STOCK PREDICTION

A Project Report

MASTERS OF SCIENCE (INFORMATION TECHNOLOGY)

By

Rechael Vincent Lopes

Seat Number – 4136040

Under the Esteemed Guidance of

Mrs. Calvina Maharao

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. GONSALO GARCIA COLLEGE OF ARTS AND COMMERCE

(Affiliated with University of Mumbai)

Vasai Road(W) Maharashtra 401202

2022 - 2023

ST. GONSALO GARCIA COLLEGE OF ARTS AND COMMERCE

(Affiliated with University of Mumbai)

Vasai Road(W) Maharashtra 401202

DEPARTMENT OF INFORMATION TECHNOLOGY

CRETIFICATE

This is to certify that the project entitled, **'Stock Prediction'** is bonafied work of **Ms. Rechael Vincent Lopes** bearing **seat no - 4136040** submitted in partial fulfillment of the requirements for the award of degree **Masters of Science** in **Information Technology** from University of Mumbai.

Internal Guide Coordinator

External Examiner

Date: College Seal

ACKNOWLEDGEMENT

The successful completion of this project would be impossible with the immense help and support that I have received from my faculties. I take pride in presenting before you my project which is a result of dedication both in research and knowledge.

I would like to express my sincere gratitude to **Dr. Somnath Vibhute** our Principal, **Dr. Sangita Dubey** Head of the department of Information Technology. A special thanks goes to **Prof. Calvina Maharao** my project guide, for her constant support and encouragement. I am grateful for her cooperation and her valuable suggestions.

Last and also an important gratitude goes to my fellow batchmates who helped and guided me throughout the process.

ABSTRACT

The **Stock Prediction** Project is designed to predict the actual and closing price of any stock along with the actual and predicted graph. The proposed system tries to find the accurate value of the next day closing value. The Long Short Term Memory (LSTM) is an artificial neural network that is based on a Machine Learning Algorithm and is in the field of Deep Learning. LSMT is an advanced neural network with a memory cell that stores a small amount of data for future references. LSMT is a feedback link that makes it a 'General Purpose Computer'. LSMT algorithm is more suitable in predicting the future stock prices than the Super Vector Machine (SVM) algorithm and Back Propagation algorithm as it filters unnecessary data unlike the other two. The time and memory requirements are also reduced and the algorithm is more proper in handling non-linear data.

DECLARATION

I hereby declare that the Project entitled, '**Stock Prediction'** done at St. Gonsalo Garcia College has not been in any form of case duplication submitted to any other university for the award of any degree. To the best of my knowledge this project is completely my work and no one else is involved in it.

The Project is done in partial fulfilment of the requirement for the award of degree of **Masters of Science** in **Information Technology** to be submitted as the final year project as part of the curriculum.

Rechael Vincent Lopes

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SYNOPSIS

Name: Rechael Vincent Lopes **Seat No:** 4136040

Class: MSc. IT. Part II **Semester:** Sem 3

Project Topic: Stock Prediction.

Introduction:

Stock Market is a shared place where public trading is done on listed companies. The primary market is where general public view company shares in an initial public offering (IPO) to raise capital. The secondary market is where actual stock exchange happens. These stock exchanges are regulated by the Security and Exchange Board of India (SEBI).

In current times the demand for buying and selling of stocks and also the general interest of public in stock marketing has increased tremendously with each new company investing into stock market to companies trying to enroll into it. As the demand has increased so has the competition increased with each individual trying their best to trade with maximum profit possible. Hence trading with the available resources is not enough, what we need more is to be able to not just buy and sell profitable stocks but also to predict the future outcome this will help us to stay ahead in the game.

Stock market prediction is the act of trying to determine the future outcomes of the company's stock or other financial memos traded on an exchange. Stock market is an economic asset for a company and plays a major role in the growth and development of the industry and the commences of the native country that eventually prospers the economy of the company. An accurate prediction of future prices can lead to a higher yield of profit for investors.

The project aims to predict the actual and closing price of any stock along with the actual and predicted graph. The proposed system tries to find the accurate value of the next day closing value. The Long Short Term Memory (LSTM) is an artificial neural network that is based on a Machine Learning Algorithm and is in the field of Deep Learning. LSMT is an advanced neural network with a memory cell that stores a small amount of data for future references. LSMT is a feedback link that makes it a 'General Purpose Computer'. LSMT algorithm is more suitable in predicting the future stock prices than the Super Vector Machine (SVM) algorithm and Back Propagation algorithm as it filters unnecessary data unlike the other two. The time and memory requirements are also reduced and the algorithm is more proper in handling non-linear data.

Frontend:

Steamlit:

Steamlit software is used to turn data scripts into shareable web apps. Its based on python and is an open source software.

Backend:

Python:

Python is a high-level, general-purpose programming language which supports multiple programming paradigms like structed, object oriented and functional programming.

Software Requirements:

Windows 10 OS having Web Browser.

Hardware Requirements:

Processor: AMD A6-4455M APU 2.10GHz or above.

Hard-disk space: 20GB or above.

Display: Color Monitor.

Memory Requirements:

128MB of RAM.

Reasons for Selecting this Topic:

Everyone aspires to become wealthy with minimum efforts possible and great advantages. Similarly, we want to have a look into our future with a desire to have a good life where we can possibly avoid any chances of taking risks or at least try to reduce the risks. Stock market is a place of buying and selling of stocks in exchange of funds but, it also comes with its own risks. This project aims at audience that are willing to invest with an intention that the outcome with yield minimum risk possible. Below are few reasons behind the making of this project:

1. **Minimum Risk Factor:**
   This project aims to provide maximum profit with minimum casualty. This is achieved by predicting the future outcome. Many organizations provide a stock exchange platform for the investors but simply depend on the current data. That's where this project outstands as it not just helps in exchange but also predicts the future outcomes which eventually help the investors to take a right decision.

2. **Requirements**:
   The Project is based on a Machine Learning Algorithm called the Long Short Term Memory (LSTM) algorithm. This algorithm requires less time to implement and also requires memory allocation as compared to other algorithms.

3. **Prediction based on Live Data:**
   Prediction can be done in various ways but the most common way is to take the past data analyze it and predict the future outcome. This is a good method when predicting the likelihood of a thing but is not very convenient and reliable when it comes to stock predictions as actions of the past don't greatly affect the future rather the present action does. Hence the project focus on the prediction of stocks based on live i.e. current or present data.

4. **Thinking About the Future:**
   The main functionlality of the project is based on a Machine Learning Algorithm (LSTM Algorithm). This is an Artificial Neural Network based algorithm that works on the concept of Feedforward Link. An important reason behind selecting this algorithm is that it acts as a link between Machine Learning and Deep Learning. With enhancement in the functionality the same system can implement Deep Learning concept with not just prediction but also accuracy of the predictions.

