

Documentação do Projeto | e-Dashboard

Introdução

Este projeto é um painel administrativo de e-commerce desenvolvido com **Next.js 15**, **React**, **Tailwind CSS**, **ShadCN** e **Supabase**. Ele permite que usuários gerenciem produtos, pedidos e estatísticas de sua loja virtual.

Tecnologias Utilizadas

- **Next.js 15**: Framework para React com renderização no servidor.
- **React**: Biblioteca para interfaces de usuário.
- **Tailwind CSS**: Framework CSS utilitário.
- **ShadCN**: Componentes acessíveis e estilizados para React.
- **Supabase**: Banco de dados e autenticação baseado em PostgreSQL.

Funcionalidades Principais

- **Autenticação**: Login, cadastro e reset de senha via Supabase Auth.
- **Gestão de Produtos**: Visualização, criação, edição e remoção de produtos.
- **Gerenciamento de Pedidos**: Visualização e controle de pedidos.
- **Estatísticas**: Gráfico sobre vendas.

Estrutura do Projeto

O projeto segue a estrutura padrão do Next.js 15:

```
.
├── app/
│   ├── dashboard/
│   │   ├── [rotas]/
│   │   ├── layout.tsx
│   │   └── page.tsx
│   └── login/
│       ├── [rotas]/
│       └── page.tsx
├── components/
├── lib/
├── public/
└── README.md
```

- **app/**: Contém as páginas do projeto.
- **components/**: Componentes reutilizáveis.
- **lib/**: Funções auxiliares.
- **public/**: Arquivos estáticos.

Estrutura do Banco de Dados

Tabela `stores`

Coluna	Tipo	Descrição
<code>id</code>	UUID	Identificador único da loja
<code>user_id</code>	UUID	ID do usuário dono da loja
<code>name</code>	TEXT	Nome da loja
<code>created_at</code>	TIMESTAMP	Data de criação da loja

Tabela `products`

Coluna	Tipo	Descrição
<code>id</code>	UUID	Identificador único do produto
<code>store_id</code>	UUID	Referência à loja do produto
<code>name</code>	TEXT	Nome do produto
<code>description</code>	TEXT	Descrição do produto
<code>buy_price</code>	NUMERIC	Preço de custo do produto
<code>retail_price</code>	NUMERIC	Preço de venda do produto
<code>category</code>	TEXT	Categoria do produto
<code>created_at</code>	TIMESTAMP	Data de criação do produto

Tabela `orders`

Coluna	Tipo	Descrição
<code>id</code>	UUID	Identificador único do pedido
<code>store_id</code>	UUID	Referência à loja do pedido
<code>product_id</code>	UUID	ID do produto comprado
<code>customer_email</code>	TEXT	E-mail do cliente
<code>quantity</code>	INTEGER	Quantidade adquirida
<code>total_price</code>	NUMERIC	Valor total da compra
<code>status</code>	TEXT	Status do pedido (ex: Pending)
<code>created_at</code>	TIMESTAMP	Data de criação do pedido

View `dynamic_orders`

Essa view ajusta dinamicamente a data de criação dos pedidos com base em uma data de referência. É usada para garantir que haja dados disponíveis para visualização, independentemente da data de acesso ao dashboard.

Coluna	Tipo	Descrição
id	UUID	ID do pedido
product_id	UUID	ID do produto
customer_email	TEXT	E-mail do cliente
quantity	INTEGER	Quantidade
total_price	NUMERIC	Valor total
status	TEXT	Status do pedido
store_id	UUID	ID da loja
created_at	TIMESTAMP	Data ajustada com base na <code>date_reference</code>

Configuração e Instalação

1. Clone o repositório:

```
git clone https://github.com/teixeiralc/e-comm-dashboard.git
```

2. Instale as dependências:

```
cd e-comm-dashboard
npm install
```

3. Crie um arquivo `.env.local` e configure as variáveis de ambiente contidas no arquivo `env.example`.
4. Inicie o servidor de desenvolvimento:

```
npm run dev
```

Contato

Caso tenha alguma dúvida ou sugestão, entre em contato pelo e-mail lucascteixeira0@gmail.com.