Bom começo

**1.0 Descrição do projeto**

O projeto consiste na criação de uma aplicação intuitiva e eficiente, projetada especificamente para atender às necessidades de pequenos empreendedores familiares. O objetivo principal é oferecer uma solução acessível para o controle de estoque, permitindo que esses negócios administrem seus recursos de forma simplificada e com o mínimo de dependência de ferramentas externas.

|  |  |
| --- | --- |
| **Tecnologia** | **Função no Projeto** |
| **Django** | Framework Python usado tanto no backend quanto no frontend. No backend, gerencia lógica do sistema, autenticação, controle de estoque, relatórios e integração com o banco. No frontend, serve páginas web responsivas e formulários de cadastro/movimentação de produtos. |
| **Supabase** | Banco de dados em nuvem (PostgreSQL) que armazena dados do sistema: produtos, usuários, movimentações, relatórios. Também pode fornecer autenticação e armazenamento de arquivos se necessário. |
| **Render** | Plataforma de hospedagem que vai executar o backend Django e servir o frontend web, garantindo disponibilidade e escalabilidade. |
| **Python** | Linguagem principal do projeto, usada para programação do backend, regras de negócio e scripts auxiliares. |
| **HTML/CSS/JS** | Linguagens usadas no frontend do Django para criar interfaces web, formulários e relatórios. |

**Requisitos Funcionais (RF)**

**RF01 – Login e autenticação**

* O sistema deverá permitir que o usuário entre no sistema informando seu ID cadastrado e senha.

**RF02 – Recuperação de senha**

* O sistema deverá permitir a recuperação de senha pelo email cadastrado.

**RF03 – Gestão de produtos (CRUD)**

* O sistema deverá permitir cadastrar produtos com nome, descrição, preço de custo, preço de venda e quantidade inicial e ID.
* O sistema deverá permitir editar, excluir e consultar produtos em estoque.

**RF04 – Movimentações de estoque**

* O sistema deverá permitir registrar entradas (compras) e saídas (vendas) de produtos.

**RF05 – Relatórios**

* O sistema deverá gerar relatórios de vendas, compras, kardex e sugestões de compras.

**RF06 – Gestão de usuários**

* O sistema deverá permitir cadastrar usuários com nome, e-mail, senha, permissões e ID.
* O sistema deve permitir editar, excluir, e consultar usuários

**RF07 – Gestão de clientes**

* O sistema deverá permitir cadastrar clientes com nome, contato, e ID.
* O sistema deve permitir editar, excluir, e consultar clientes

**Requisitos Não Funcionais (RNF)**

**RNF01 – Conformidade legal**

* O sistema deverá seguir a LGPD.

**RNF02 – Segurança**

* O sistema deverá estabelecer mecanismos de autenticação para garantir a integridade e privacidade dos dados dos usuários.

**RNF03 – Usabilidade**

* O sistema deverá ser intuitivo e fácil de usar por pessoas com pouca familiaridade tecnológica.
* O tempo médio para cadastrar um produto não deverá exceder 1 minuto.

**RNF04 – Desempenho**

* O sistema deverá suportar até 1000 produtos cadastrados sem degradação de desempenho.
* O sistema deverá permitir acesso simultâneo de múltiplos usuários sem perdas de desempenho.

**RNF05 – Disponibilidade**

* O sistema deverá estar disponível 99% do tempo.

**RNF06 – Manutenibilidade**

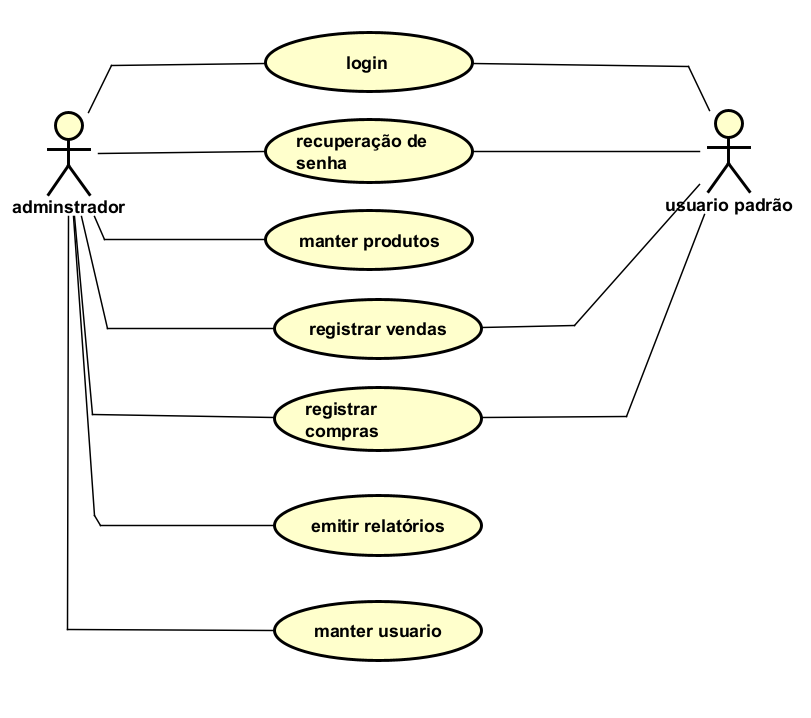
* O código do sistema deverá ser modular e documentado para facilitar futuras alterações ou adição de funcionalidades.

**RNF07 – Hospedagem**

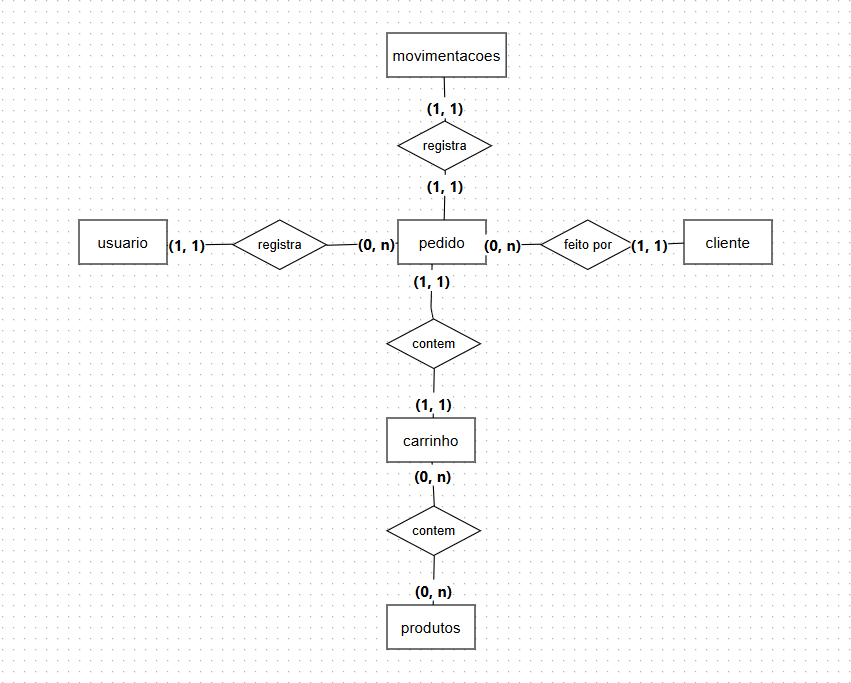
* O sistema deverá ser hospedado em serviços de nuvem.

**Regras de negocio**

Diagrama de caso de uso



Entidade relacionamento



Nomes para o sistema

Bom começo

Prosperar

Jaspy

small storage

smile storage