização: 📍 Parque Villa-Lobos - Av. Prof. Fonseca Rodrigues, 2001 - Alto de Pinheiros, São Paulo - SP
O local oferece fácil acesso por transporte público e conta com bicicletário, estacionamento terceirizado e estrutura para receber pessoas com mobilidade reduzida.
Ingressos: disponíveis online a partir de R\$120,00 (mêa-entrada)
Classificação etária: Livre. Menores de 16 anos devem estar acompanhados de um responsável legal.

Exemplo de descrição detalhada:

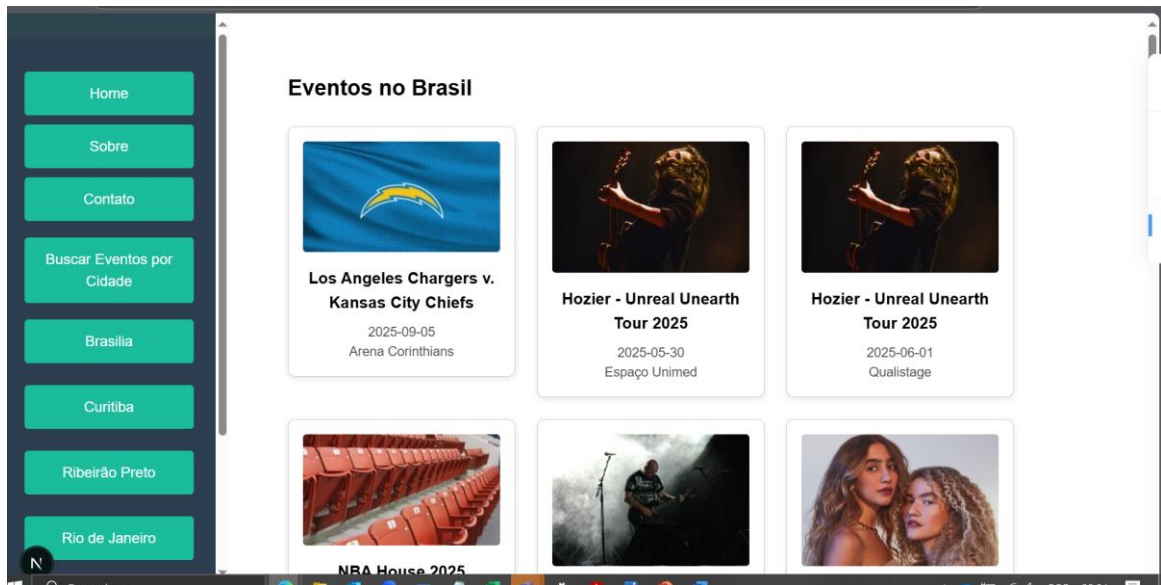
Título do Evento: Festival de Música Alternativa São Paulo 2025
Descrição: Prepare-se para viver uma experiência única no Festival de Música Alternativa São Paulo 2025, um dos eventos mais aguardados do calendário cultural da cidade! Reunindo artistas consagrados e novos talentos da cena alternativa, o festival promete dois dias intensos de música, arte e conexão.
O evento contará com mais de 30 atrações divididas em três palcos simultâneos, além de espaços gastronômicos, feira de vinhos, workshops criativos e instalações artísticas interativas. É o lugar ideal para quem busca diversidade sonora, encontros culturais e momentos inesquecíveis com amigos.
Data e horário: 📅 Dias: 15 e 16 de Novembro de 2025
🕒 Horário: das 12h00 às 23h00 (abertura dos portões às 11h)
Localização: 📍 Parque Villa-Lobos - Av. Prof. Fonseca Rodrigues, 2001 - Alto de Pinheiros, São Paulo - SP
O local oferece fácil acesso por transporte público e conta com bicicletário, estacionamento terceirizado e estrutura para receber pessoas com mobilidade reduzida.
Ingressos: disponíveis online a partir de R\$120,00 (mêa-entrada)
Classificação etária: Livre. Menores de 16 anos devem estar acompanhados de um responsável legal.