Scope:

This project focuses on maximizing the investor's profit by predicting the future outcomes. It can benefit the economy of a company and can help users make wise investment choices. It can be applied to various fields like finance, stock market, small/large businesses, personal use, etc.

The goal of stock prediction is to help organizations deliver maximum business value. Periodization and managing resource capacity help us to make 'the most effective use of human resources'. The project is based on a machine learning algorithm that is used as a feedforward link. The same artificial neural network can be used in various different domains to predict the future outcome like

1. **Health Care Facility:**
   A patient's response to a future procedure can be predicted based on the history of the patient.

2. **Traffic Controller:**
   Past as well as present data can be used to predict the traffic on a road on a particular day and a particular time. The same can be applied for Air Traffic Controlling.

3. **Event Sourcing:**
   The probability and the number of crowed that attend a particular event can roughly be calculated.

4. **Weather Forecasting:**
   The machine learning algorithm can best predict the weather if provided with accurate past and present data.

CHAPTER 1: INTRODUCTION

1.1 Background:

Stock Market is a shared place where public trading is done on listed companies. The primary market is where general public view company shares in an initial public offering (IPO) to raise capital. The secondary market is where actual stock exchange happens. These stock exchanges are regulated by the Security and Exchange Board of India (SEBI).

In current times the demand for buying and selling of stocks and also the general interest of public in stock marketing has increased tremendously with each new company investing into stock market to companies trying to enroll into it. As the demand has increased so has the competition increased with each individual trying their best to trade with maximum profit possible. Hence trading with the available resources is not enough, what we need more is to be able to not just buy and sell profitable stocks but also to predict the future outcome this will help us to stay ahead in the game.

Stock market prediction is the act of trying to determine the future outcomes of the company's stock or other financial memos traded on an exchange. Stock market is an economic asset for a company and plays a major role in the growth and development of the industry and the commences of the native country that eventually prospers the economy of the company. An accurate prediction of future prices can lead to a higher yield of profit for investors.

The project aims to predict the actual and closing price of any stock along with the actual and predicted graph. The proposed system tries to find the accurate value of the next day closing value. The Long Short Term Memory (LSTM) is an artificial neural network that is based on a Machine Learning Algorithm and is in the field of Deep Learning. LSMT is an advanced neural network with a memory cell that stores a small amount of data for future references. LSMT is a feedback link that makes it a 'General Purpose Computer'. LSMT algorithm is more suitable in predicting the future stock prices than the Super Vector Machine (SVM) algorithm and Back Propagation algorithm as it filters unnecessary data unlike the other two. The time and memory requirements are also reduced and the algorithm is more proper in handling non-linear data.

1.2 Objectives:

Following are the objective about the project:

- The main objective behind making this project is to ease the user in handling and provide efficient decision making during a stock exchange. The project is based on Machine Learning Artificial Neural Network Back Propagation Feedforward Link Algorithm. This algorithm is called the Long Short Term Memory (LSTM) Algorithm.
- The LSTM is a advanced neural network having a memory cell that stores a small amount of data for future reference. LSTM has a feedback link that makes it a 'general purpose computer'. LSTM can also process an entire series of data not only a single value like image. The dropout process that takes place in LSTM algorithm makes it comparatively faster than the SVM Algorithm (Support Vector Machine) and Back Propagation Algorithm.
- As a result, LSTM algorithm is also more efficient in predicting the future stock than the other two algorithms. This is because it removes undesired data unlike other algorithms.
- The model is deployed on a python based software called 'Streamlit'. Streamlit cloud is an open-source app framework for machine learning and data science.
- The project is based on the Software-as-a-Service (SaaS) module of Cloud Computing. Cloud is a well known and a popular choice of developers for hosting projects. Cloud provides a variety of functionality and also proves to be a great platform for hosting. Cloud is also a very trusted platform as it facilities proper safety and security to the project.
- Along with this it also provides two more modules namely, Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) which enables a user to add on more modules to the project as and when required. Hosting this project on cloud gives an additional benefit of access across all platforms globally.
- The project also implements live graphs to show the progression of the stock market. This help the user to visually see the changes and hence better understand them rather than just displaying it as a grid.
- System Prediction not only gives a glance into future actions but also help make better and appropriate decision choices at an individual scale or for a company.
- The same algorithm can be modified to fit into various other domains like Health care facility, Traffic controlling, Event management, etc.

1.3 Purpose and Scope:

1.3.1 Purpose:

Everyone aspires to become wealthy with minimum efforts possible and great advantages. Similarly, we want to have a look into our future with a desire to have a good life where we can possibly avoid any chances of taking risks or at least try to reduce the risks. Stock market is a place of buying and selling of stocks in exchange of funds but, it also comes with its own risks. This project aims at audience that are willing to invest with an intention that the outcome with yield minimum risk possible. Below are few reasons behind the making of this project:

1. **Minimum Risk Factor:**
   This project aims to provide maximum profit with minimum casualty. This is achieved by predicting the future outcome. Many organizations provide a stock exchange platform for the investors but simply depend on the current data. That's where this project outstands as it not just helps in exchange but also predicts the future outcomes which eventually help the investors to take a right decision.

2. **Requirements**:
   The Project is based on a Machine Learning Algorithm called the Long Short Term Memory (LSTM) algorithm. This algorithm requires less time to implement and also requires memory allocation as compared to other algorithms.

3. **Prediction based on Live Data:**
   Prediction can be done in various ways but the most common way is to take the past data analyze it and predict the future outcome. This is a good method when predicting the likelihood of a thing but is not very convenient and reliable when it comes to stock predictions as actions of the past don't greatly affect the future rather the present action does. Hence the project focus on the prediction of stocks based on live i.e. current or present data.

4. **Thinking About the Future:**

   The main functionality of the project is based on a Machine Learning Algorithm (LSTM Algorithm). This is an Artificial Neural Network based algorithm that works on the concept of Feedforward Link. An important reason behind selecting this algorithm is that it acts as a link between Machine Learning and Deep Learning. With enhancement in the functionality the same system can implement Deep Learning concept with not just prediction but also accuracy of the predictions.

1.3.2 Scope:

This project focuses on maximizing the investor's profit by predicting the future outcomes. It can benefit the economy of a company and can help users make wise investment choices. It can be applied to various fields like finance, stock market, small/large businesses, personal use, etc.

The goal of stock prediction is to help organizations deliver maximum business value. Periodization and managing resource capacity help us to make 'the most effective use of human resources'. The project is based on a machine learning algorithm that is used as a feedforward link. The same artificial neural network can be used in various different domains to predict the future outcome like

1. **Health Care Facility:**
   A patient's response to a future procedure can be predicted based on the history of the patient.

