Documentação Técnica – Meganium Sales Analysis

# 1. Visão Geral

O Meganium Sales Analysis é um projeto destinado à consolidação e análise de dados de vendas extraídos de múltiplos marketplaces. Seu principal objetivo é transformar os dados brutos em informações valiosas para a fabricante, com destaque para a identificação de produtos mais vendidos e a otimização logística, visando aprimorar a tomada de decisões e o planejamento estratégico.

# 2. Arquitetura do Sistema

## 2.1 Diagrama de Arquitetura

A arquitetura do sistema é dividida em quatro componentes principais: a coleta de dados, o processamento, o armazenamento e a análise visual dos dados. A seguir está o diagrama que ilustra a arquitetura do sistema.

## Data Warehouse: Cruzamento de Dados e Inteligência.

## 2.2 Descrição da Arquitetura

• Dados Brutos (CSVs): Dados provenientes de diferentes marketplaces, armazenados no diretório data/raw\_data/.  
• Processamento (Pandas): Os dados são carregados, limpos e transformados para um formato uniforme usando a biblioteca Pandas.  
• Armazenamento: Dados processados e consolidados são armazenados em arquivos CSV dentro do diretório data/processed\_data/.  
• Análise e Visualização: Utilização das bibliotecas Seaborn e Matplotlib para gerar gráficos e insights a partir dos dados processados.

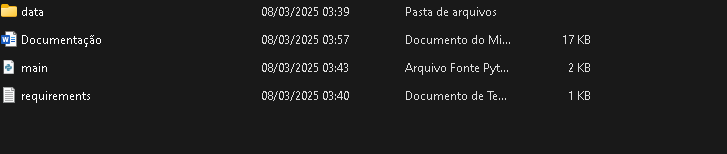
# 3. Tecnologias Utilizadas

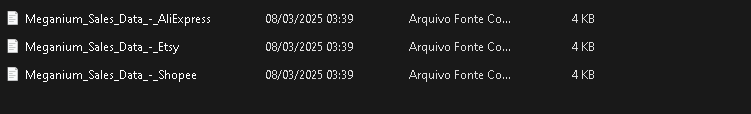
As principais tecnologias utilizadas no projeto são as seguintes:  
  
• Python 3.9+: Linguagem de programação principal.  
• Pandas: Biblioteca para manipulação e análise de dados.  
• Seaborn e Matplotlib: Bibliotecas para visualização gráfica de dados.  
• Virtualenv (venv): Ferramenta para o gerenciamento de ambientes virtuais.

# 4. Estrutura de Diretórios

A estrutura de diretórios foi organizada da seguinte forma:

Meganium-Sales-Analysis/  
├── data/  
│ ├── raw\_data/ # Dados brutos (CSVs)  
│ ├── processed\_data/ # Dados processados  
├── scripts/  
│ ├── analysis.py # Script principal de análise  
│ ├── data\_processing.py # Script de processamento de dados  
├── requirements.txt # Dependências do projeto  
├── README.md # Documentação do projeto  
├── docs/ # Documentação técnica e relatórios





# 5. Fluxo de Processamento

## 5.1 Leitura e Consolidação dos Dados

Os dados brutos, em formato CSV, são carregados do diretório data/raw\_data/ e unificados em um único DataFrame utilizando Pandas.

## 5.2 Limpeza e Transformação

Durante o processo de transformação, as seguintes etapas são executadas:  
  
• Remoção de Duplicatas: Garantir que não existam registros duplicados nos dados.  
• Padronização de Colunas: Ajustar os nomes das colunas para um formato uniforme e compreensível.  
• Conversão de Formatos: Conversão de campos como datas e valores monetários para o formato apropriado.

## 5.3 Salvamento dos Dados Processados

Após a limpeza e transformação, os dados são salvos no arquivo sales\_data\_consolidated.csv no diretório data/processed\_data/.

## 5.4 Análise e Visualização

A análise dos dados é feita por meio de gráficos gerados com Seaborn e Matplotlib, incluindo visualizações como:  
  
• Produtos mais vendidos.  
• Distribuição de vendas por país.  
• Tendências de preços.

# 6. Execução do Projeto

## 6.1 Configuração do Ambiente

Para configurar o ambiente de desenvolvimento, execute os seguintes comandos no terminal:  
  
python -m venv venv  
source venv/bin/activate # Linux/macOS  
venv\Scripts\activate # Windows  
pip install -r requirements.txt

## 6.2 Rodar a Consolidação de Dados

Para rodar o processo de consolidação de dados, execute o script:  
  
python scripts/data\_processing.py

## 6.3 Rodar a Análise e Geração de Gráficos

Para rodar a análise e gerar os gráficos, execute:  
  
python scripts/analysis.py

# 7. Melhorias Futuras

## 7.1 Automação da Coleta de Dados

Automatizar a coleta de dados através de APIs fornecidas pelos marketplaces, permitindo uma atualização contínua e em tempo real dos dados.

## 7.2 Armazenamento em Banco de Dados Relacional

Implementar o uso de um banco de dados relacional como PostgreSQL, melhorando a escalabilidade e a performance no armazenamento dos dados processados.

## 7.3 Painel Interativo com Streamlit

Desenvolver um painel interativo utilizando Streamlit para que os usuários possam visualizar e explorar os dados de forma dinâmica, sem a necessidade de rodar scripts manualmente.

# 8. Conclusão

O Meganium Sales Analysis é um projeto focado na análise e visualização de dados de vendas provenientes de marketplaces. Utilizando ferramentas poderosas como Pandas, Seaborn e Matplotlib, o projeto oferece insights valiosos sobre o comportamento de vendas, ajudando a melhorar a logística e a tomada de decisões estratégicas. As melhorias planejadas, como a automatização da coleta de dados e a integração com banco de dados, irão ampliar a funcionalidade e a escalabilidade do sistema.