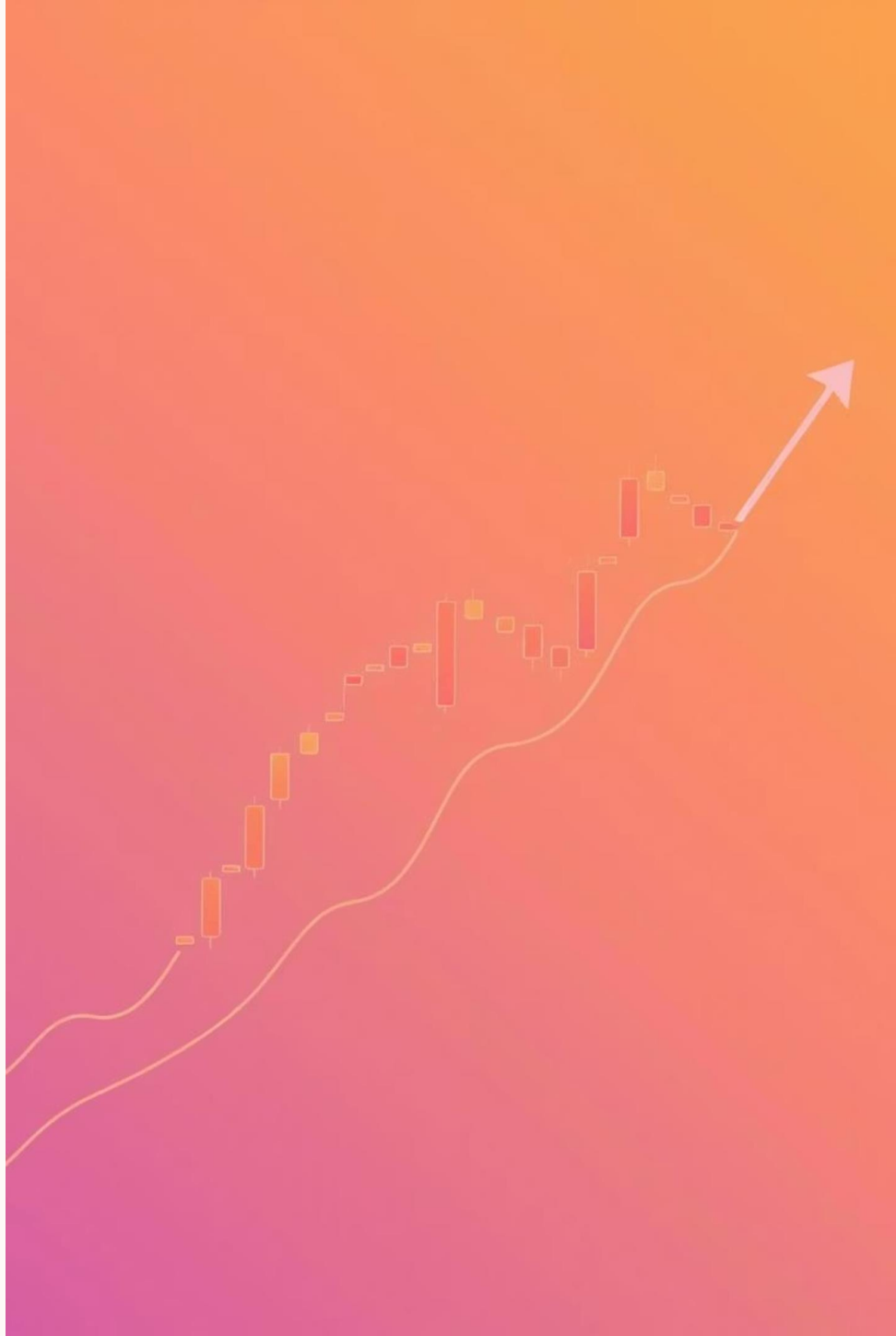
The Trading Guide App

A Streamlit-Based Stock Analysis and Forecasting Tool. Empowering informed financial decisions through intuitive data visualization and advanced machine learning.



Introduction

The Trading Guide App is an interactive, web-based platform developed using Streamlit. It combines traditional financial analysis with cutting-edge machine learning techniques to provide an intuitive tool for stock analysis, price forecasting, portfolio assessment, and financial model evaluation.

Comprehensive Stock Analysis

Dive deep into company fundamentals and technical indicators with ease.

