

**EEEP JEOVÁ COSTA LIMA**

**DISCIPLINA DE MOBILE**

**EMYLLI HAIKA SANTIAGO TORRES(Nº: 08)**

**SABRYNA SILVA LIMA (Nº: 33)**



**PROFESSOR (ES):**

**EDUARDO ALMEIDA E RENAN MOREIRA**

**RUSSAS, CE**

**JUNHO, 2025**

# Documentação do Projeto Mobile

## 1. Informações Iniciais

**Nome do Projeto:**  CDEM – Controle de Estoque de Móveis

**Turma:**  Sistemas 3

**Professor** : Eduardo

## 2. Visão Geral do Projeto

O projeto é um sistema de controle de estoque de móveis, que permite gerenciar móveis e fornecedores através de um aplicativo desenvolvido no FlutterFlow, com backend em Laravel e banco de dados Supabase. Todas as funcionalidades estão integradas via chamadas HTTP.

**Problema que Resolve / Necessidade que Atende:**

Facilita o controle de estoque em empresas do ramo moveleiro, eliminando processos manuais, e garantindo atualização em tempo real de itens e fornecedores.

**Público-Alvo:**

* Empresas e lojas que comercializam móveis e precisam de uma solução prática para controle de estoque e gestão de fornecedores.

## 3. Funcionalidades Implementadas

**Funcionalidades atuais:**

* Tela de Login
* Tela de Cadastro
* Lista de Móveis (CRUD completo)
* Lista de Fornecedores (CRUD completo)
* Integração via API com backend Laravel
* Banco Supabase como base de dados
* Chamadas HTTP para ações de CRUD

## 4. Como Executar o Projeto

**O que preciso fazer para executar seu projeto ?**

**4.1. FlutterFlow (Frontend)**

1. Acesse o projeto no FlutterFlow:

https://app.flutterflow.io/project/projeto-emilly-e-sabryna-gywpym

1. Telas disponíveis:

* Login
* Cadastro
* Lista de Móveis
* Lista de Fornecedores

**4.2. API Backend**

## Clonar o repositório

## Executar composer install

## Configurar arquivo .env com dados do Supabase

## Executar php artisan migrate

## Executar php artisan serve

* Exposição com Ngrok: ngrok http 8000

## 5. Estrutura do Banco de Dados (Supabase)

**Tabelas Criadas:**

Tabela: móveis

## id (Integer)

## nome (String)

## estoque (Integer)

## preco (Float)

## Tabela: fornecedores

## id (Integer)

## nome (String)

## email (String)

## telefone (String)

## endereco (String)

## 6. Integração App + API

A integração é feita por meio de requisições HTTP entre o FlutterFlow e a API Laravel. O banco de dados Supabase é utilizado como armazenamento dos dados e está conectado ao backend.

**Endpoints Utilizados:**

Móveis:

* GET /api/moveis
* POST /api/moveis
* GET /api/moveis/{id}
* PUT /api/moveis/{id}
* DELETE /api/moveis/{id}

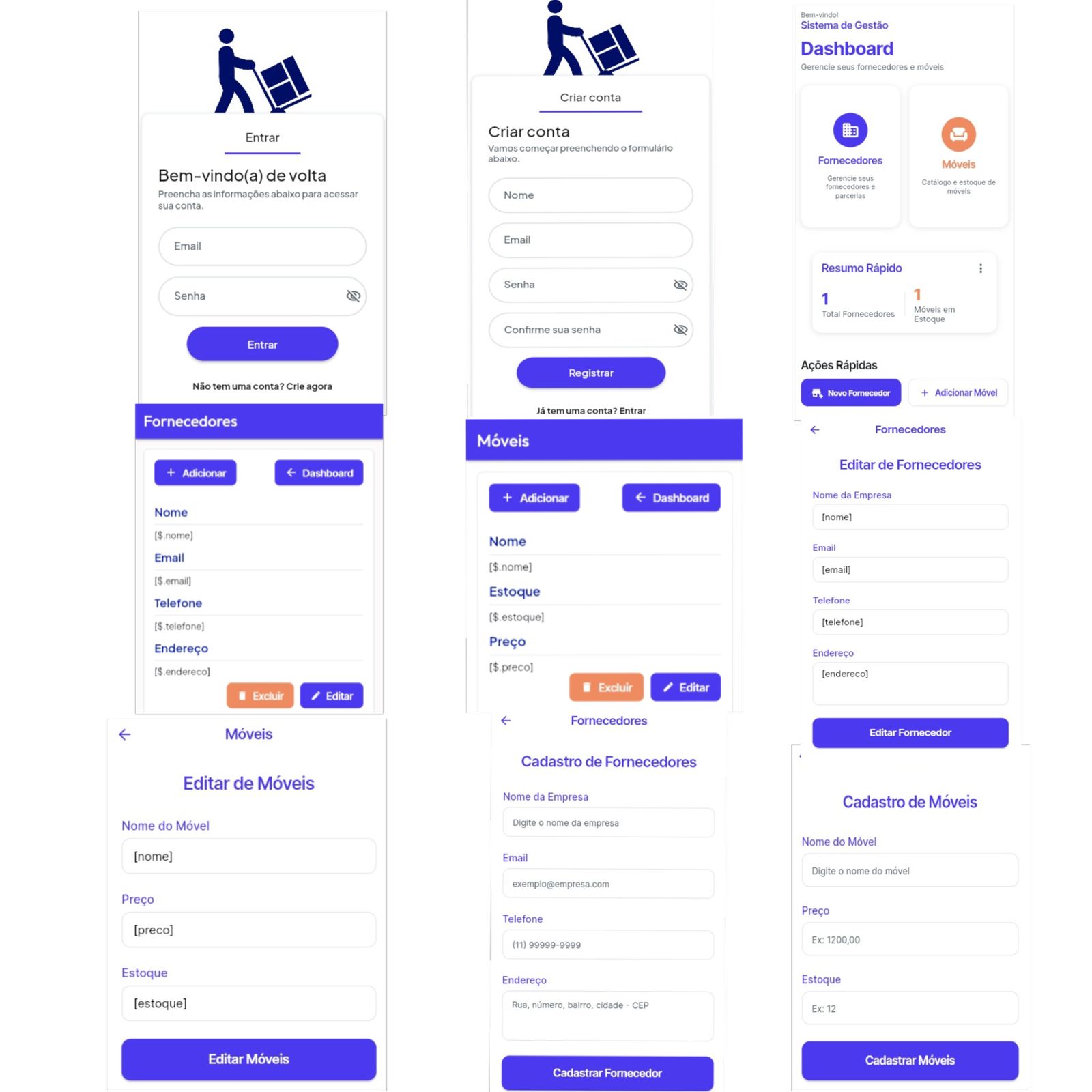
Fornecedores:

* GET /api/fornecedores
* POST /api/fornecedores
* GET /api/fornecedores/{id}
* PUT /api/fornecedores/{id}
* DELETE /api/fornecedores/{id}

**Segurança:**

As requisições utilizam HTTPS, e o acesso pode ser protegido com autenticação via token (JWT), caso implementado.

1. **Prints do App:**



1. **Repositório GitHub:**

https://github.com/aikasantiago/crud-moveis.git