



ENSINO MÉDIO INTEGRADO - INFORMÁTICA

Disciplina de Desenvolvimento Web

Avaliação B - 2º Bimestre

Gil Eduardo de Andrade

INSTRUÇÕES PARA RESOLUÇÃO DA AVALIAÇÃO

1. A prova é resolvida de forma individual ou em dupla, não havendo troca de informações entre os alunos, nem mesmo troca de material como cadernos, livros ou pen drive.
2. Ao final da prova o aluno deve enviar apresentar a aplicação funcionando ao professor.
3. O desenvolvimento das aplicações deve usar como base a aplicação disponibilizada neste link <https://github.com/Instituto-Federal-PR/vite-react-avalicao> (porta 12007), execução via comando “`docker compose up`”.
4. A estrutura de diretórios e arquivos do projeto deve seguir o padrão já definido para a aplicação disponibilizada no repositório apresentado anteriormente.
5. Todos os dados de músicas devem ser carregados do arquivo JSON (src/data/songs.json), incluso na aplicação disponibilizada no repositório apresentado anteriormente.
6. As seguinte funcionalidades devem estar disponíveis na aplicação:
 - a. **(1.0) Navegação** entre as telas de Home (lista todas as músicas - formato de “Card” com imagem e descrição da música - definidas no arquivo songs.json), Favoritos (lista todas as músicas marcadas como favoritas - formato de “Card” com imagem e descrição da música), Álbuns (lista de todos os álbuns - formato de “Card” - criados pelo usuário, para agrupar um conjunto de músicas) e Detalhes do Álbum, que apresenta a lista e músicas vinculadas ao álbum;

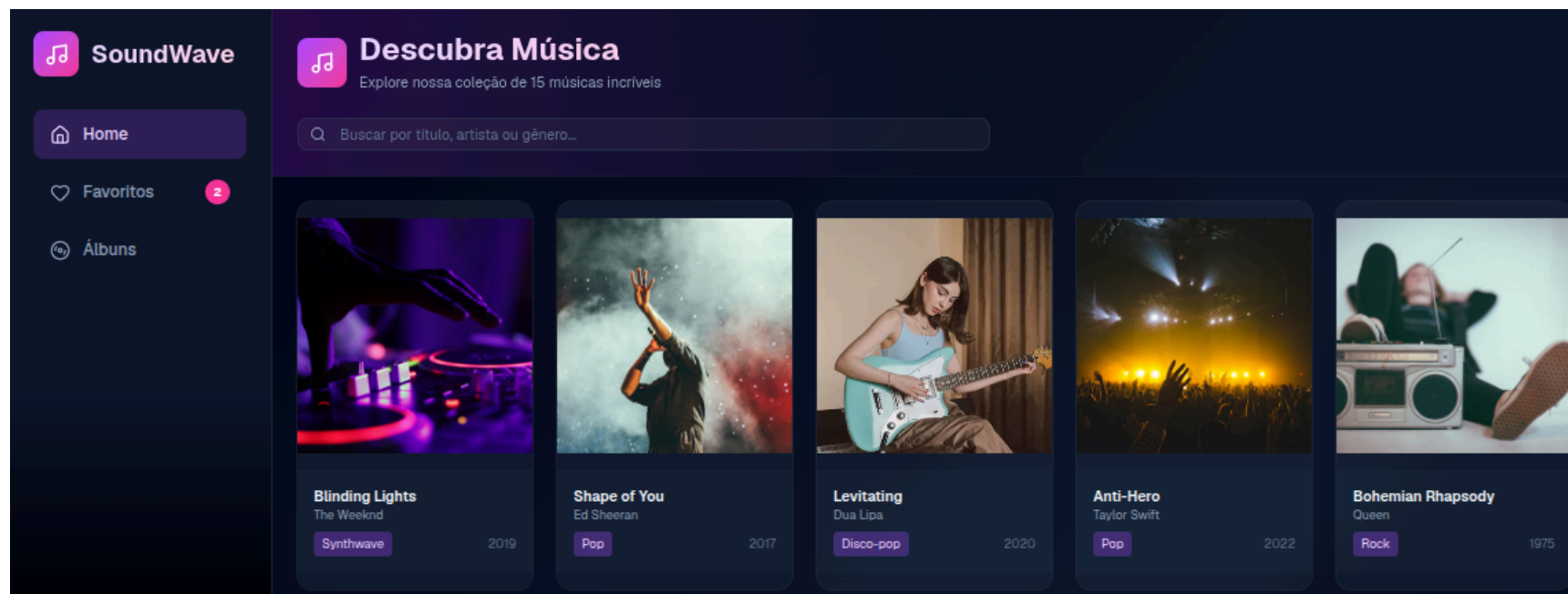
- b. **(1.0) Adicionar aos Favoritos** - permite que usuário adicione um conjunto de músicas a sua pasta de favoritos (tela “Home”) - essa lista deve ser exibida na tela de “Favoritos”. Além disso, um marcador de quantidade de músicas adicionadas aos favoritos deve ser atualizado no menu lateral, ao lado do item “Favoritos”;
- c. **(1.0) Listagem das Músicas Favoritas** - apresenta a lista de músicas - formato de “Card” com imagem e descrição da música - que foram definidas como favoritas. Também permite que as músicas sejam removidas da lista de favoritas, quando isso ocorre, automaticamente a lista deve ser atualizada, com a música removida deixando de ser exibida.
- d. **(0.5) Barra para Busca de Músicas** - as telas de “Home” e “Favoritos” devem possuir uma barra de busca para que usuário possa encontrar músicas de acordo com o seu título. Ao digitar as letras nessa caixa, automaticamente as músicas apresentadas devem ser filtradas.
- e. **(0.5) Criação de Álbuns** - na tela de “Álbuns”, além das listagem de todos os álbuns criados pelo usuário, este deve encontrar uma opção (botão ou algo similar) que lhe permita criar novas listas de músicas, para tal ele deve especificar o título do álbum. Essa tela também permite que um usuário remova um álbum que deseja;
- f. **(0.5) Adicionar Músicas aos Álbuns** - na tela de “Home”, onde a lista de músicas disponíveis é apresentada, no formato de um “Card”, deve ser disponibilizado ao usuário uma opção (botão) onde o usuário poderá adicionar as músicas que deseja, aos álbuns criados anteriormente.
- g. **(0.5) Apresentação das Músicas de um Álbum** - na tela de “Álbuns” o usuário poderá selecionar um “álbum” específico para visualizar a lista de músicas contidas no mesmo. Ao listar as músicas (tela de “Detalhes do Álbum”) o usuário terá a opção de remover uma música do Álbum que foi aberto.