Advanced Price Forecasting

Leverage AI/ML models to predict future stock trends with higher accuracy.

Strategic Portfolio Management

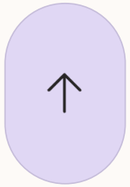
Optimize your investments by understanding correlation and risk.

Core Objectives

The primary goal of the Trading Guide App is to serve as a complete financial dashboard, equipping users with the necessary tools for in-depth market analysis and strategic decision-making.

Stock Performance & Fundamentals

In-depth analysis of individual stock behavior.



AI/ML Price Forecasting

Predictive modeling for future price movements.

Portfolio Correlation & Returns

Assessment of diversified investment strategies.

CAPM Risk-Return Evaluation

Measuring expected returns against systematic risk.

Model Accuracy Comparison (RMSE)

Benchmarking forecasting model performance.

Technology Stack

The Trading Guide App leverages a robust set of open-source technologies, primarily built within the Python ecosystem, to deliver its powerful functionality.

Programming Language: Python

The core language for all backend logic and data processing.

Framework: Streamlit

Enables rapid development of interactive web applications and dashboards.

Data: yfinance API

Provides real-time and historical stock market data.

Data Science Libraries

Pandas, NumPy, Matplotlib, Plotly, Seaborn, Scikit-learn for data manipulation, visualization, and ML tasks.

Forecasting Models

Statsmodels (ARIMA, SARIMA), Prophet (Meta), TensorFlow/Keras (LSTM) for predictive analytics.

System Architecture

The application's architecture is designed for clear separation of concerns, ensuring efficient data flow, processing, and interactive presentation.

Frontend: Streamlit

User-facing interface for interactive inputs and dynamic dashboard visualizations.

Backend: Python Modules

Handles data retrieval, complex financial calculations, and machine learning model execution.

Data Source: yfinance API

Serves as the primary conduit for real-time and historical stock data.

Forecasting Models

Integrates ARIMA, SARIMA, Prophet, and LSTM for diverse predictive capabilities.

Evaluation Metric: RMSE

Used consistently across forecasting modules for objective model performance comparison.

Key Application Modules

The Trading Guide App is structured into several modules, each serving a distinct function in the financial analysis workflow.

Home Page

Provides an application overview, navigation, and essential introductory information.

2

Stock Analysis

Detailed company metrics, technical charts (Candlestick, MA, RSI, Volume), and downloadable historical data.

3

Stock Prediction

30-day price forecasts with RMSE scores and comparative visualizations against actual prices.

Portfolio Analysis

Daily trends, returns, correlation matrices, and cumulative return plots for diversified portfolios.

5

Model Comparison

Direct comparison of ARIMA, SARIMA, Prophet, and LSTM performance via RMSE and visual plots.

