Department of Artificial Intelligence and Data Science

FinTrack: Smart Portfolio Management and Analytics

Dr.Manoranjini
M.E.,Ph.D.,
Professor

R JAI SAARATHI (221801019)
O JOEL SAM (221801021)

Introduction

This project introduces **"FinTrack: Smart Portfolio Management & Analytics"**, a Streamlit-based web application designed to simplify the process of portfolio tracking, analysis, and optimization. It allows users to input stock tickers, define quantities, and view visualizations and metrics to analyze the portfolio's performance and risk over time. This tool is targeted at both individual investors and financial analysts who seek to optimize their portfolio for better returns while managing risk effectively.

Existing System

Traditionally, investors have relied on offline tools such as spreadsheets (Excel, Google Sheets) and standalone financial software for portfolio management and analysis. However, these solutions often lack automation, real-time data fetching, and advanced statistical analysis capabilities. Some of the existing methods include

- **Excel/Google Sheets**
- **Standalone Financial Software**
- **Brokerage Platforms**

Advantages and Disadvantages of Existing System

Advantages:

- Customizability
- Wide Adoption
- Feature-Rich Platforms

Disadvantages:

- Lack of Real-Time Data
- Lack of real-time analysis
- Complexity
- Inefficient handling of large datasets

Proposed System

The **Proposed System** is a modern, interactive, and user-friendly portfolio management application that integrates real-time stock data, advanced performance and risk analytics, and dynamic visualizations. The system is built using **Streamlit**, a Python framework, and incorporates data from **Yahoo Finance** to provide up-to-date information on stock prices and portfolio performance.

Key Features:

- Real-time data analysis
- Advanced risk assessment tools
- User-friendly interface
- Automated portfolio optimization algorithms

Advantages and Disadvantages of Proposed System

Advantages:

- Enhanced data accuracy
- Real-time decision-making
- Improved user experience
- Comprehensive risk analysis and diversification strategies

Disadvantages:

- High initial development cost
- Potential complexity for novice users
- Dependence on real-time data feeds

SRS (Software Requirements Specification)

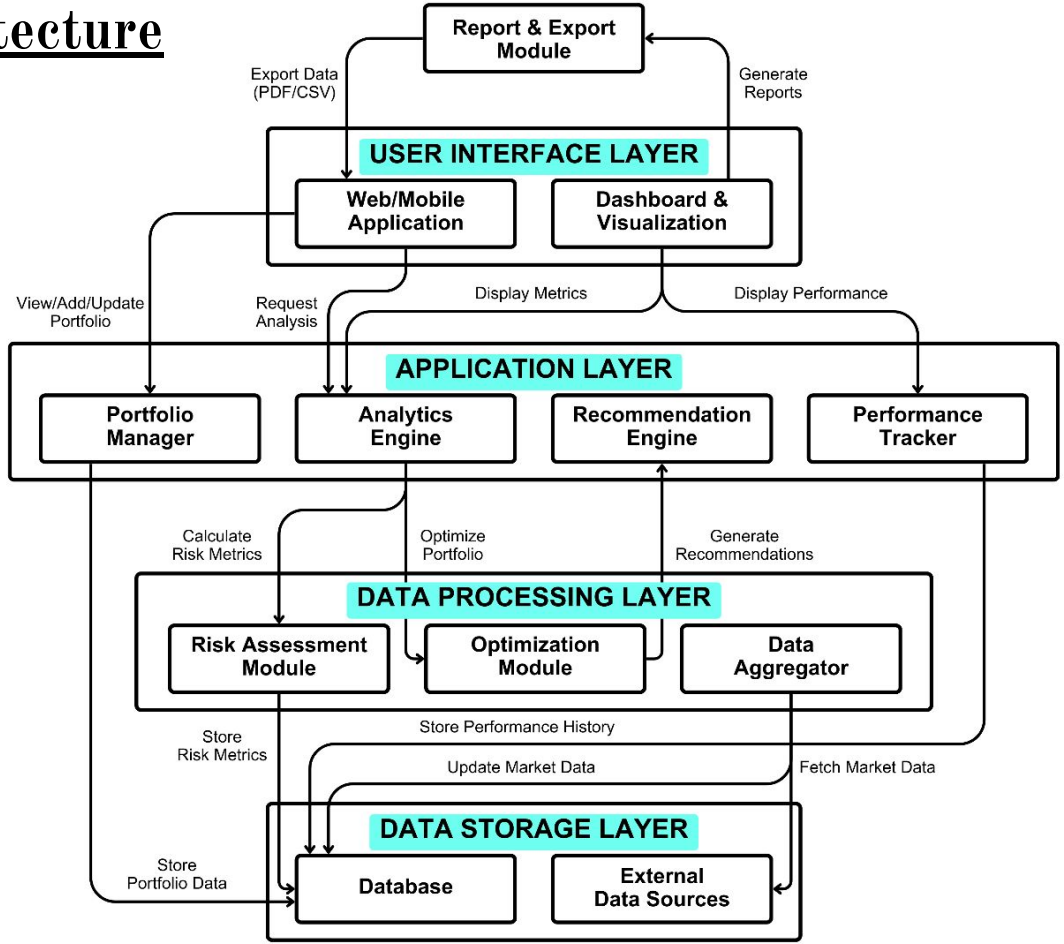
Functional Requirements:

