



UNIVERSITY OF MADRAS  
INSTITUTE OF DISTANCE EDUCATION

சென்னைப் பல்கலைக்கழகம்  
தொலைதூரக் கல்வி நிறுவனம்

## PREDICT PRO ON STOCK INVESTMENT PORTFOLIO MANAGER

Submitted by : Ponyogesh P  
Roll No. : A22101PCA6926  
Guided By : Mr. Thirunavukarasu

# Objectives

- Develop and implement cutting-edge predictive analytics using machine learning algorithms to forecast stock market trends, enabling users to make informed investment decisions.
- Integrate with real-time market data feeds to ensure timely updates on stock prices, market movements, and economic indicators, providing users with the latest information for strategic decision-making.
- Implement sophisticated risk assessment tools that analyze various risk factors associated with each investment. Provide users with risk-reward analyses to optimize portfolios based on individual risk tolerance and financial goals.

# Abstract

In the dynamic landscape of financial markets, effective portfolio management is crucial for investors seeking optimal returns while managing risk. PREDICT PRO emerges as an advanced Stock Investment Portfolio Manager designed to empower investors with intelligent tools and analytics for informed decision-making. Stock price prediction plays a crucial role in financial markets, enabling investors, traders, and financial analysts to make informed decisions. The primary objective of this research is to explore the effectiveness of models in predicting stock prices. This project combines cutting-edge technology, data analysis, and financial expertise to create a robust platform that enhances the efficiency and effectiveness of stock portfolio management.

## Hardware Specifications :

Processor	Intel (R) Dual Core
Processor Speed	2.4Ghz
RAM	4 GB RAM
Storage	512 MB

## Software Specifications :

System Type	64-bit Operating System
Operating System	Windows
Programming language	Python, JavaScript, HTML, CSS
IDE & Text Editor	Visual Studio Code
Database	SQLite

# Existing System

Existing models may have limitations in capturing the complexity of financial markets, especially during unusual or highly volatile events.

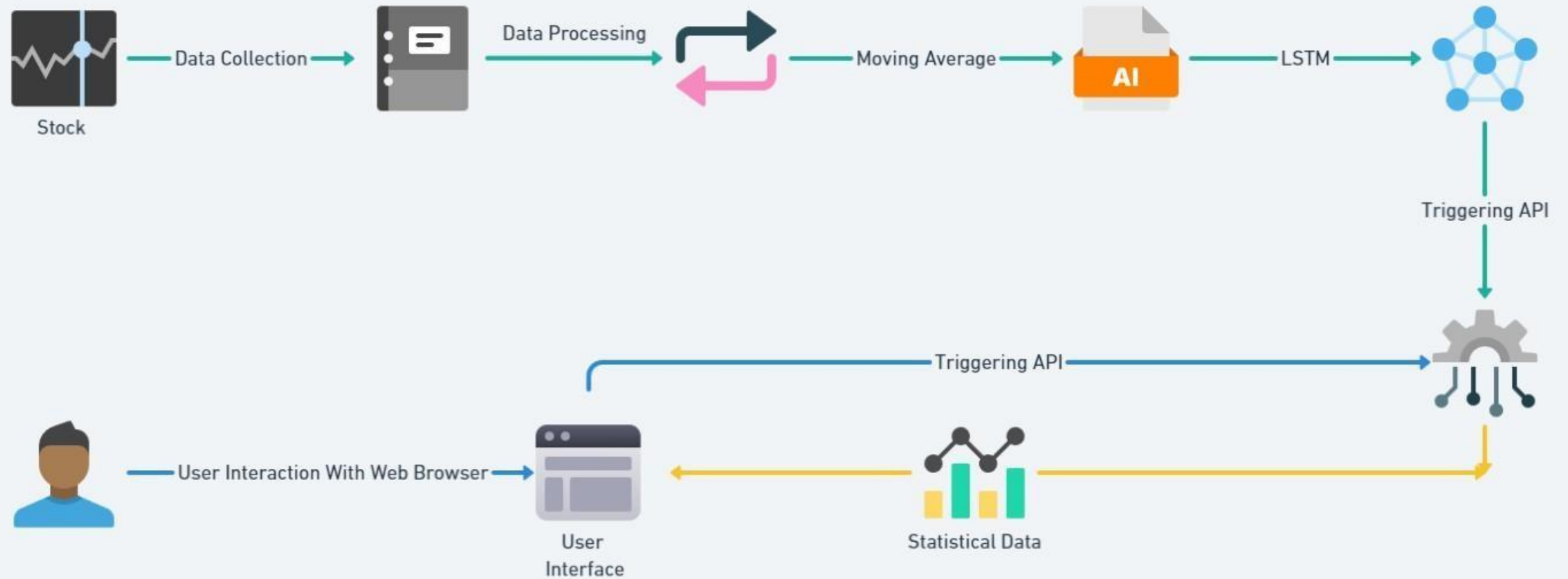
In this existing system, the collaboration between human agents and automated tools strives to offer a comprehensive customer support experience. While human agents handle intricate inquiries, automated tools like FAQs serve as valuable self-help resources. However, the challenge lies in addressing the limitations related to human agent availability, aiming to enhance responsiveness and minimize user waiting times