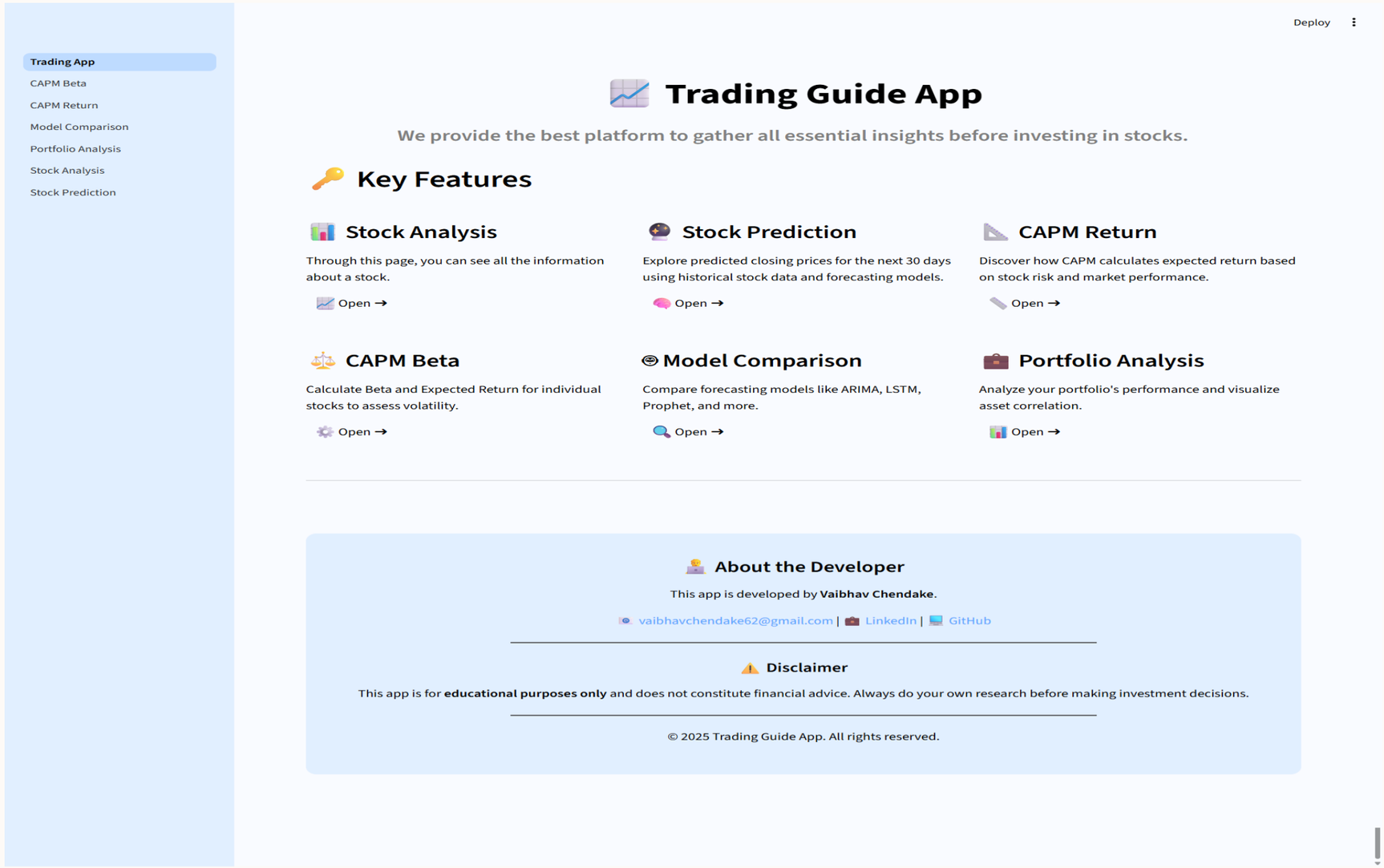
CAPM Return & Beta

Calculations of Beta and expected returns for individual or multiple stocks, with regression plots.

Modules

Home Page

Provides an application overview, navigation, and essential introductory information.



CAPM Return & Beta

Calculations of Beta and expected returns for individual or multiple stocks, with regression plots.

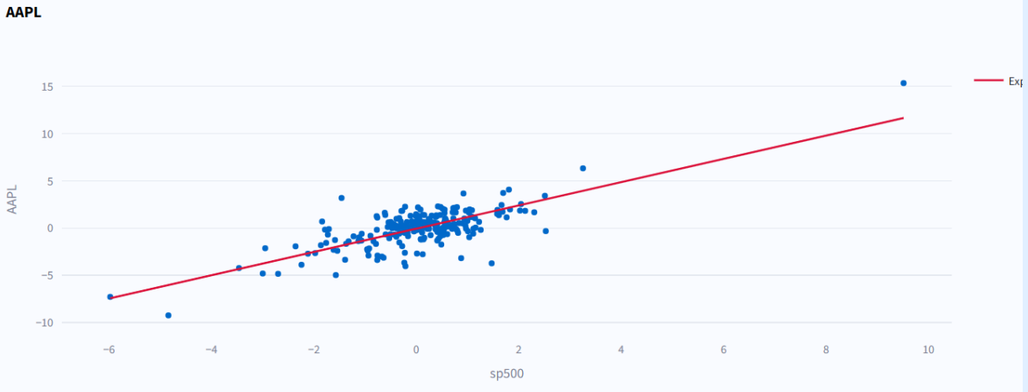
Calculate Beta and Return for individual stock

Choose a stock: AAPL

Number of Years: 1

Beta : 1.231620642328709

Return : 21.86



Trading App
CAPM Beta
CAPM Return
Model Comparison
Portfolio Analysis
Stock Analysis
Stock Prediction