2. **Traffic Controller:**
   Past as well as present data can be used to predict the traffic on a road on a particular day and a particular time. The same can be applied for Air Traffic Controlling.

3. **Event Sourcing:**
   The probability and the number of crowed that attend a particular event can roughly be calculated.

4. **Weather Forecasting:**
   The machine learning algorithm can best predict the weather if provided with accurate past and present data.

CHAPTER 2: SYSTEM ANALYSIS

2.1 Existing System:

Stock Exchange has always been a fascination for many people. It was a widely spread platform for money dealing starting from the 80's and is still very popular, besides it has gained even more crowd in the recent decade. The process of checking the Sensex was never this smooth though. In its initial time people had to manual check the Sensex be visiting the Stock Exchange office, standing in crowed for a long time just to get the gleams of the stock. But thanks to the ever-growing technology gone are those days of physically checking the Sensex, nowadays all this can be done virtually and even at the comfort of house very easily. All this is because of the use of AI technology.

Artificial Intelligence (AI) has gained a high pace in the past few years with introduction of various AI based software that are launched into the market. The purpose is to simplify our daily tasks with minimum resources possible and with the maximum output. As AI is implemented into majorly every domain of work so has the need of using advanced AI that is Machine Learning (ML) increased. This is because both Artificial Intelligence and Machine Learning complement each other. Machine Learning is a sub-domain of a bigger umbrella called Deep Learning.

Machine Learning gives systems the capability to learn without being explicitly programmed. Gone are those days when each and every feature or function needed to be physically coded into the systems. Modern systems are designed in a way that they self-learn things and need not need to be hard coded. This is all thanks to the introduction of AI and Machine Learning.

A system learns when it is provided with relevant data. This data can be anything ranging from facts, values, images, sound, texts, and even unprocessed data. Data is a very important part of AI, Machine Learning, Data Analytics. Without relevant data we will not be able to train our model. Data can be of primarily two types: Processed Data and Unprocessed Data.

There are various applications of how machine learning is implemented into modern concepts like, Movie Recommendation System, Heart Disease Prediction System, Traffic Management System, Crop Classification System, Product Quality Prediction System, etc. All these examples and many more are all based on the concept of machine learning and use various algorithm in its implementation. The algorithms mostly include clustering algorithm for unsupervised learning. Clustering is the method of dividing the data points into groups or clusters such that the data point in each cluster are similar to the other data points in the same cluster.

One of the simplest clustering algorithms is the K-Means Clustering Algorithm. It partitions 'n' observations into 'k' clusters where each observation belongs to the nearest mean. This algorithm is used in marketing to characterize and discover marketing segments. It is also used in Biology for classification of different species, in libraries for classification and clustering of different books based on their type, etc. Apart from K-Means there are various

other clustering algorithm used such as: Centroid based clustering, Connectivity based clustering, Density based clustering.

All the algorithms are possible because of the data that they are given. For prediction-based systems it's very important to wisely choose data as the data itself we result into prediction. Its important to process the data appropriately if using unprocessed data as even a little discrepancy in the data with be alter the outcome of the prediction.

Most of the algorithms either used past data, some used present data along with the past data to make predictions. A very important issue that the algorithm faces in such situation is to constantly keep the data updated and to clean is from time to time. This all can be done with the use of various other software's but the it just increased the memory usage and creates a burden on the system itself. A Simple and convenient way to go about is to use an algorithm that does the process of cleaning the data and also takes less space.

2.2 Proposed System:

Stock market prediction is the act of trying to determine the future outcomes of the company's stock or other financial memos traded on an exchange. Stock market is an economic asset for a company and plays a major role in the growth and development of the industry and the commences of the native country that eventually prospers the economy of the company. An accurate prediction of future prices can lead to a higher yield of profit for investors.

Stock price prediction is a challenging task owing to the complexity involved behind time series. Autoregressive Integrated Moving Average (ARIMA) and Backpropagation Neural Network (BPNN) are popular linear and non-linear models for time series forecasting. The linear and non-linear patterns can be effectively captured using the integration of these two models.

Prediction and modelling of the financial indices is a very challenging and demanding problem because its dynamic, noisy and multivariate in nature. Modern approaches have also to challenge the fact that there are dependencies between different global financial indices. All this complexity in combination with the large volume of historic financial data raises the need for advanced machine learning solutions to the problem. This article proposes a Deep Learning approach utilizing Long Short-Term Memory (LSTM) Networks for the modelling and trading of financial indices. Along with the usage of the appropriate algorithm the system also combines the usage of not the past or present data but also dynamic data/live data i.e. data that is generated on basis of ongoing events.

Experiments has shown that Convolutional Neural Network (CNN) can be better than Recurrent Neural Networks (RNN) on catching semantic from texts and RNN is better on catching the context information and modeling complex temporal characteristics for stock market forecasting. So, there are two models compared in this paper: a hybrid model composed by a CNN for the financial news and a Long Short-Term Memory (LSTM) for technical indicators, named as SI-RCNN; and a LSTM network only for technical indicators, named as I- RNN. The output of each model is used as input for a trading agent that buys stocks on the current day and sells the next day when the model predicts that the price is going up, otherwise the agent sells stocks on the current day and buys the next day. The proposed method shows a major role of financial news in stabilizing the results and almost no improvement when comparing different sets of technical indicators.

Long Short Term Memory (LSTM) is a kind of Recurrent Neural Network (RNN) wherein the output of the last step is feed as input to the current step. It is mainly used for classification, processing and predicting on the basis of time series data. LSTM has three gates that manipulate information:

1. Forget Gate:
   The Forget Gate does the task of discarding the information that is no longer useful.
2. Input Gate:
   Once the information is filtered and ready to use the information is then passed on through the Input Gate.
3. Output Gate:
   The Output Gate does the task of extracting useful information from the current state to be presented.

The working of the algorithm can be briefly described in the following steps:

Input: Stock Price Data.

Output: Prediction of stock prices based on stock price variations.

Step 1: Stock Data is taken and stored in a numpy array of 3 dimensions.

Step 2: A network structure is built with [1, a, b] dimensions where there is 1 input layer, a neurons in the next layer, b neurons in the subsequent layers and a single layer with a linear activation function.

Step 3: Train the constructed network on the data.

Step 4: Use the output of the last layer as prediction in the next time step

Step 5: Repeat steps 3 and 4 until optimal convergence is reached

Step 6: Obtain predictions by providing test data as input to the network.

Step 6. Evaluate accuracy by providing test data as input to the network.

Step 7: End

2.3 Requirement Analysis:

Non-functional Requirements:

These are basically the quality constraints that the system must satisfy to make the project fully functional and accurate. The priority or extent to which these factors are implemented varies from one project to another. They are also called non-behavioral requirements.

