



Documentação do Projeto

E-commerce

Desenvolvedor: Ismael nunes dos santos

Data da Geração: 08/06/2025

1. 📌 Visão Geral do Projeto

Este projeto é uma plataforma de e-commerce completa, desenvolvida com **Flask** no backend e tecnologias web no frontend. A aplicação permite que usuários comprem produtos, gerenciem pedidos e acompanhem seu histórico. Conta com um painel administrativo robusto para controle total do sistema.

Funcionalidades principais:

- Cadastro e login de usuários
 - Catálogo de produtos com paginação
 - Carrinho de compras persistente por sessão
 - Checkout e histórico de pedidos
 - Painel administrativo completo
 - Templates responsivos com Jinja2
-

2. 🧱 Stack Tecnológico

Camada	Tecnologias
Backend	Python, Flask, SQLite
Frontend	HTML5, CSS3, JavaScript, Bootstrap
Templates	Jinja2
Auth	Sistema de autenticação com hash de senhas (Werkzeug)

3. 📁 Estrutura de Arquivos

```
ecommerce/  
├─ app.py                # Aplicação principal Flask  
├─ ecommerce.db          # Banco de dados SQLite  
├─ static/  
│   ├─ css/              # Folhas de estilo  
│   └─ uploads/          # Imagens do sistema  
├─ templates/  
│   ├─ admin/            # Templates administrativos  
│   ├─ base.html         # Template base  
│   ├─ index.html        # Página inicial  
│   └─ ... outros templates  
├─ README.md             # Documentação básica  
└─ requirements.txt      # Dependências do projeto
```

4. ⚙️ Como Rodar o Projeto Localmente

Siga os passos abaixo para executar o projeto na sua máquina:

4.1 Pré-requisitos

- Python 3.8 ou superior
- pip (gerenciador de pacotes do Python)
- Git (opcional, para clonar o repositório)

4.2 Passo a passo

Clone o repositório (ou copie os arquivos):

```
git clone https://github.com/seuusuario/ecommerce-flask.git
cd ecommerce-flask
```

1. **Crie um ambiente virtual (opcional, mas recomendado):**

```
python -m venv venv

source venv/bin/activate      # Linux/macOS
venv\Scripts\activate        # Windows
```

2. **Instale as dependências:**

```
pip install -r requirements.txt
```

3. **Execute a aplicação:**

```
python app.py
```

4. **Acesse no navegador:**

```
http://127.0.0.1:5000/
```

5. Funcionalidades Implementadas

Módulo	Descrição
Autenticação	Login e cadastro com verificação de sessão segura
Produtos	CRUD completo com imagens, estoque e categorias
Carrinho	Carrinho persistente por sessão do usuário
Checkout	Fluxo de finalização de compra com confirmação
Admin	Painel com dashboard, gestão de produtos, pedidos e usuários
Responsivo	Interface adaptável para dispositivos móveis e desktop

6. Desafios e Soluções

Desafio	Solução Implementada
Conflito de rotas	Rotas com prefixo <code>/admin</code> para separar painel administrativo
Templates faltantes	Criação de templates separados por contexto (admin, usuário, geral)
Upload de imagens	Verificação do tipo de arquivo e segurança de upload
Sessão do carrinho	Uso da <code>session</code> do Flask para armazenar dados temporários de carrinho

7. Notas Finais

- A aplicação pode ser facilmente migrada para PostgreSQL ou MySQL com pequenas alterações no backend.
- Para produção, recomenda-se utilizar **Gunicorn** ou **uWSGI** com um servidor como **Nginx**.