Versão Final / Wireframe

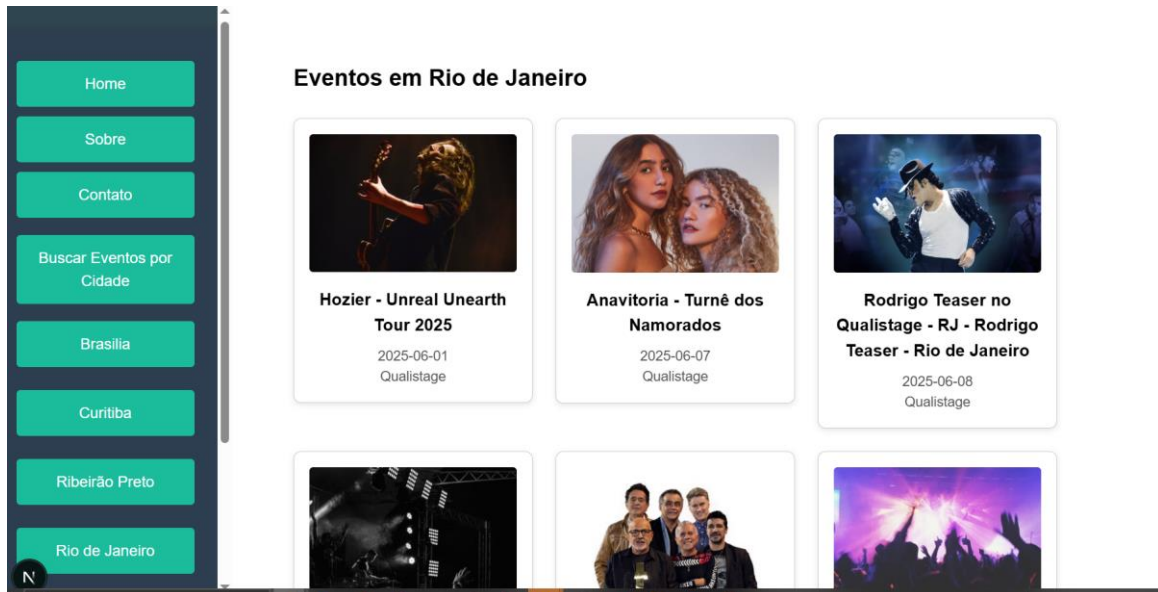
HOME



Ao acionar buscarEventos (botões dinâmicos)



Ao clicar em uma cidade as geradas dinamicamente nos botões :



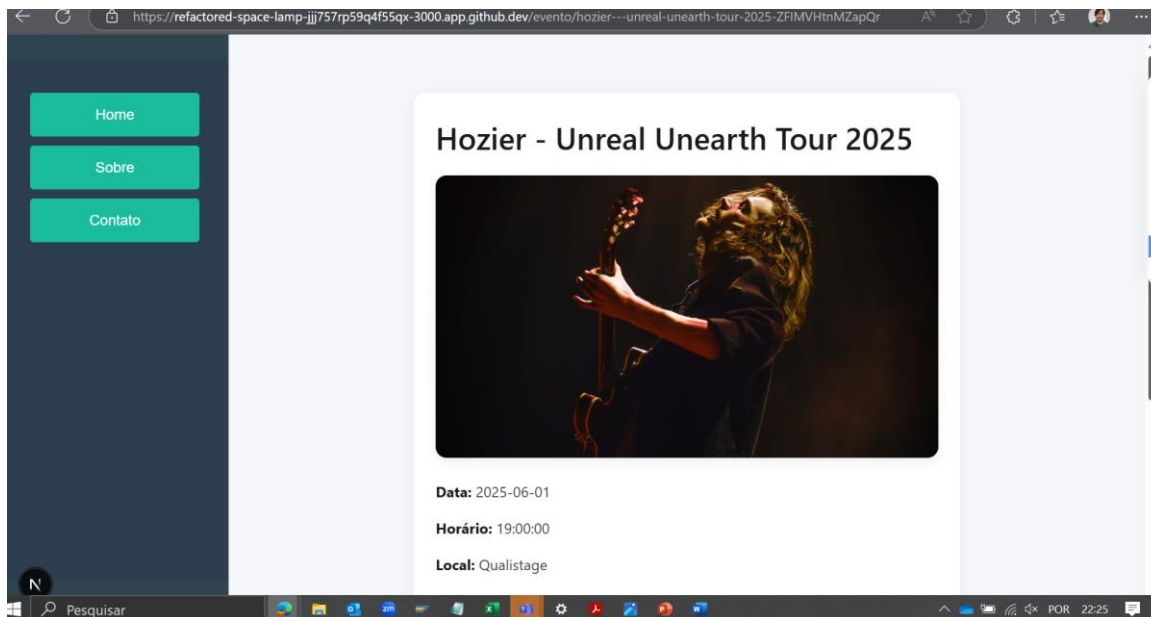
Como os botões de cidade são gerados dinamicamente?