- Time constraint is a very important aspect when making a dynamic system. The faster the system is able to process and retrieve the information the better. The processing of each request should be done within a specific time limit. In the case of this project its about 10 seconds.
- The site takes a few seconds more to predict close price depending on how large the client has entered the input.
- The system opts for providing better accuracy and optimized results.
- A very lowkey yet an important feature that gets neglected often times the systems easy to be use and its user-friendly behavior.

Specific Requirements:

User interfaces:

The use of a User Interface is very crucial for user-based systems as this is the face that will connect the users to the working of the system. Through this the users will be able to navigate the system and perform required actions. The user interface is also important as it hides all the unnecessary details related to the system that the user would otherwise not find very useful.

The new system provides a very intuitive and simple interface to the client as well as investor or the visitors, so that the user can easily navigate through software. The client can Check the stock's adjacent close, volume, etc. recorded. It can compare the close and predicted price of the date entered by the user. The user can check the flow of the market with the help of close price graph as well as predicted price graph.

2.4 Hardware Requirements:

CPU: Intel Pentium IV 600MHz or above.

Mother Board: Intel 810 or above.

Hard-disk space: 20GB or above.

Display: Color Monitor.

Memory: 128MB of RAM.

Other Devices:

Monitor screen – The software shall display information to the user via the monitor screen.

Mouse – The software is made interactive with the movement of the mouse and the mouse buttons.

The mouse shall activate areas for data input, command buttons and select options from the menu.

Keyboard – The software shall interact with the keystrokes of the keyboard.

The keyboard will input data information into the active area of the database.

2.5 Software Requirements:

System Type: 64-bit operating system, x64 based processor.

Intel Pentium IV 600MHz or above.

Windows 10 OS having Web Browser.

Front End: Streamlit

Back End: Python Connection

Web server: The actual program that will perform the operations is written in Python. All data will be stored in a database.

Communication Interfaces: The HTPP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

2.6 Justification of the Platform:

The project uses the following software platform for implementation:

Python:

Python is a high-level, general-purpose programming language which supports multiple programming paradigms like structed, object oriented and functional programming.

Python supports multiple programming languages. Python has various libraries that are used to implement functionality in the programs. Here is the list of libraries that will be required for this system:

Pandas Datareader: Used to create pandas DataFrame objects by using various data sources available through various platforms.

NumPy: NumPy stands for 'Numerical Python'. It supports large metrices and multi-dimensional data. It consists of built-in mathematical functions for computation.

Pandas: Pandas library provides flexible high-level data structures and numerous tools for Data Analysis.

Sklearn: This library is used for machine learning and statistical modeling of various algorithms like classification, regression, clustering.

Keras:

Matplotlib: Matplotlib is used for plotting numerical data. It plots high defined figure like pie chart, histogram, scatterplot, graphs, etc.

TensorFlow: This library is used for high-level computation. It is also used for implementation of machine learning and deep learning algorithms.

Streanlit:

Streamlit software is used to turn data scripts into shareable web apps. It is based on python and is an open source software.

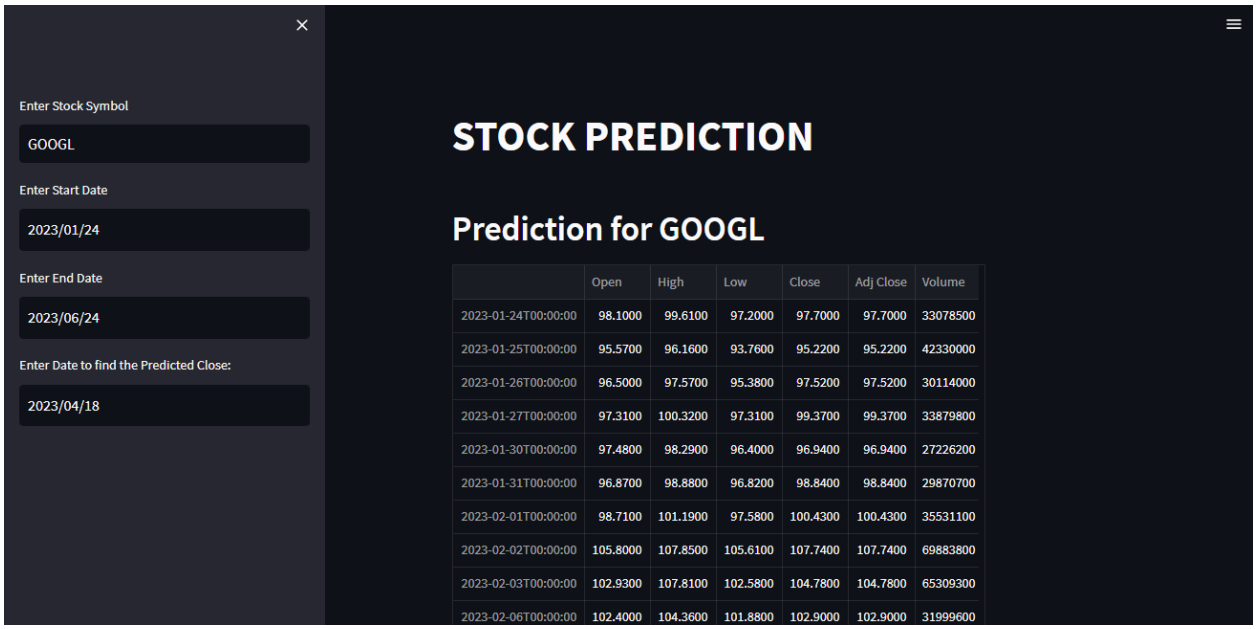
Cloud:

Cloud is a very popular platform used today for hosting multiple applications. It has gained demand in the past few years because of its easy access, robust nature and its safety features. Cloud hosting is platform independent and can be accessed from anywhere. It provides various services in the form of Platform-as-a-Service (PaaS), Infrastructure-as-a-Service (IaaS) and Software-as-a-Service (SaaS).

CHAPTER 3: SYSTEM DESIGN

3.1 Module Design:

3.1.1. User Interface:



	Open	High	Low	Close	Adj Close	Volume
2023-01-24T00:00:00	98.1000	99.6100	97.2000	97.7000	97.7000	33078500
2023-01-25T00:00:00	95.5700	96.1600	93.7600	95.2200	95.2200	42330000
2023-01-26T00:00:00	96.5000	97.5700	95.3800	97.5200	97.5200	30114000
2023-01-27T00:00:00	97.3100	100.3200	97.3100	99.3700	99.3700	33879800
2023-01-30T00:00:00	97.4800	98.2900	96.4000	96.9400	96.9400	27226200
2023-01-31T00:00:00	96.8700	98.8800	96.8200	98.8400	98.8400	29870700
2023-02-01T00:00:00	98.7100	101.1900	97.5800	100.4300	100.4300	35531100
2023-02-02T00:00:00	105.8000	107.8500	105.6100	107.7400	107.7400	69883800
2023-02-03T00:00:00	102.9300	107.8100	102.5800	104.7800	104.7800	65309300
2023-02-06T00:00:00	102.4000	104.3600	101.8800	102.9000	102.9000	31999600

Fig – 3.1: User Interface.

