High-Level Design (HLD) Document Cryptocurrency Volatility Prediction System

**Contents**

1. [System Overview](#_bookmark0) 2
2. [Architecture](#_bookmark1) 2
3. [Data Flow](#_bookmark2) 2
4. [Technology Stack](#_bookmark3) 2
5. [Assumptions](#_bookmark4) 2

# System Overview

The Cryptocurrency Volatility Prediction System is designed to forecast volatility for multiple cryptocurrencies using historical market data. The system processes daily OHLC (open, high, low, close), volume, and market capitalization data, engineers features, trains a machine learning model, and provides predictions through a user interface.

# Architecture

The system follows a modular architecture with the following components:

* + **Data Ingestion**: Loads and validates CSV data containing historical cryptocurrency data.
  + **Data Preprocessing**: Cleans data, engineers features, and normalizes numerical values.
  + **Exploratory Data Analysis (EDA)**: Generates statistical summaries and visualiza- tions to understand data patterns.
  + **Model Training**: Trains a machine learning model to predict volatility.
  + **Model Evaluation**: Assesses model performance using standard metrics.
  + **Deployment**: Deploys the model via a web interface for user interaction.

# Data Flow

* + **Input**: CSV file with columns: date, crypto\_name, open, high, low, close, volume, marketCap.
  + **Processing**: Clean data, engineer features (daily return, rolling volatility, liquidity ratio, moving average), and normalize.
  + **Model Training**: Train a model on time-series data to predict volatility.
  + **Output**: Predictions displayed via a web interface, with evaluation metrics saved.

# Technology Stack

* + **Programming Language**: Python
  + **Libraries**: Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, Streamlit
  + **Model**: RandomForestRegressor
  + **Deployment**: Streamlit for local web interface

# Assumptions

* + The dataset is representative of market conditions from 2013 to 2022.
  + Volatility is calculated as the standard deviation of daily returns over a 7-day win- dow.
  + Time-series splitting is used to respect temporal dependencies.