- **User Authentication**: Users must be able to create accounts and log in.
- **Portfolio Management**: Users should be able to create, modify, and analyze portfolios.
- **Real-time Data Integration**: The system should fetch and display real-time market data.
- **Risk Analysis**: Provide detailed risk assessments for portfolios.
- **Optimization**: Offer portfolio optimization based on user-defined criteria.

Non-Functional Requirements:

- **Performance:** The system should process data quickly and provide real-time updates.
- **Security:** Ensure that user data and transactions are secure.
- **Scalability:** The system should handle increasing numbers of users and data without performance degradation.
- **Usability:** The interface should be intuitive and easy to navigate.
- **Reliability:** The system should have high availability and minimal downtime.

System Architecture



Software Development Model(Waterfall Model):

Phase	Activities	Deliverables
1. Requirements Gathering	Collect user needs, define system features, document functional/non-functional requirements.	Requirements Specification Document
2. System Design	High-level system architecture, database design, UI/UX wireframes, and security specifications.	System Design Document
3. Implementation (Coding)	Develop the front-end and back-end systems, integrate APIs, and implement the database.	Codebase (source code)
4. Testing	Unit testing, integration testing, system testing, security testing, performance testing, and user acceptance testing.	Test Reports
5. Deployment	Deploy the application to production, migrate data, and set up monitoring tools.	Deployed System
6. Maintenance	Bug fixes, updates, system performance monitoring, and adding new features based on feedback.	Updated System

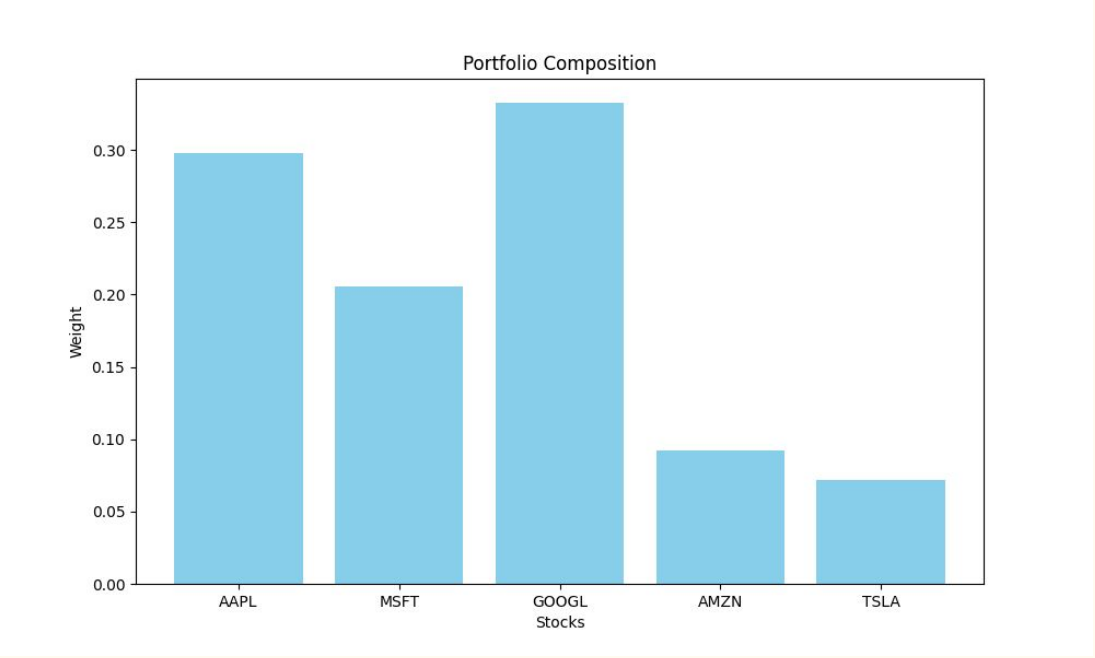
Unit Testing

```
PS C:\Users\ADMIN\OneDrive\Desktop\SE> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe c:/Users/ADMIN/OneDrive/Desktop/SE/main.py
....Pie Chart Execution Time: 0.1846 seconds
Line Chart Execution Time: 0.0648 seconds
Bar Chart Execution Time: 0.0558 seconds
Scatter Chart Execution Time: 0.0568 seconds
..
----------------------------------------------------------------------
Ran 6 tests in 0.362s

OK
```

