High-Level Design Document

Objective

Build an AI-powered application to predict the next 30 closing prices of Nifty50 using an LSTM model. The system is containerised and monitored with Prometheus/Grafana, with feedback capabilities for continual learning or user satisfaction tracking.

Design Highlights

- **Model Architecture**: LSTM-based model using a single feature (close price).
- **Feature Selection**: Only the close column is used from the raw dataset.
- **Prediction**: Next 30 timestamps of closing prices.
- **Deployment**: FastAPI for inference, Streamlit for UI, Dockerized environment.
- **Monitoring**: Prometheus exporter for CPU, memory, API, and other system metrics.
- **Feedback Loop**: Users provide satisfaction feedback, which gets emailed to the developer.
- **Data Versioning**: Using DVC to manage and track data pipeline stages.

Design Choices & Rationale

- **LSTM Model**: Chosen for its effectiveness in modelling time-series sequential data.
- **FastAPI**: Lightweight and fast for serving ML models as REST APIs.
- **Streamlit**: Easiest UI layer for non-technical users to interact with ML apps.
- **Docker**: Ensures consistent environments across development and deployment.
- **Prometheus + Grafana**: Industry-standard for system-level and application monitoring.