

## **Relatório Objetivo do Projeto – Análise de Volatilidade de Criptomoedas**

O propósito do projeto é realizar uma análise completa e automatizada de um dataset com preços e volumes em nível minuto de diversas criptomoedas (Bitcoin, Ethereum, Solana, Near Protocol, Pepe, Dogwifhat, Bonk, XRP, Dogecoin, Shiba Inu e Render) entre 3 e 9 de dezembro de 2025. O objetivo é calcular estatísticas descritivas, volatilidade baseada em retornos percentuais minuto a minuto, volume médio, correlação entre ativos e gerar gráficos interativos, oferecendo uma visão clara do comportamento de preço e risco de cada criptomoeda no período.

A arquitetura é totalmente containerizada com Docker, garantindo que o ambiente seja 100% portátil e reproduzível em qualquer computador sem instalação manual de dependências. O processamento utiliza PySpark em modo local (local[\*]), permitindo explorar todo o poder do Spark mesmo em máquinas pessoais, enquanto a geração de gráficos e cálculos adicionais é feita com bibliotecas Python tradicionais.

Os componentes e ferramentas utilizados são:

- PySpark 3.5.x (leitura, transformação e agregações do dataset)
- Pandas e NumPy (conversões e manipulações complementares)
- Plotly (gráficos interativos de preço, volume e matriz de correlação)
- Docker (containerização)
- Imagem base apache/spark:3.5.3 ou bitnami/spark:3.5.0-debian-12-r0 (ambas testadas e funcionais)
- spark-submit como ponto de entrada
- Arquivos do projeto: analyze\_crypto.py, crypto\_volatility\_dec2025\_FINAL\_2025-12-09.csv, Dockerfile e requirements.txt

Para executar o ambiente em contêiner em qualquer computador:

1. Ter o Docker Desktop instalado.
2. Descompactar a pasta do projeto.
3. Abrir terminal na pasta crypto-analysis-docker.
4. Executar uma única vez: docker build -t crypto-analysis .

5. Executar sempre que quiser rodar a análise: docker run --rm -it crypto-analysis

O processo é automático: o container constrói o ambiente Spark, carrega o CSV, executa toda a análise e exibe estatísticas, rankings e gráficos diretamente no terminal ou salva-os como arquivos HTML quando configurado com volume. Não é necessário instalar Python, Java, Spark ou qualquer outra ferramenta na máquina host.