Capital Asset Pricing Model

Choose 4 Stocks: TSLA × AAPL × MSFT × NFLX ×

Number of Years: 1

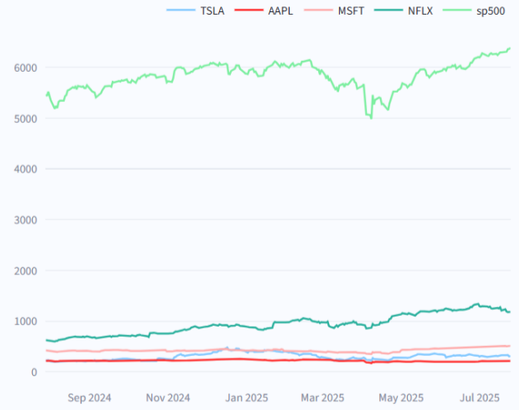
Dataframe head

	Date	TSLA	AAPL	MSFT	NFLX	sp500
0	2024-07-29 00:00:00	232.1	217.2241	423.4832	626.96	5463.54
1	2024-07-30 00:00:00	222.62	217.7815	419.7022	622.58	5436.44
2	2024-07-31 00:00:00	232.07	221.0462	415.1669	628.35	5522.3
3	2024-08-01 00:00:00	216.86	217.3436	413.9363	624.85	5446.68
4	2024-08-02 00:00:00	207.67	218.8366	405.382	613.64	5346.56

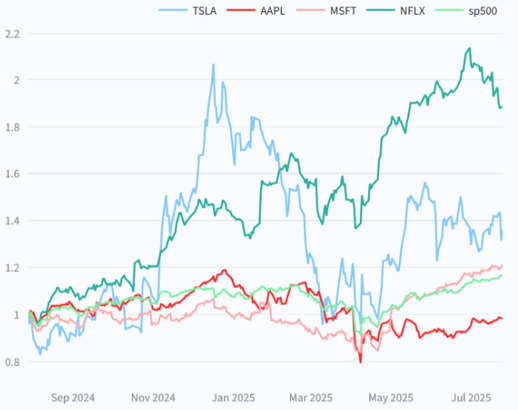
Dataframe tail

	Date	TSLA	AAPL	MSFT	NFLX	sp500
244	2025-07-21 00:00:00	328.49	212.48	510.06	1233.27	6305.6
245	2025-07-22 00:00:00	332.11	214.4	505.27	1190.08	6309.62
246	2025-07-23 00:00:00	332.56	214.15	505.87	1176.78	6358.91
247	2025-07-24 00:00:00	305.3	213.76	510.88	1180.76	6363.35
248	2025-07-25 00:00:00	316.06	213.88	513.71	1180.49	6388.64

Price of all the Stocks



Price of all the Stocks (After Normalizing)



Calculated Beta Value

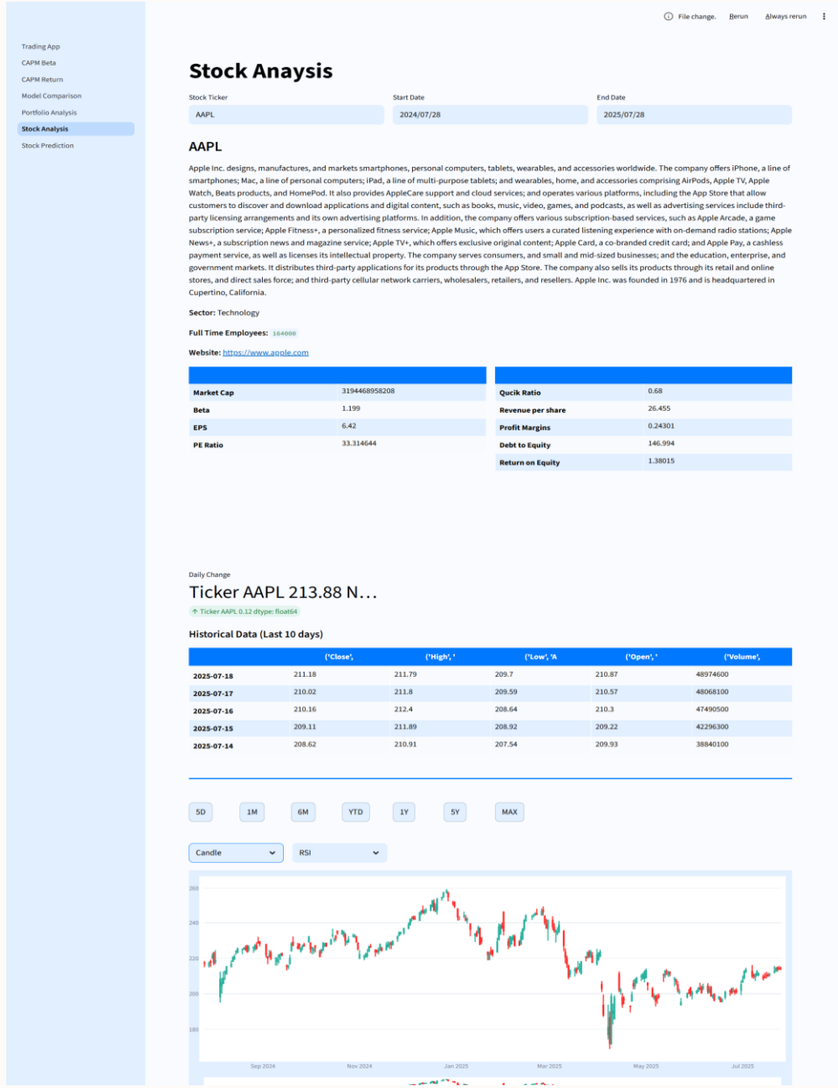
	Stock	Beta Value
0	TSLA	2.43
1	AAPL	1.23
2	MSFT	0.95
3	NFLX	0.96