Functional Testing

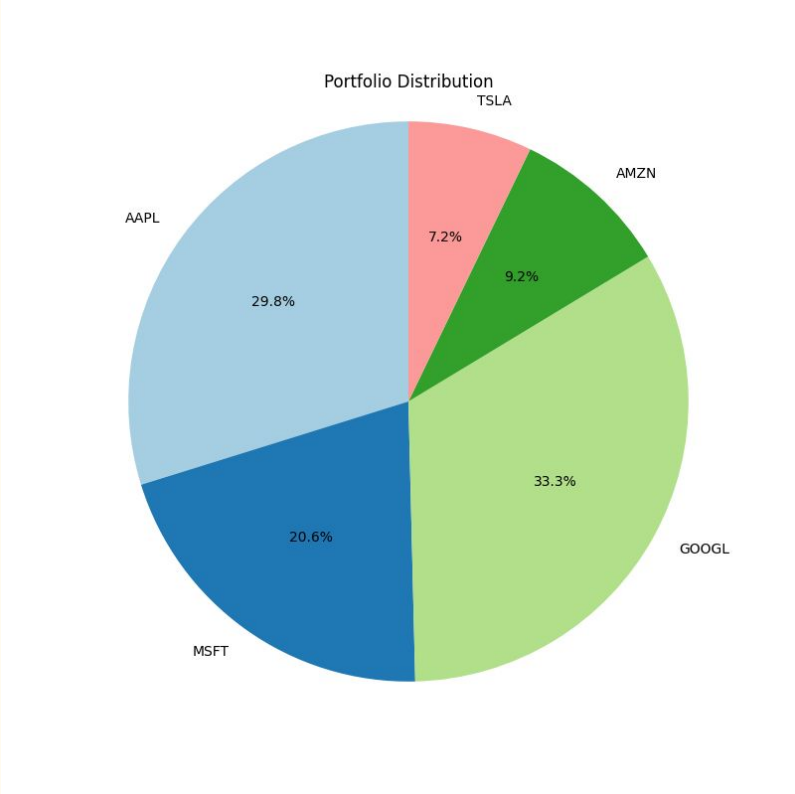
```
PS C:\Users\ADMIN\OneDrive\Desktop\SE> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe c:/Users/ADMIN/OneDrive/Desktop/SE/Functional_test.py
..Pie Chart Execution Time: 0.1027 seconds
Line Chart Execution Time: 0.0682 seconds
Bar Chart Execution Time: 0.0561 seconds
Scatter Chart Execution Time: 0.0558 seconds
...
----------------------------------------------------------------------
Ran 5 tests in 0.284s

OK
```