Os botões exibidos ao clicar em "Buscar Eventos por Cidade" são gerados dinamicamente a partir da resposta da API, e apenas para cidades que realmente possuem eventos. Isso evita mostrar cidades irrelevantes ou vazias.

🔍 Lógica no código:

```
const eventos = data._embedded?.events || [];  
  
const cidades = new Set(  
  eventos  
    .map((e) => e._embedded?.venues?.[0]?.city?.name)  
    .filter(Boolean) // remove nulos/undefined  
);  
  
const cidadesUnicas = Array.from(cidades).sort();
```

Ao clicar em um card (redireciona pra page de Eventos)



Ao clicar no botao Comprar ingresso :

Descrição: Sem descrição disponível.

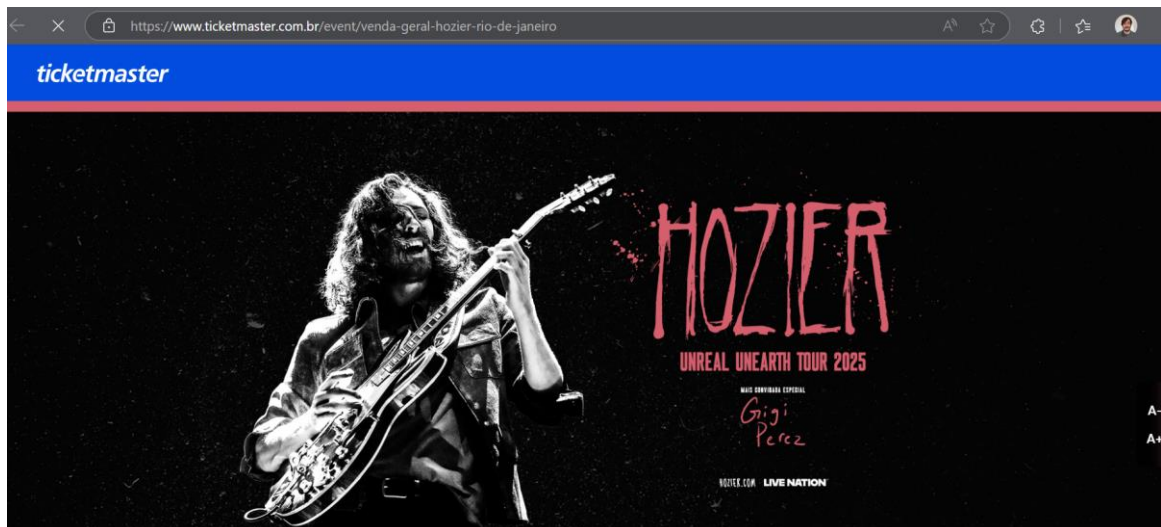
Organizador: Não informado

Categoria: Music

Vendas: Início 1/1/2025, 3:00:00 AM

Comprar Ingressos

Leva ate a pagina de compra da ticketmaster:



Conclusão: Aprendizados Adquiridos

Durante o desenvolvimento deste projeto, foram adquiridos diversos aprendizados importantes no processo de construção de aplicações web modernas. Entre os principais destaques:

Aprendizados Técnicos:

- **Organização do código com componentes funcionais em React e Next.js**, facilitando manutenção e reuso.
- **Criação e uso de Context API** para compartilhar estados e funções globalmente na aplicação.
- **Uso de rotas dinâmicas no Next.js** com Link e slugs amigáveis.
- **Integração com APIs públicas** (como a da Ticketmaster), lidando com estruturas JSON complexas.
- **Estilização modular com CSS Modules**, garantindo encapsulamento de estilo por componente.
- **Aplicação de boas práticas de responsividade e usabilidade.**

Aprendizados de Projeto:

- **Pensamento centrado no usuário** para facilitar a navegação e filtragem dos eventos.
- **Separação clara entre lógica, estrutura e estilo**, respeitando o padrão de arquitetura de componentes.
- **Gerenciamento eficiente de estado**, reduzindo requisições desnecessárias e melhorando a performance.

Essa experiência mostrou na prática como **desenvolver uma aplicação real com foco em interatividade, performance e organização de código** — habilidades altamente valorizadas no mercado atual de desenvolvimento web.