Calculated Return using CAPM

	Stock	Return Value
0	TSLA	43.11
1	AAPL	21.86
2	MSFT	16.91
3	NFLX	17.09

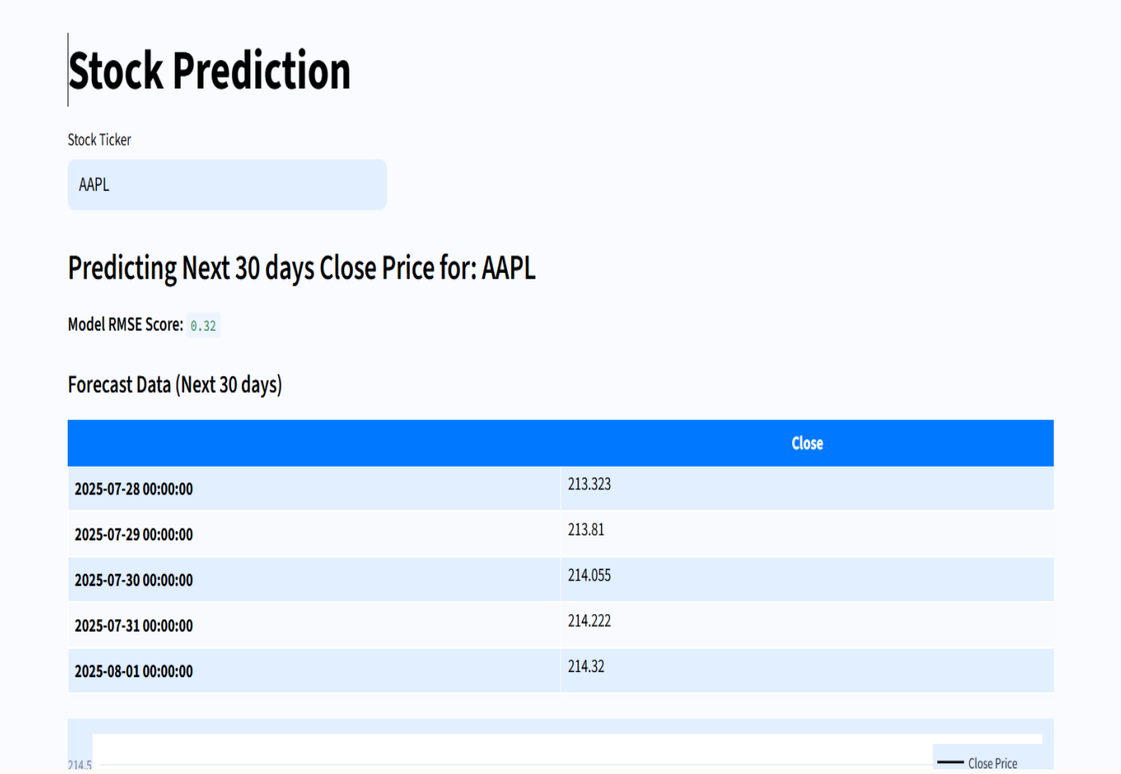
Stock Analysis

Detailed company metrics, technical charts (Candlestick, MA, RSI, Volume) , and downloadable historical data.