# Proposed System

The proposed system, aims to revolutionize stock investment portfolio management by addressing the drawbacks of existing systems and introducing innovative features. PREDICT PRO is designed to provide investors with a comprehensive, and technologically advanced platform. Here are the key features and improvements proposed for the system.

The system will seamlessly integrate with real-time market data feeds, ensuring that portfolio information is continuously updated.

# Overall system design



## ► Data Retrieval:

- Yahoo API returns real time data.

## ► Algorithmic Analysis:

- LSTM: Algorithm LSTM to be applied.
- Moving Average: Statistical analysis provides insights into the dataset.

## ► Creating AI Model:

- Creating a Model to Process new records in the location and update in Database.

## ► Visualization:

- Analyzed data is plotted on customized UI.
- Statistical insights represented through charts/graphs.

## ► Reporting:

- Generation of reports summarizing insights.

## ► User Interface:

- Development of a user-friendly web interface.



## Modules completed with screenshots :

1. Importing From Yahoo Finance
2. Fetching Data Set
3. Analyzing the Data Set
4. Understanding Data
5. Adding Custom Column For Reference
6. Figuring the multiple datasets
7. Visualize the Model Training in Graph

# Objectives and Scope of Proposed Work

## 1) Importing From Yahoo Finance

```
In [16]: # For reading stock data from yahoo
from pandas_datareader import DataReader
import yfinance as yf
# install Datareader and yfinance using pip
from pandas_datareader import data as pdr

yf.pdr_override()
#The above line is to override Python datareader with Yahoo library to use rich and effiecient datas

# For time stamps
from datetime import datetime
```

```
In [21]: # For reading stock data from yahoo
from pandas_datareader import DataReader
import yfinance as yf
# install Datareader and yfinance using pip
from pandas_datareader import data as pdr

yf.pdr_override()
#The above line is to override Python datareader with Yahoo library to use rich and effiecient datas

# For time stamps
from datetime import datetime
```

## ► 2) Fetching Data set

```
In [17]: #Here i am choosing my Favourite stocks
techs_list=["TSLA", "GOOG"]
tech_list=["TSLA", "GOOG","RELIANCE","TCS"]
# Here i am choosing the start date from when i want the data to be imported from yahoo finance api
end = datetime.now()
start=datetime(end.year - 1, end.month, end.day)
```

### ► 3) Analyzing the Data Set

```
In [19]: TCS.head()  
# Checking Wheather the download is successfull or not
```

```
Out[19]:
```

	Open	High	Low	Close	Adj Close	Volume
Date						
2022-10-31	3199.000000	3199.000000	3177.800049	3193.149902	3090.592529	1919414
2022-11-01	3214.500000	3262.600098	3203.449951	3259.699951	3155.005127	2991060
2022-11-02	3270.000000	3270.000000	3222.050049	3241.699951	3137.583496	1405955
2022-11-03	3228.050049	3228.050049	3195.000000	3206.750000	3103.755859	1422652
2022-11-04	3217.000000	3220.050049	3166.149902	3217.399902	3114.063721	1464013