3.1.2. Entry Ticker for Stocks:

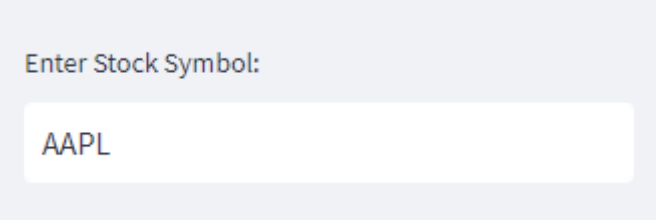


Fig – 3.2: Stock Entry.

3.1.3. Various Functions offered by Streamlit:

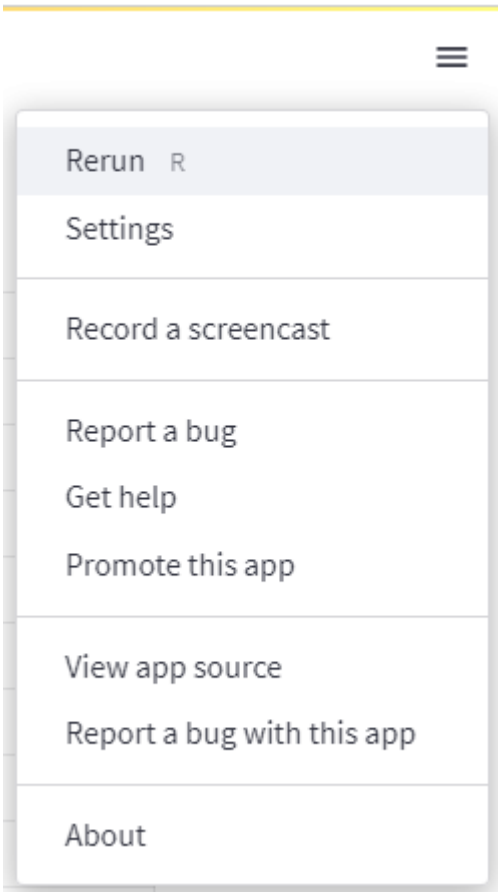


Fig – 3.3: Streamlit Functionality.

3.1.4. Calendar to enter the Start Date and End Date:

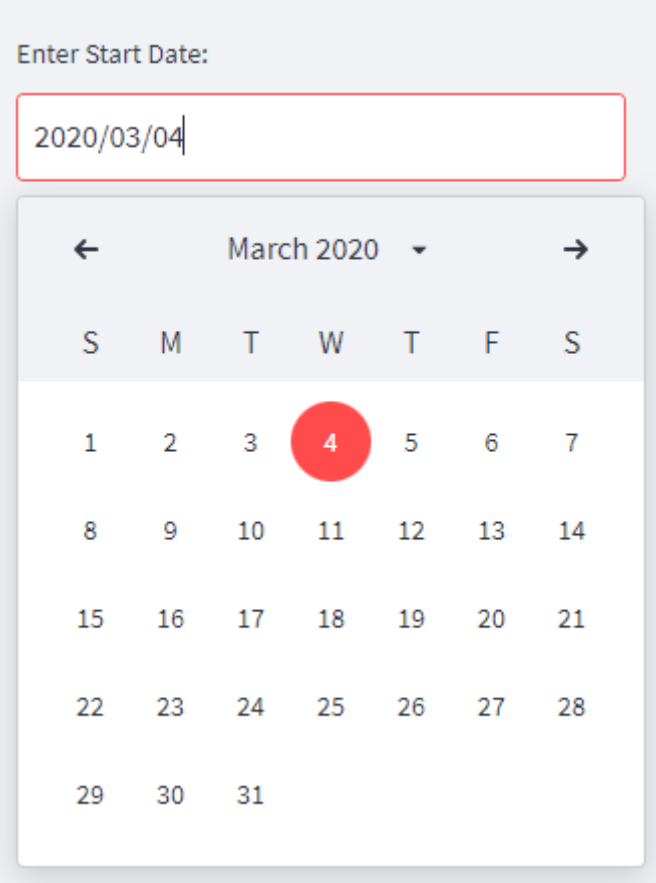


Fig – 3.4: Start Date and End Date:

