

Trabalho Final Frameworks e Banco de Dados

Grupo: Clara Baida, Eli Jorge, Lívia Capilé e Letícia de Jesus

Tema: Mercadinho

Links Repositórios:

<https://github.com/clarabaida2008/projeto-backend>

<https://github.com/clarabaida2008/projeto-frontend>

Visão Geral

O sistema Mercadinho é uma aplicação web de gerenciamento para um pequeno mercado, seus usuários são unicamente os funcionários do mercado que podem cadastrar, editar, excluir e atualizar produtos, funcionários, fornecedores e vendas cadastrados no sistema, além de visualizar um relatório geral de tudo.

Tecnologias Utilizadas

- Frontend: React com TypeScript
- Estilização: CSS
- Roteamento: React Router
- Tipagem: TypeScript interfaces

Componentes Principais

- App.tsx: Componente raiz que configura o roteamento da aplicação em forma de cabeçalho com o React Router;
- App.css: Estilização do cabeçalho de navegação e do css base de todo o projeto, como cores e fonte;
- Home.tsx: Página inicial com relatório de produtos e fornecedores;
- Home.css: Estilização da página inicial;
- Fornecedor.tsx: CRUD de fornecedores com fetch para ligar ao back;
- Funcionario.tsx: CRUD de funcionários com fetch para ligar ao back;
- Produto.tsx: CRUD de produtos com fetch para ligar ao back;
- Vendas.tsx: CRUD de vendas com fetch para ligar ao back;
- Fornecedor/Funcionario/Produto/Vendas.css: Estilização dos CRUDs;

- Index.ts: Conexão do frontend como backend e o banco de dados, ajuda a cadastrar, deletar e editar os dados no banco.

Funcionalidades

Navegação

- Barra de navegação superior com links para todas as seções
- Layout responsivo que se adapta a dispositivos móveis

Fornecedores

- Cadastro de novos fornecedores (ID, Nome, CNPJ, Cidade)
- Listagem de fornecedores cadastrados
- Edição e exclusão de fornecedores
- Validação de campos (tipos de dados, campos obrigatórios)

Funcionários

- Cadastro de novos funcionários (ID, Nome, Função, CPF)
- Listagem de funcionários cadastrados
- Edição e exclusão de funcionários
- Validação de campos (tipos de dados, campos obrigatórios)

Produtos

- Cadastro de novos produtos (ID, Nome, Preço, Categoria, Fornecedor)
- Listagem de produtos cadastrados
- Edição e exclusão de produtos
- Validação de campos (tipos de dados, campos obrigatórios)
- Relacionamento com fornecedores através de dropdown

Vendas

- Cadastro de novas vendas

- Listagem de vendas cadastradas
- Edição e exclusão de vendas
- Validação de campos (tipos de dados, campos obrigatórios)

Home

- Exibe um relatório consolidado de produtos e seus fornecedores
- Mostra informações como preço, CNPJ do fornecedor e telefone

Estilização

- Design limpo e moderno com paleta de cores verde
- Layout responsivo usando flexbox
- Componentes estilizados individualmente em arquivos CSS separados
- Mensagens de erro/sucesso com cores diferenciadas

Fluxo de Dados

- Todas as operações CRUD são feitas através de chamadas a uma API REST local (<http://localhost:8000>)
- Os dados são armazenados em estado local usando o hook `useState`
- Efeitos colaterais são gerenciados com `useEffect`

Validações

- Todos os formulários possuem validação de campos obrigatórios
- Verificação de tipos de dados (números onde necessário)
- Validação de formatos (nomes não podem conter números)
- Mensagens de erro claras para o usuário

Como Executar

- Certifique-se que o backend está rodando em <http://localhost:8000>
- Instale as dependências com `npm install`

- Ligue o Laragon
- Instale o banco de dados no seu computador pelo arquivo:
https://drive.google.com/drive/folders/1s8sDzVMpT_LiXOr3VsEHZ3b1dsdR5lpx?usp=sharing
- Inicie a aplicação com `npm run dev`
- Abra o link disponibilizado no terminal do frontend