## ► 4) Understanding Data

```
In [24]: TSLA.describe() --understanding data  
#just a overall stats to see wheather we have any missed values
```

```
Out[24]:
```

	Open	High	Low	Close	Adj Close	Volume
<b>count</b>	250.000000	250.000000	250.000000	250.000000	250.000000	2.500000e+02
<b>mean</b>	206.711440	211.139080	202.068120	206.665200	206.665200	1.360863e+08
<b>std</b>	44.944099	45.314290	44.646124	44.988614	44.988614	4.015948e+07
<b>min</b>	103.000000	111.750000	101.809998	108.099998	108.099998	5.067270e+07
<b>25%</b>	176.367504	182.379997	172.502499	178.187496	178.187496	1.076827e+08
<b>50%</b>	198.234993	202.845001	192.885002	197.474998	197.474998	1.283030e+08
<b>75%</b>	249.542500	255.235001	244.472496	250.217503	250.217503	1.610446e+08
<b>max</b>	296.040009	299.290009	289.519989	293.339996	293.339996	3.065906e+08



## ► 5) Adding Custom Column For Reference

```
In [21]: for company, com_name in zip(company_list, company_name):  
         company["company_name"] = com_name  
  
         ## i am iterating through two lists to add a column in comapny list for future Representation
```

```
In [22]: TSLA.head()  
         # As you can see now all variables have extra column with name of "company name"
```

```
Out[22]:
```

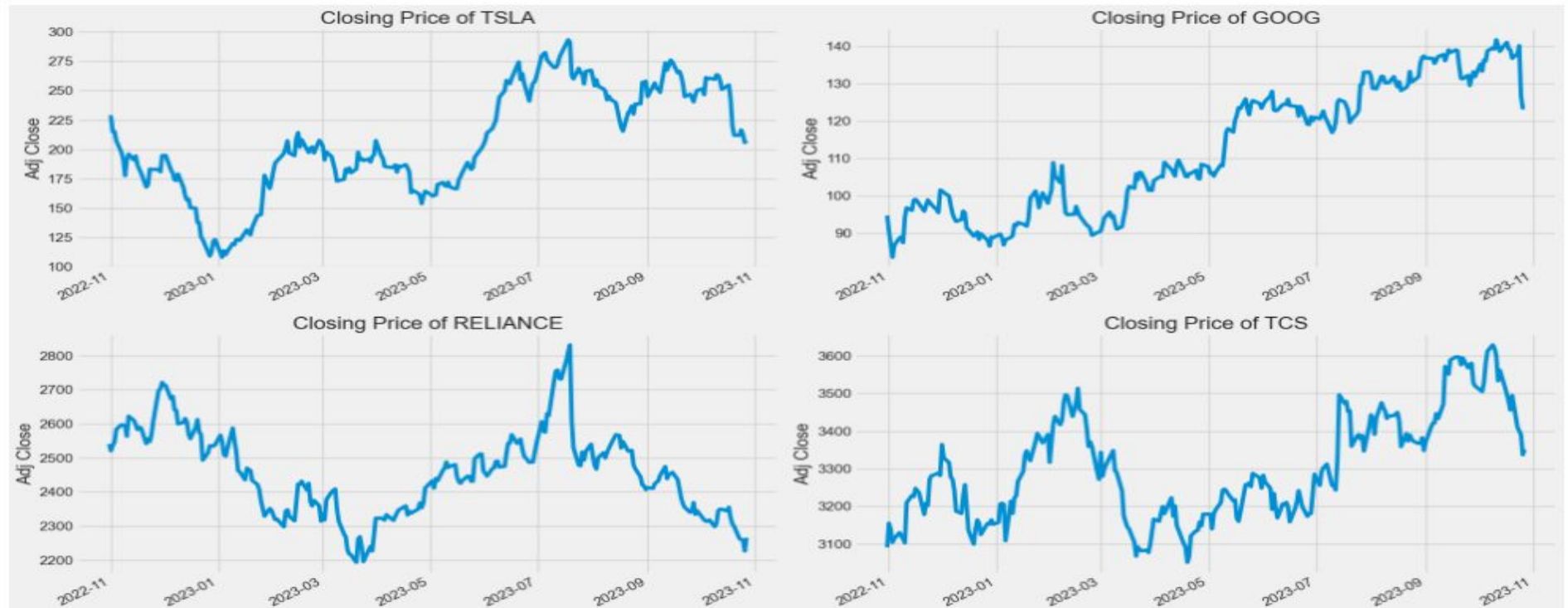
	Open	High	Low	Close	Adj Close	Volume	company_name
Date							
2022-10-31	226.190002	229.850006	221.940002	227.539993	227.539993	61554300	TESLA
2022-11-01	234.050003	237.399994	227.279999	227.820007	227.820007	62688800	TESLA
2022-11-02	226.039993	227.869995	214.820007	214.979996	214.979996	63070300	TESLA
2022-11-03	211.360001	221.199997	210.139999	215.309998	215.309998	56538800	TESLA
2022-11-04	222.600006	223.800003	203.080002	207.470001	207.470001	98622200	TESLA

