PHASE-1 PROJECT DOCUMENT SUBMISSION

TOPIC : **CRACKING THE MARKET CODE WITH AI-DRIVEN STOCK PRICE PREDICTION USING TIME SERIES ANALYSIS**

Student Name: GOBINATH P

Register Number: 620123106027

Institution: AVS ENGINNERING COLLEGE, AMMAPET, SALEM.

Department: ELECTRONICS AND COMMUNICATION ENGINEERING

Date Of Submission: 30.04.2025

1. Problem Statement

The stock market is a complex system that fluctuates daily due to various global and local factors. Predicting stock prices accurately is challenging, especially for beginners who lack analytical tools. Most people rely on intuition or news, which often leads to losses. Our goal is to build a simple AI-powered time series analysis model that can study past stock prices and predict future trends. This solution makes stock prediction understandable for students and beginners, bridging the gap between finance and technology.

2. Objectives of the Project

- To collect and understand historical stock price data of selected companies.

- To analyze how stock prices behave over time using basic visualization and statistics.

- To use multiple programming languages (Python,C, C++) to demonstrate concept implementation.

- To build a user-friendly program or script that shows predictions based on selected data.

- To provide insights into how AI can be applied in finance, even with basic tools.

- To help students gain practical experience in time series analysis and machine learning basics.

3.Scope of the Project

- Analyze key features: open price, close price, volume, daily returns, and moving averages.

- Use basic models like Linear Regression, Moving Average, and simple logic in Java/C++.

- Limit the scope to short-term prediction (e.g., next day's price or weekly trend).

- Static dataset only; no real-time trading, deep learning, or advanced financial modeling.

- Focus on model understanding and code development in different languages.

- Compare how Python, Java, and C/C++ can be used in basic AI applications.

4.Data Sources

Source: Yahoo Finance or Kaggle (public dataset)

Format: CSV files with columns like Date, Open, High, Low, Close, Volume

Tools for Download: Python's yfinance library or manual download from Kaggle

Type: Static dataset

Company Examples: TCS, Reliance, Tesla, Apple

5.High-Level Methodology

● **Data Collection:**
Download stock data using yfinance in Python or obtain CSV files from Kaggle. Data from 2020 2023 will be used.

● **Data Cleaning:**
Remove missing values, ensure date formatting is correct, handle duplicate rows. Convert data into usable format across languages.

● **Exploratory Data Analysis (EDA):**
Use graphs (matplotlib/seaborn in Python), observe trends, spikes, and dips. Find weekly and monthly trends using averages.

● **Feature Engineering:**
Create new columns like:
7-day and 30-day Moving Average
Daily percentage change
Volatility using rolling standard deviation

● **Model Building:**
Use basic models like:
Python: Linear Regression (scikit-learn), Moving Average
Java/C++: Implement logic to read file and calculate moving averages
C: Basic file reading and numerical operations

● **Model Evaluation:**
Measure accuracy using:
Mean Absolute Error (MAE)

Root Mean Square Error (RMSE)
Visual inspection: Predicted vs Actual graph

● **Visualization & Interpretation:**
Line plots of actual vs predicted stock prices. Highlight price trend zones, moving averages, and error ranges.

● **Deployment:**
No web app deployment. All outputs will be run and shown in:

Python: Google Colab/Jupyter

Java: Console application using NetBeans

C/C++: Executable programs via Turbo C++ or Code::Blocks

6.Tools and Technologies

Programming Languages: Python, Java, C, C++

Notebook/IDE: Google Colab (Python), NetBeans (Java), Turbo C++, Code::Blocks (C/C++)

Libraries:
Python: pandas, numpy, matplotlib, seaborn, scikit-learn

Java: Apache POI (for CSV reading), basic I/O operations

C/C++: stdio.h, stdlib.h, file operations

Deployment: Not applicable (basic output via console/plots)

7.Team Members and Roles

Gobinath P: The Team Leader. He is responsible for overall coordination and developing the Python model for stock prediction.

Keerthiveraj P: Handles the Java programming part and is also in charge of writing and organizing the project documentation.

Arun S: Takes care of data cleaning tasks and develops the C++ implementation for calculating moving averages.