

1. Front-end

Estrutura:

- O front-end da aplicação é responsável pela interface com o usuário. Ele está localizado no diretório `estoque-app`.

Componentes:

- **HTML/CSS/JavaScript:** Foram para construir a interface do usuário, permitindo a interação com a aplicação.
- **Arquivos e Diretórios:** O diretório `estoque-app` contém os arquivos necessários para a interface do usuário. Eles contém os principais componentes de uma aplicação web, como arquivos HTML, CSS para estilos e JavaScript para interatividade.

2. Back-end

Estrutura:

- O back-end é responsável pela lógica da aplicação e manipulação de dados. Ele é implementado principalmente em Python.

Componentes:

- **`app.py`:** O arquivo principal que executa a aplicação. Este script configura o servidor web (usando Flask), define as rotas e os handlers para as requisições HTTP.
- **`init_db.py`:** Script utilizado para inicializar o banco de dados, criando as tabelas necessárias com base no esquema definido em `schema.sql`.
- **`python.py`:** Um script auxiliar, cuja função específica pode variar, contém funções de suporte utilizadas no projeto.

3. Banco de Dados

Estrutura:

- O banco de dados é usado para armazenar informações persistentes da aplicação. Neste projeto, foi utilizado o SQLite, um banco de dados leve e integrado.

Componentes:

- **`db.sqlite`:** Arquivo do banco de dados SQLite que armazena todas as informações persistentes.
- **`schema.sql`:** Arquivo contendo o esquema do banco de dados, definindo as tabelas e suas colunas. Este arquivo é usado pelo script `init_db.py` para configurar o banco de dados inicialmente.

4. Controle de Versão

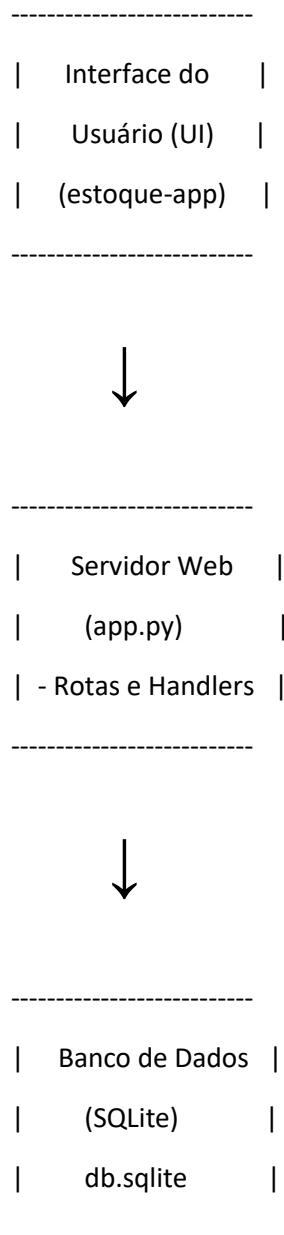
Estrutura:

- O controle de versão é gerenciado pelo Git, permitindo rastrear mudanças no código e colaborar com outros desenvolvedores.

Componentes:

- **.git:** Diretório que contém todos os dados do repositório Git, incluindo histórico de commits, branches e outros metadados necessários para o controle de versão.

Diagrama de Arquitetura



Descrição da Interação entre os Componentes

1. **Usuário Interage com a UI:**
 - O usuário utiliza o navegador para acessar a interface web.
 - As páginas HTML/CSS/JavaScript enviam requisições ao servidor web.
2. **Servidor Web Processa Requisições:**
 - O servidor web, implementado em `app.py`, recebe as requisições e as processa de acordo com as rotas definidas.
 - Dependendo da rota, o servidor pode acessar o banco de dados para obter ou modificar dados.
3. **Banco de Dados Armazena Dados:**
 - As operações de leitura e escrita no banco de dados são realizadas utilizando a biblioteca SQLite em `app.py`.
 - O banco de dados armazena informações persistentes que são necessárias para o funcionamento da aplicação.
4. **Controle de Versão:**
 - Todas as mudanças no código são rastreadas utilizando o Git, permitindo gerenciar versões diferentes do código e colaborar eficientemente com outros desenvolvedores.

Conclusão

A arquitetura do sistema desenvolvido é uma arquitetura típica de uma aplicação web simples, onde o front-end e o back-end são bem separados, o banco de dados é usado para persistência de dados e o controle de versão garante a integridade e a colaboração no desenvolvimento do projeto.