3.1.5. Calendar to enter the Predicted Close:

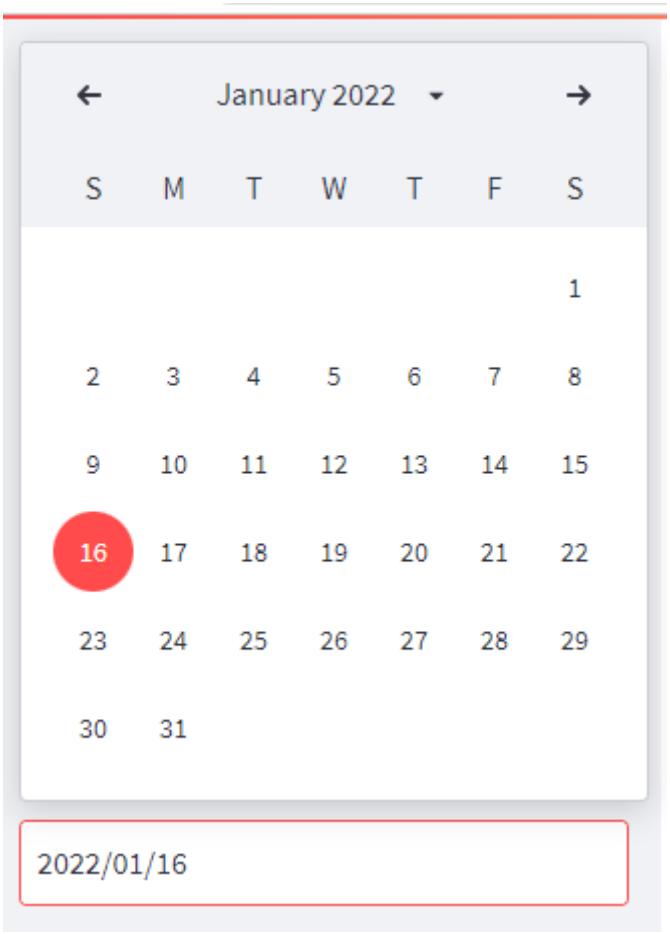
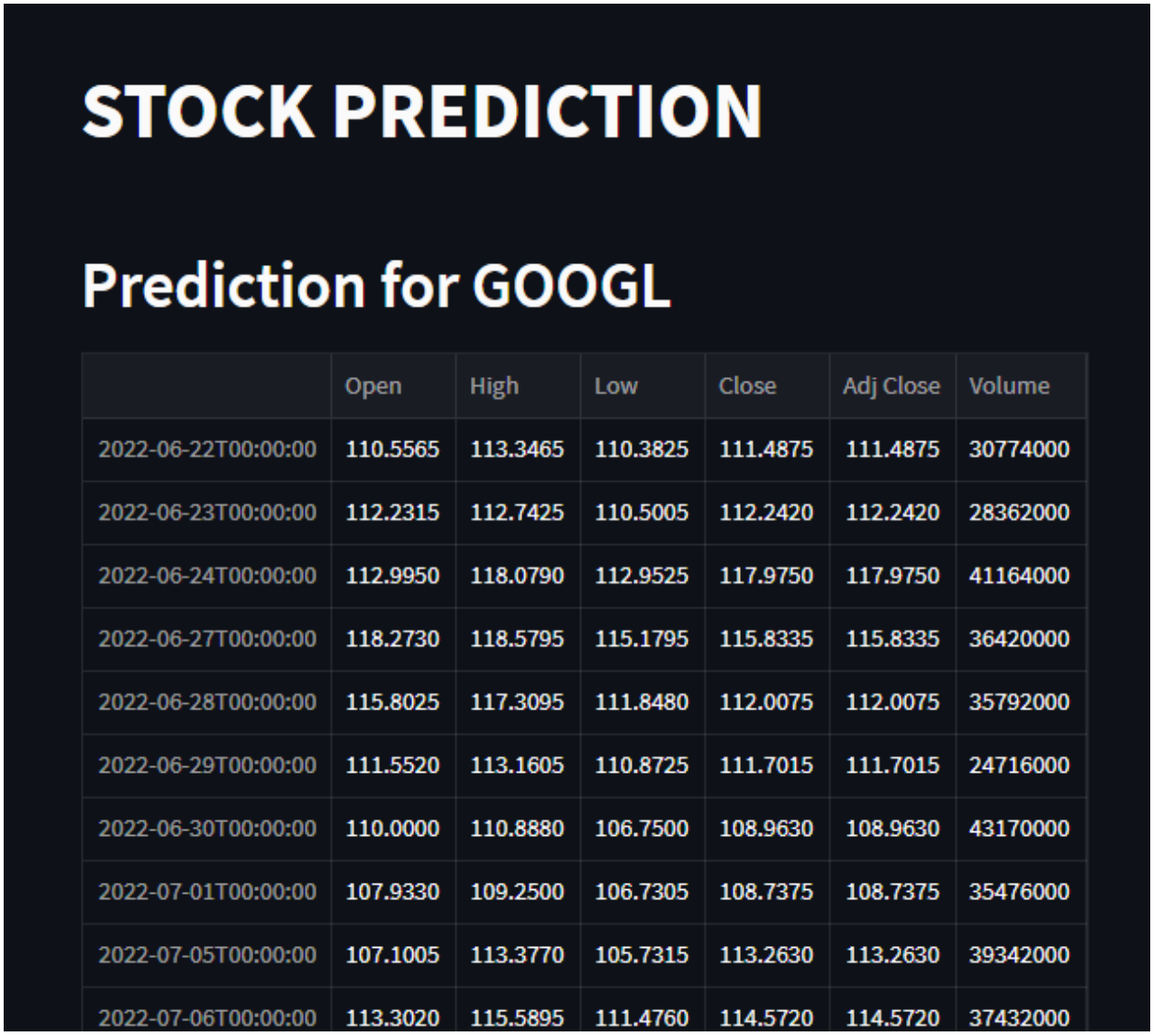


Fig – 3.5: Predicted Close.

3.1.6. Closing Price Table for selected Company:



	Open	High	Low	Close	Adj Close	Volume
2022-06-22T00:00:00	110.5565	113.3465	110.3825	111.4875	111.4875	30774000
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000

Fig – 3.6: Close Price Table.

3.1.7. Closing Graph:

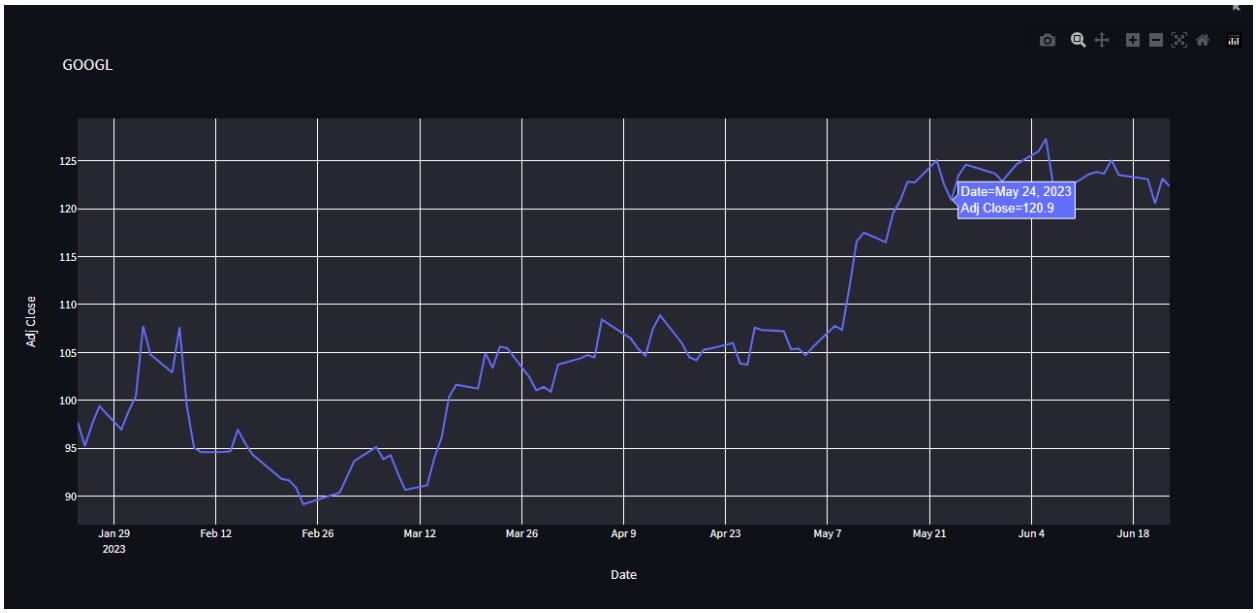


Fig – 3.7: Closing Graph.