## ► 6) Figuring the multiple datasets

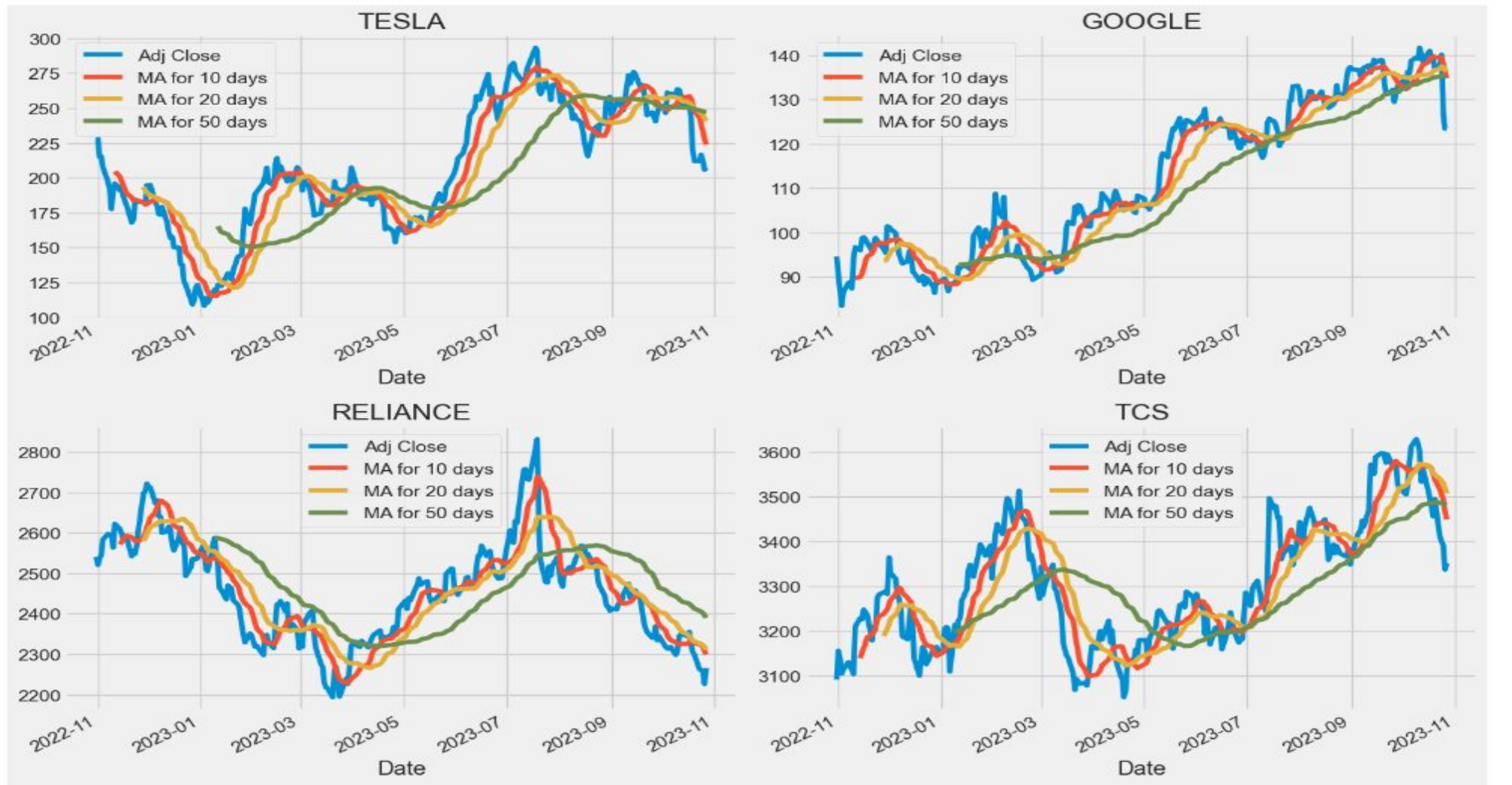
```
In [26]: plt.figure(figsize=(20, 10))
plt.subplots_adjust(top=2.25, bottom=2.2)

for i, company in enumerate(company_list, 1):
    plt.subplot(2, 2, i)
    company['Adj Close'].plot()
    plt.ylabel('Adj Close')
    plt.xlabel(None)
    plt.title(f"Closing Price of {tech_list[i - 1]}")

plt.tight_layout()
```