Output

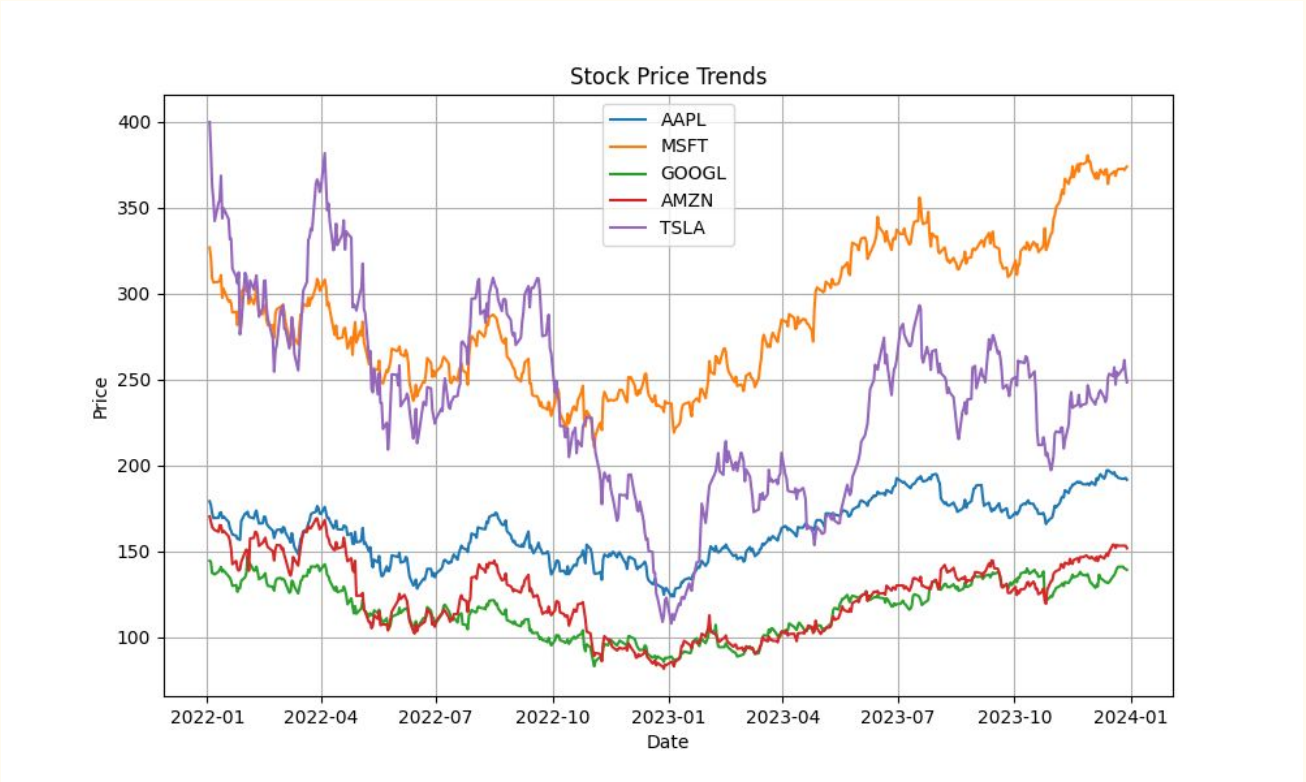



```
PS C:\Users\ADMIN\OneDrive\Desktop\SE> & C:/Users/ADMIN/AppData/Local/Programs/Python/Python312/python.exe c:/Users/ADMIN/OneDrive/Desktop/SE/try.py
[*********************100%***********************]  5 of 5 completed
c:\Users\ADMIN\OneDrive\Desktop\SE\try.py:87: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.
  stock_data = stock_data.fillna(method='ffill')  # Forward fill missing data
Portfolio Return: 0.05
Portfolio Risk: 0.31
Optimal Portfolio (Max Sharpe Ratio): Return=0.09, Risk=0.28, Sharpe Ratio=0.33
```