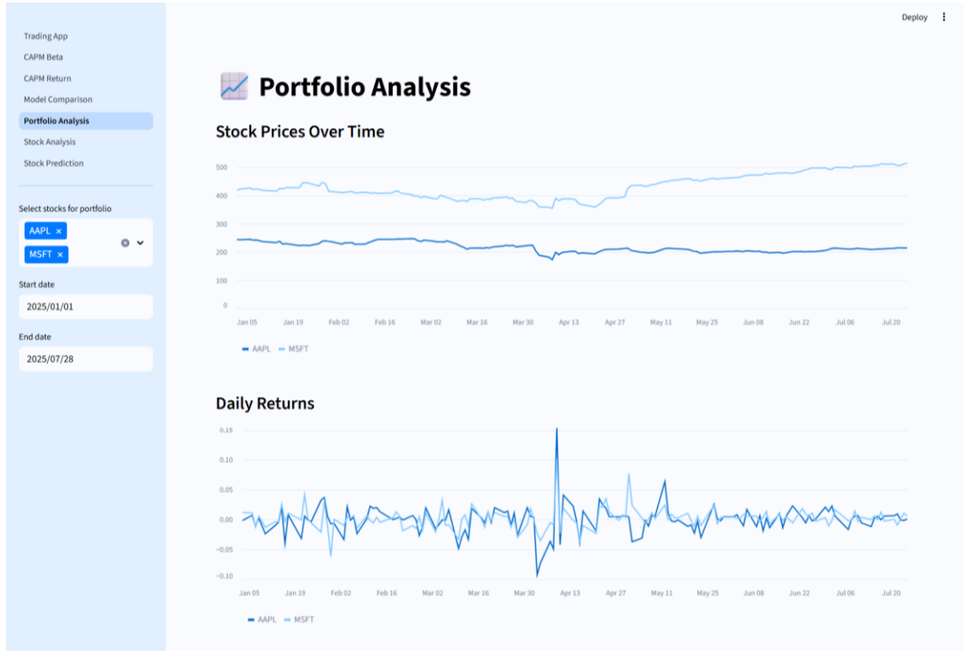
Stock Prediction

30-day price forecasts with RMSE scores and comparative visualizations against actual prices.