► 7) Visualize the Model Training in Graph





► Data Set:

#	Column
---	-----
0	Open
1	High
2	Low
3	Close
4	Adj Close
5	Volume
6	company_name

► Pseudo code for incomplete Modules :

► # Data Analysis

Apply Descriptive Statistics for insights into the dataset.

► # Visualization

1. Visualize the analyzed data on personalized UI for easy interpretation.
2. Create charts and graphs to represent statistical insights.

► # Reporting

1. Generate reports based on the analysis.
2. Provide insights in graph.

▶ Pseudo code for incomplete Modules :

▶ # User Interface

1. Develop a web-based tool for users to interact with the data.

▶ # Testing

1. Perform rigorous testing of the tool to ensure functionality.
2. Debug and fix any issues identified during testing.

▶ # Deployment

1. Deploy the web-based tool on a suitable server.
2. Make the tool accessible to users.

# End of Pseudo-code

# Test Cases

Module	Test Cases
Yahoo Data Retrieval	1. Ensure that the application successfully retrieves map data from Google Maps API
	2. Verify that the retrieved data includes details such as business locations, types and ratings
Duplicate Removal	3. Test the functionality to remove duplicate entries from the dataset in MongoDB.
	4. Ensure that the dataset in MongoDB does not contain any duplicate records.
Data Analysis	5. Validate the accuracy of the algorithm analysis results.
	8. Test the response of the system to different types of data inputs.
Model Functionality	9. Confirm that the modal for analyzing the dataset opens and displays relevant insights.
	10. Check for proper error handling if the modal fails to load or analyze the dataset.

# Test Cases :

User Interface	11. Ensure that the user interface is responsive and displays information correctly
	12. Validate that the map is interactive, allowing users to explore local business landscapes.
Algorithm Validation	13. Verify the correctness of the K-Means clustering algorithm in providing meaningful insights.
	14. Confirm that the decision tree algorithm produces accurate predictions for business-related data.
Compatibility	19. Verify that the application works on different web browsers (Chrome, Firefox, Safari, etc.)
	20. Test the tool's compatibility with different devices (desktop, tablet, mobile).

# Conclusion

In conclusion, the PREDIT PRO - Stock Investment Portfolio Manager is a comprehensive and innovative solution designed to empower investors with intelligent tools and analytics for effective portfolio management. The project combines advanced technology, data analysis, and financial expertise to create a robust platform that enhances the efficiency and effectiveness of stock portfolio management. The key features and modules, including Predictive Analytics, Real-time Market Data Integration, Risk Management and Analysis, Decision Support, and Integration with Emerging Technologies, collectively contribute to a sophisticated platform.