

Documentação do Projeto `DIST-SUPL.V2`

Visão Geral

Este projeto é uma aplicação **React Native** desenvolvida com o framework **Expo**, estruturada com base em boas práticas de organização de código. Ele utiliza **TypeScript** como linguagem principal e é dividido em várias pastas para facilitar a manutenção e escalabilidade.

Estrutura do Projeto

Diretório Principal

A raiz do projeto contém arquivos essenciais para o funcionamento e configuração da aplicação:

- `app.json`: Configurações gerais do Expo.
- `App.tsx`: Arquivo principal que inicializa a aplicação.
- `firebase.js`: Configuração e inicialização do Firebase.
- `tsconfig.json`: Configurações do TypeScript.
- `eas.json`: Configuração do **Expo Application Services (EAS)**.
- `babel.config.js`: Configuração do Babel para transpilar o código.

Estrutura de Pastas

app/

Contém as rotas da aplicação organizadas em abas (tabs) e categorias, usando o padrão de estruturação de rotas do **Expo Router**.

- `tabs/`
 - `categories/`
 - `_layout.tsx`: Define o layout padrão para a rota de categorias.
 - `[id].tsx`: Rota dinâmica para exibir detalhes de uma categoria.
 - `list.tsx`: Página com a lista de categorias.
 - `menu/`
 - `_layout.tsx`: Layout base para as rotas do menu.
 - `list.tsx`: Lista de itens no menu.
 - `about.tsx`: Página "Sobre".
 - `home.tsx`: Página inicial.
 - `product/`
 - `_layout.tsx`: Layout para as rotas de produtos.
 - `index.tsx`: Página principal de produtos.

assets/

Diretório para recursos estáticos, como imagens, ícones e outros arquivos multimídia.

components/

Componentes reutilizáveis que podem ser utilizados em várias partes do projeto, garantindo modularidade.

conection/

gerencia a conexão com APIs externas ou banco de dados.

services/

Serviços utilizados pela aplicação, como chamadas a APIs ou integração com o Firebase.

types/

Declarações de tipos TypeScript, garantindo segurança de tipo e melhorando a manutenção do código.

Dependências

Certifique-se de instalar todas as dependências do projeto antes de iniciar o desenvolvimento. Use o comando:

```
bash
Copiar código
npm install
```

Principais Dependências

- **expo:** Framework para desenvolvimento de aplicações React Native.
 - **firebase:** Utilizado para serviços de backend como autenticação e banco de dados.
 - **typescript:** Fornece segurança de tipos para o código.
 - Outras dependências estão listadas no arquivo `package.json`.
-

Configuração e Inicialização

Pré-requisitos

- **Node.js** (versão recomendada: 16 ou superior)
- **npm** ou **yarn**
- **Expo CLI**

Configuração

Clone o repositório:

```
git clone <url-do-repositorio>cd DIST-SUPL.V2
```

Instale as dependências:

```
npm install
```

Configure o Firebase no arquivo `firebase.js`.

Inicialização

Para iniciar a aplicação em modo de desenvolvimento, execute:

Copiar código

npm start

Organização das Rotas

A aplicação utiliza o **Expo Router** para navegação, com as seguintes rotas principais:

- `/categories/`: Lista de categorias e detalhes de cada uma.
 - `/menu/`: Contém subpáginas como **Home**, **About** e **List**.
 - `/product/`: Rotas relacionadas a produtos.
-

Contribuindo

Boas práticas de contribuição

1. Use **TypeScript** em todos os arquivos.
 2. Garanta que os componentes sejam modulares e reutilizáveis.
 3. Sempre execute o comando `npm run lint` antes de abrir um Pull Request.
-

Próximos Passos

1. **Testes automatizados**: Adicionar testes unitários e de integração para aumentar a confiabilidade do código.
2. **Melhoria de documentação**: Detalhar o uso de componentes e serviços específicos.
3. **Otimização de performance**: Implementar otimizações para dispositivos com hardware limitado.