3.1.8. Closing and Prediction Graph:

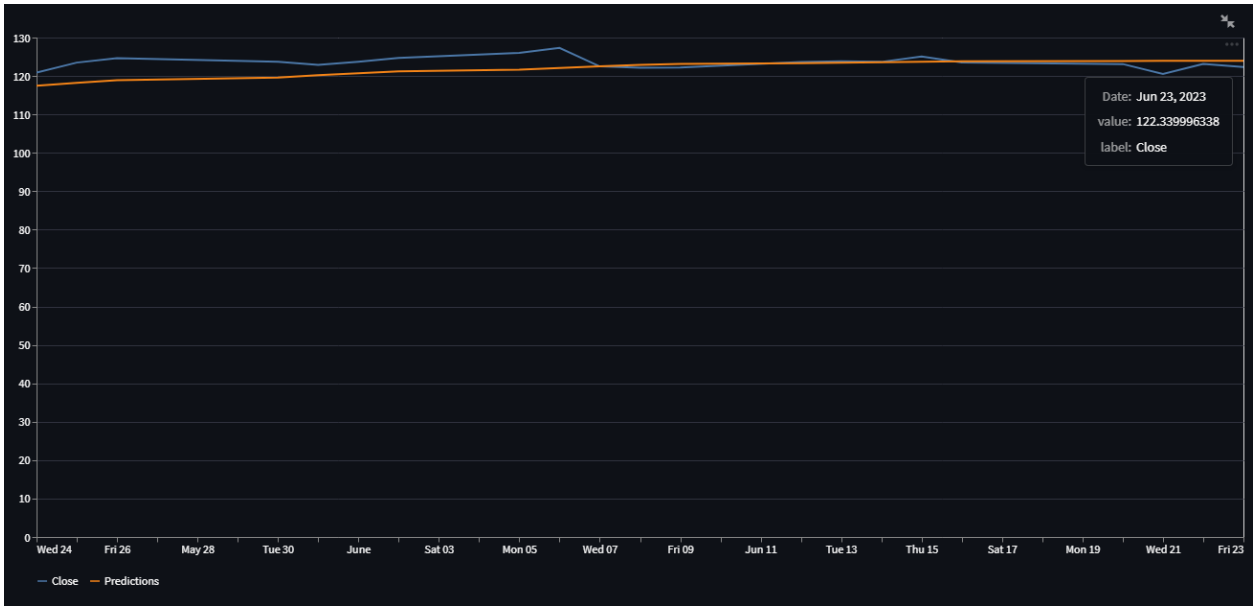


Fig – 3.8: Closing and Prediction Graph.

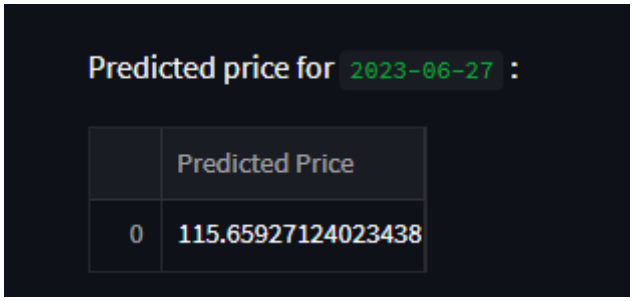
3.1.9. Close Price and Predicted Close Table:

Close Price and Predicted Close

	Open	High	Low	Close	Adj Close	Volume	Predictions
2023-04-11T00:00:00	106.5500	106.7300	104.6800	105.3500	105.3500	26311800	103.14820098876953
2023-04-12T00:00:00	106.5800	106.7500	104.3400	104.6400	104.6400	24370300	103.40604400634766
2023-04-13T00:00:00	105.8400	107.4900	105.8400	107.4300	107.4300	24843600	103.51271057128906
2023-04-14T00:00:00	106.8900	108.9400	106.8400	108.8700	108.8700	26578000	103.7257080078125
2023-04-17T00:00:00	104.6600	106.1600	104.5200	105.9700	105.9700	37571200	104.07794189453125
2023-04-18T00:00:00	106.4900	106.5400	104.0700	104.5000	104.5000	26596400	104.2590560913086
2023-04-19T00:00:00	103.5800	104.9800	103.0700	104.1800	104.1800	20905700	104.22032165527344
2023-04-20T00:00:00	103.9100	106.2500	103.8700	105.2900	105.2900	27820800	104.03350067138672
2023-04-21T00:00:00	105.4700	106.0000	104.7800	105.4100	105.4100	25800100	103.85714721679688
2023-04-24T00:00:00	105.4900	106.6300	104.7000	105.9700	105.9700	23542800	103.70703887939453

Fig – 3.9: Close Price and Predicted Close Table.