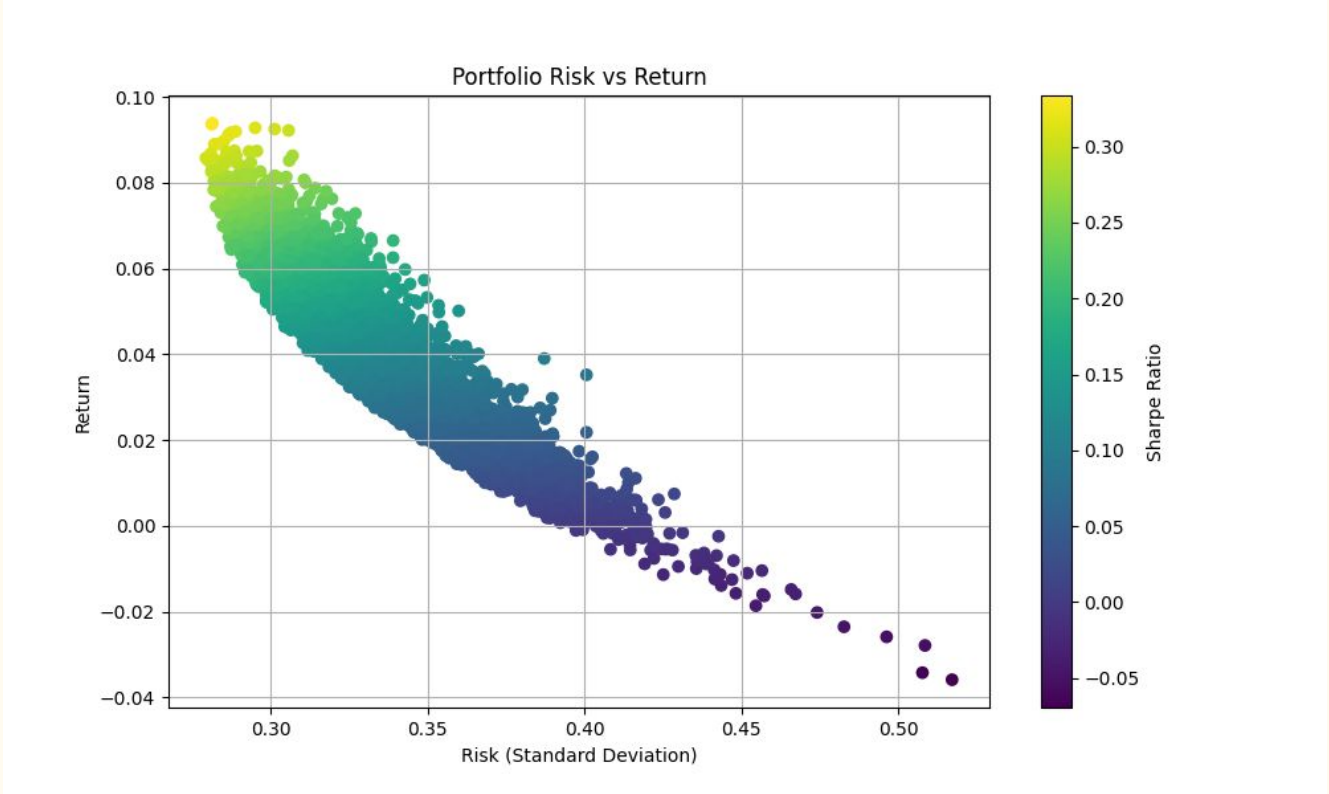
Output


Portfolio Distribution
TSLA
AMZN
AAPL
7.2%
9.2%
29.8%
33.3%
20.6%
GOOGL
MSFT

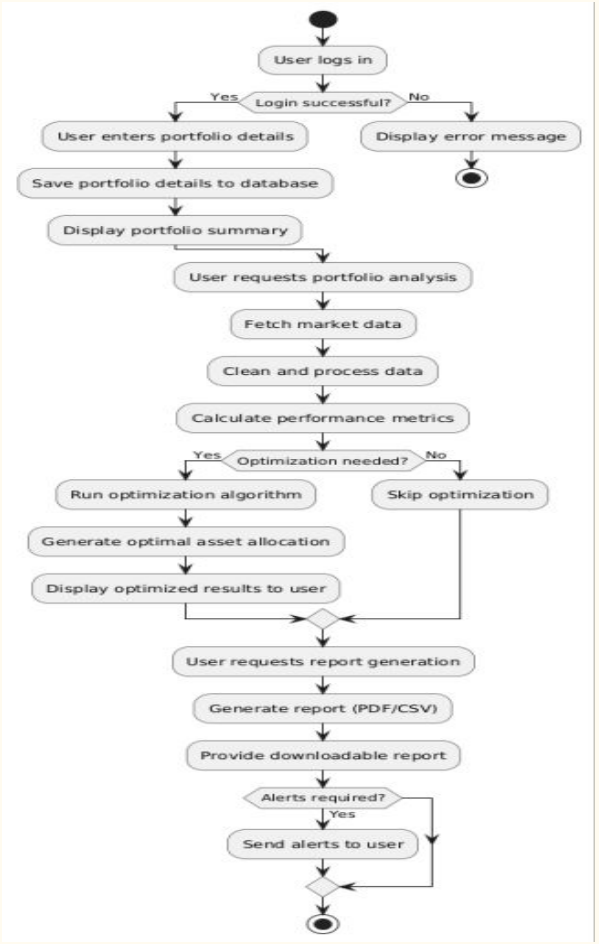
Output


Stock Price Trends
AAPL
MSFT
GOOGL
AMZN
TSLA
Price
400
350
300
250
200
150
100
2022-01
2022-04
2022-07
2022-10
2023-01
2023-04