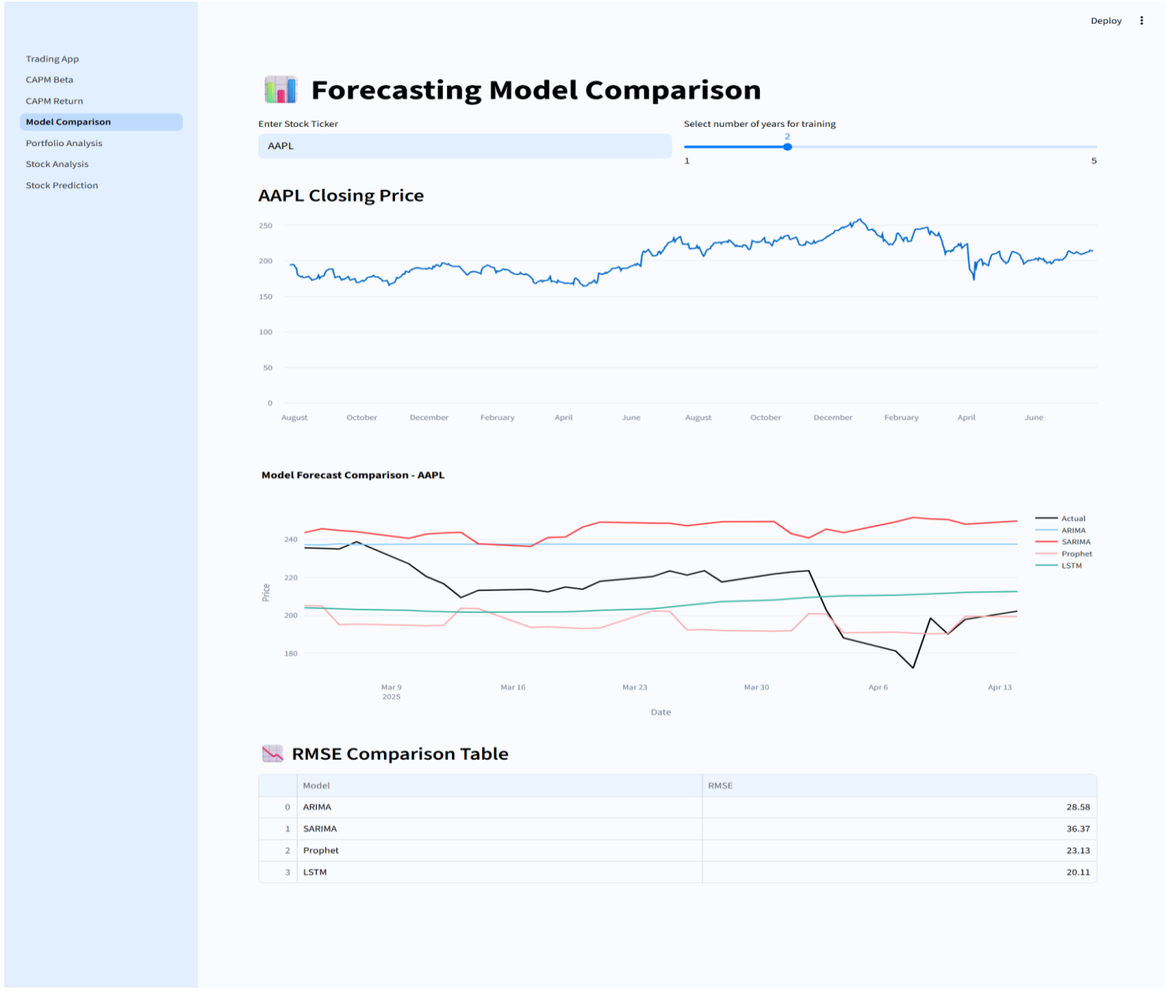
Portfolio Analysis

Daily trends, returns, correlation matrices, and cumulative return plots for diversified portfolios.



Model Comparison

Direct comparison of ARIMA, SARIMA, Prophet, and LSTM performance via RMSE and visual plots.



	Model	RMSE
0	ARIMA	28.58
1	SARIMA	36.37
2	Prophet	23.13
3	LSTM	20.11

Results & Observations

Throughout the development and testing phases, several key observations emerged regarding the efficacy and utility of the integrated models and features.

LSTM's Volatility Edge

Long Short-Term Memory (LSTM) networks demonstrated superior performance in forecasting highly volatile stocks, capturing complex non-linear patterns better than traditional models.

RMSE's Clarity

Root Mean Squared Error (RMSE) proved invaluable for clear, quantitative comparison of forecasting model accuracies, allowing for objective selection of the best-fit model.

Portfolio Insight

The Portfolio Analysis module effectively highlighted diversification benefits and identified concentration risks, aiding users in constructing more robust investment portfolios.

CAPM's Visual Power

The CAPM modules provided insightful visualizations of risk-return profiles and precise Beta calculations, enhancing understanding of individual asset and market relationships.

Real-time Practicality

The integration of real-time data analysis capabilities significantly increased the tool's practical utility for daily decision-making and market monitoring.

Conclusion

The Trading Guide App successfully integrates diverse financial and machine learning techniques into a comprehensive, user-friendly dashboard. It demonstrates the tangible benefits of applying data science to financial analysis.

> "A powerful tool for students, retail investors, and financial educators, offering a strong foundation for future advancements in fintech."

Future Scope

The current version of the Trading Guide App provides a robust foundation, with numerous exciting opportunities for future expansion and enhancement.

1 Enhanced User Experience

Implement user login and session tracking for personalized dashboards and saved analyses.

2 Sentiment-Driven Insights

Integrate news sentiment analysis using NLP (Natural Language Processing) from sources like Twitter and financial news APIs to provide qualitative market insights.

3 Strategy Backtesting

Develop a backtesting module to evaluate the historical performance of trading strategies before real-world application.

4 Wider Accessibility

Deploy the application on cloud platforms such as Streamlit Cloud, Heroku, or AWS for broader user accessibility and scalability.

5 AI for Trading Simulations

Introduce Reinforcement Learning (RL) for advanced trading simulations and automated strategy optimization.

Bibliography & Resources

This project was built upon the foundations of excellent open-source libraries and extensive financial research.

Data Acquisition	yfinance API Documentation
Application Framework	Streamlit Documentation
Time Series Forecasting	Prophet by Meta, Statsmodels Documentation
Deep Learning	TensorFlow LSTM Guide
Machine Learning & Statistics	Scikit-learn Documentation
Financial Concepts	Investopedia – Financial Concepts

Thank you for your attention.