A aplicação React deve conter, ao menos:

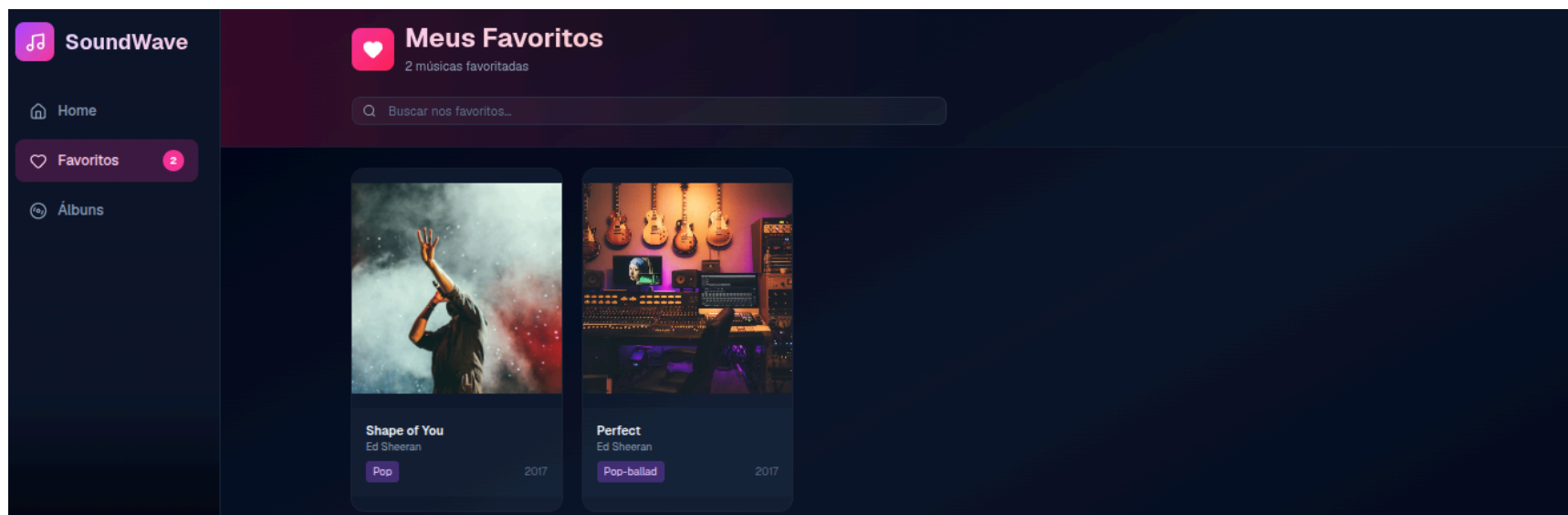
- **Components:** NavBar.tsx, SideBar.tsx, SearchBar, AlbumCard.tsx e MusicCard.tsx;
- **Pages:** Home.tsx, Favorites.tsx, Albums.tsx e AlbumDetail.tsx
- **Context:** MusicContext.ts;
- **Hooks:** useMusic.ts;

Serviço de Streaming de Áudio (SoundWave)

Tela de Home



Tela de Favoritos



Tela de Álbuns