2023-07
2023-10
2024-01
Date

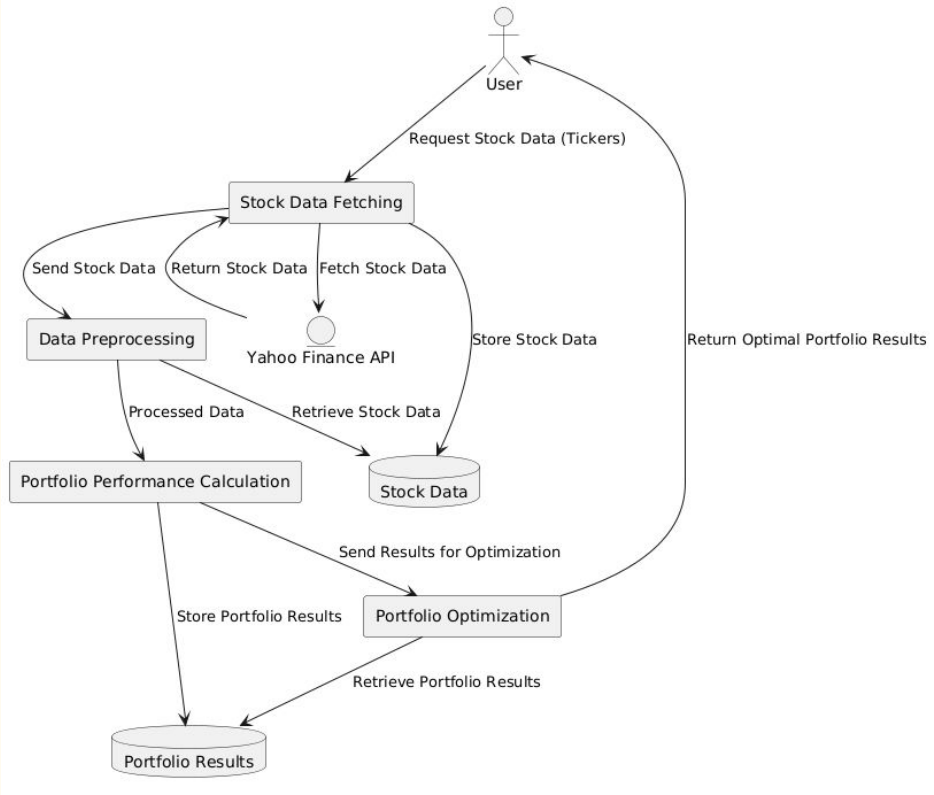
Output


Portfolio Risk vs Return
Return
Risk (Standard Deviation)
Sharpe Ratio
0.10
0.08
0.06
0.04
0.02
0.00
−0.02
−0.04
0.30
0.35
0.40
0.45
0.50
0.30
0.25
0.20
0.15
0.10
0.05
0.00
−0.05

Activity Diagram:



Data Flow Diagram:

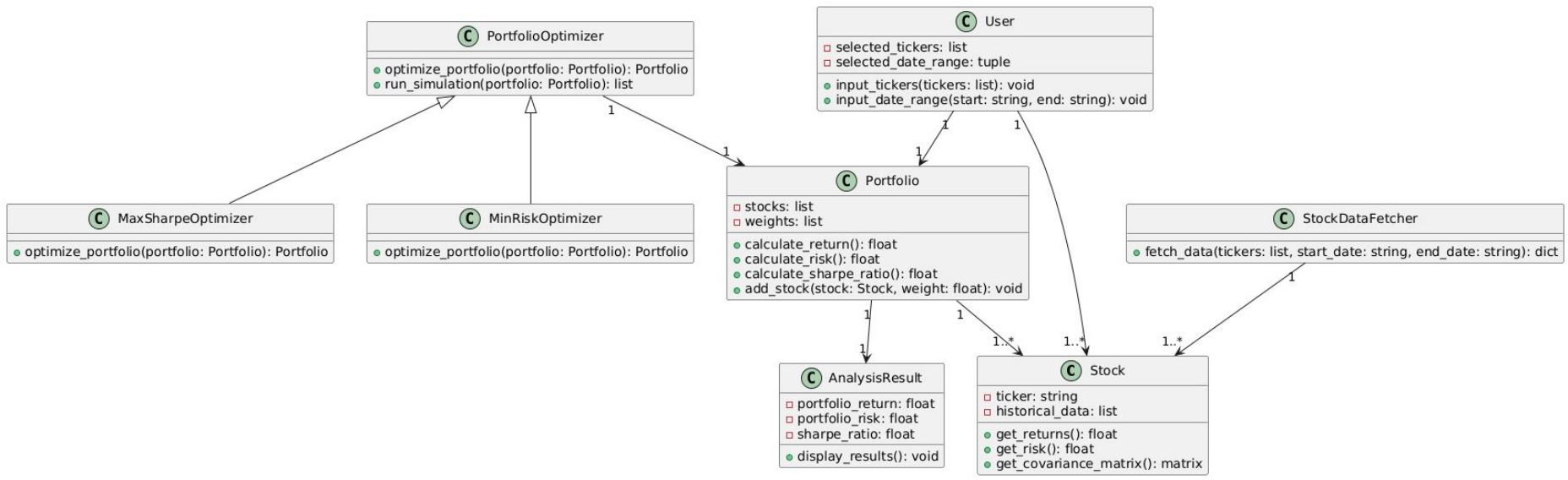


Object-Oriented Design

FinTrack uses Object-Oriented Programming (OOP) principles for flexibility and ease of maintenance. Key components include:

1. **StockDataFetcher**: Retrieves real-time stock data from Yahoo Finance using the **yfinance** API.
2. **Portfolio**: Manages portfolio composition (tickers, quantities), calculates portfolio return, risk, and asset weights.
3. **Visualization**: Generates charts (bar, pie, scatter) for portfolio analysis, helping users visualize performance and risk.
4. **Optimizer** (Future): Potential feature to optimize portfolio weights using techniques like Markowitz optimization.

Object Oriented Design:



Conclusion

FinTrack is a cost-effective, real-time portfolio management tool designed to provide individual investors with an easy-to-use platform for tracking and optimizing stock portfolios. The system offers several advantages, including real-time analysis, interactive visualizations, and customizable portfolio management features. However, it has some limitations, such as being limited to stocks and basic analytics. Despite these constraints, **FinTrack** is an ideal solution for users seeking a dynamic, intuitive platform to manage their investments and make informed decisions.

Reference

- Markowitz, H. M., "Portfolio Selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77-91, 1952. doi: 10.1111/j.1540-6261.1952.tb01525.x.

- Sharpe, W. F., "Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk," *The Journal of Finance*, vol. 19, no. 3, pp. 425-442, 1964. doi: 10.1111/j.1540-6261.1964.tb02865.x.

- Markowitz, H. M., "The Efficient Frontier," *The Journal of Financial and Quantitative Analysis*, vol. 5, no. 3, pp. 321-342, 1970. doi: 10.2307/2330072.

- Black, F., and Litterman, R., "Global Portfolio Optimization," *Financial Analysts Journal*, vol. 48, no. 5, pp. 28-43, 1992. doi: 10.2469/faj.v48.n5.28.

- Chen, J., and Zheng, Z., "Stock Market Prediction with Machine Learning Techniques," *Journal of Financial Data Science*, vol. 1, no. 1, pp. 1-19, 2019. doi: 10.3934/jfds.2019.1.1.

Thank You