3.1.10. Predicted Price:

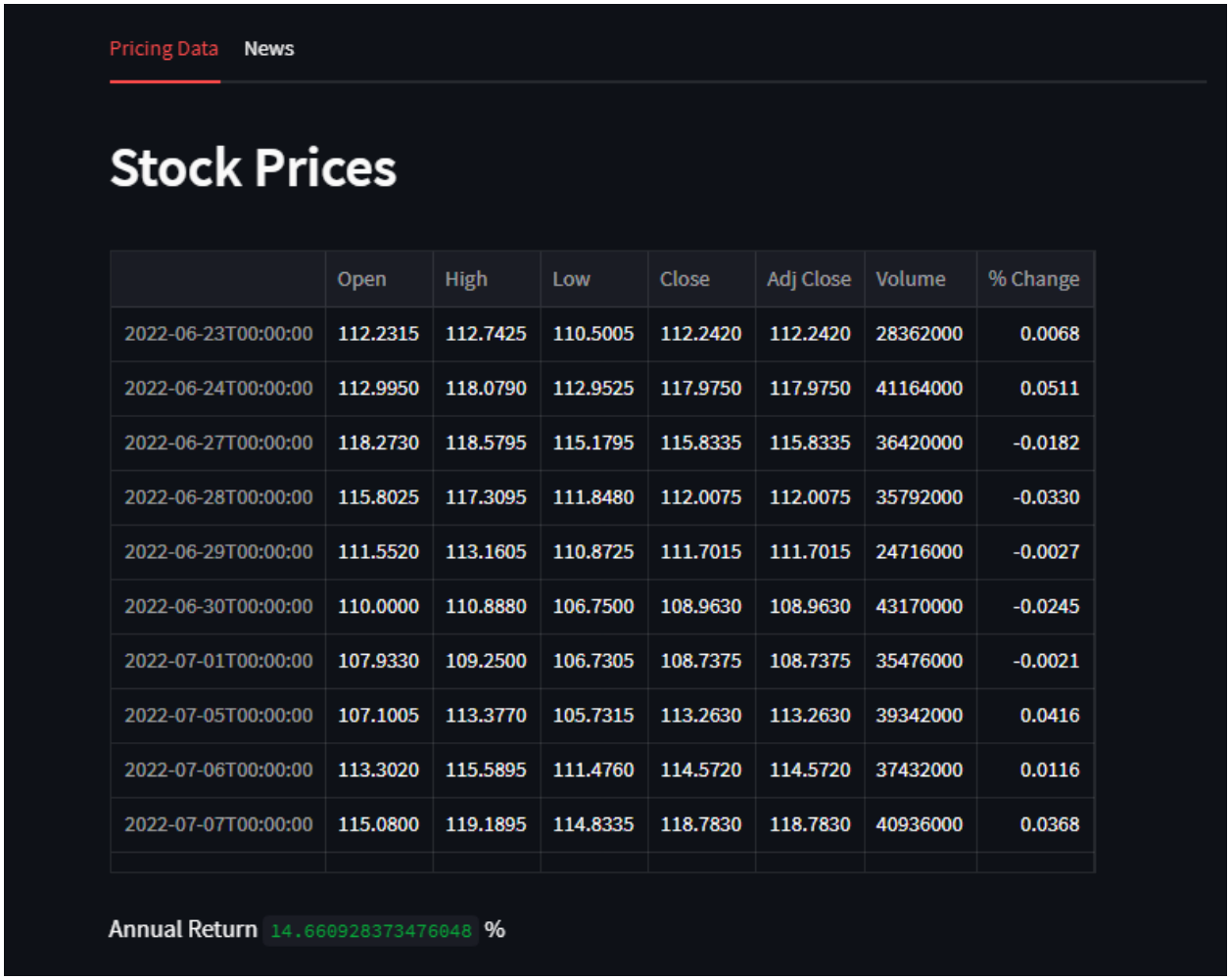


Predicted price for 2023-06-27 :

	Predicted Price
0	115.65927124023438

Fig – 3.10: Predicted Price.

3.1.11. Pricing Data



	Open	High	Low	Close	Adj Close	Volume	% Change
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000	0.0068
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000	0.0511
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000	-0.0182
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000	-0.0330
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000	-0.0027
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000	-0.0245
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000	-0.0021
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000	0.0416
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000	0.0116
2022-07-07T00:00:00	115.0800	119.1895	114.8335	118.7830	118.7830	40936000	0.0368

Fig – 3.11: Pricing Data

3.1.12. Top 10 News

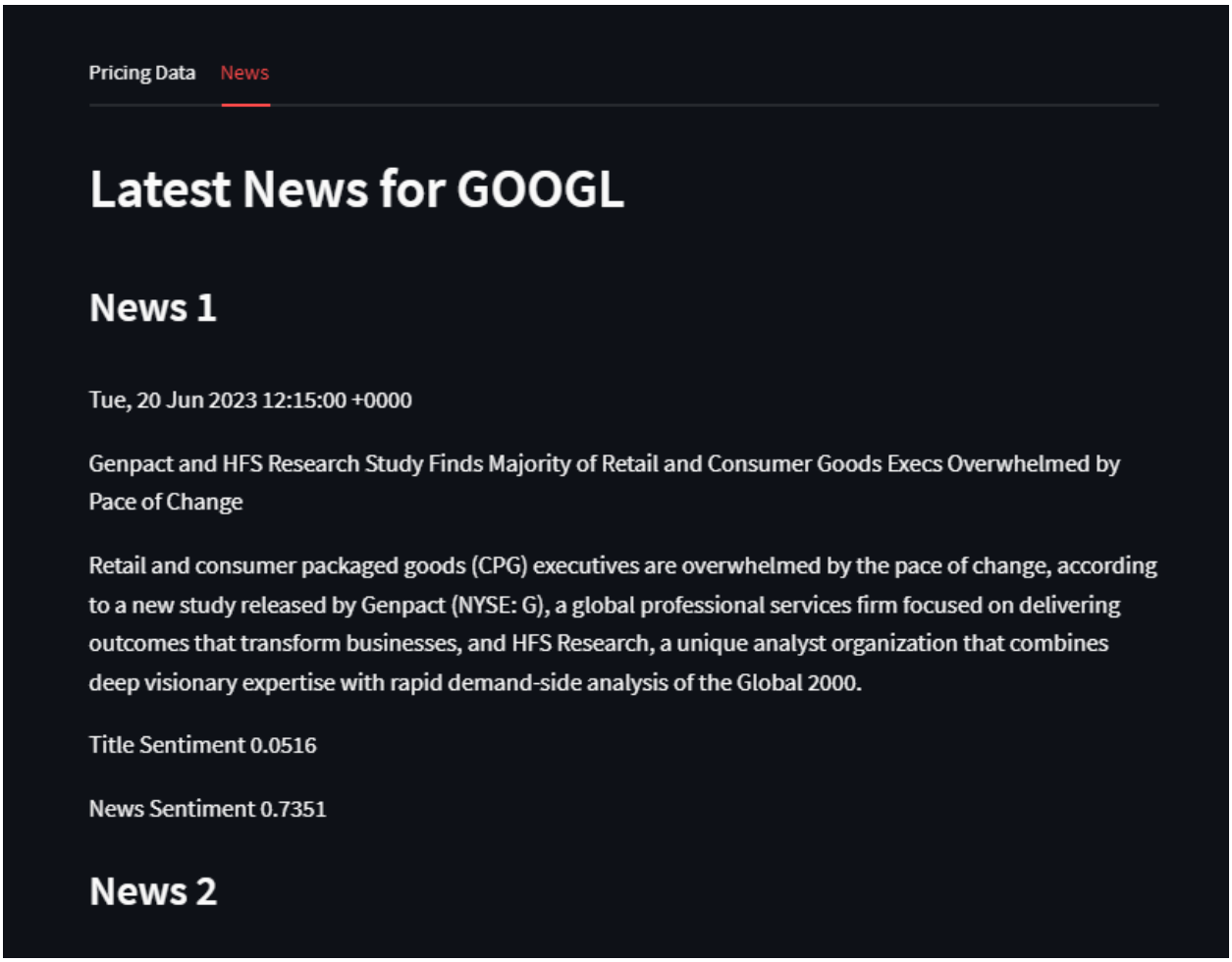


Fig – 3.12: Top 10 News

3.2 Data Dictionary:

Sr. No.	Abbreviations	Expended Form
1.	AI	Artificial Intelligence
2.	ML	Machine Learning
3.	DL	Deep Learning
4.	LSTM Algorithm	Long Short Term Memory Algorithm.
7.	ANN	Artificial Neural Network.
8.	GUI	Graphical User Interface.
9.	MVO	Mean Variation Optimization.
10.	CNN	Convolution Neural Network.
11.	RMSE	Root Mean Square Error.
12.	API	Application Programming Interface.

Table: 3.1

3.3 UML Diagrams:

3.3.1 Activity Diagram:

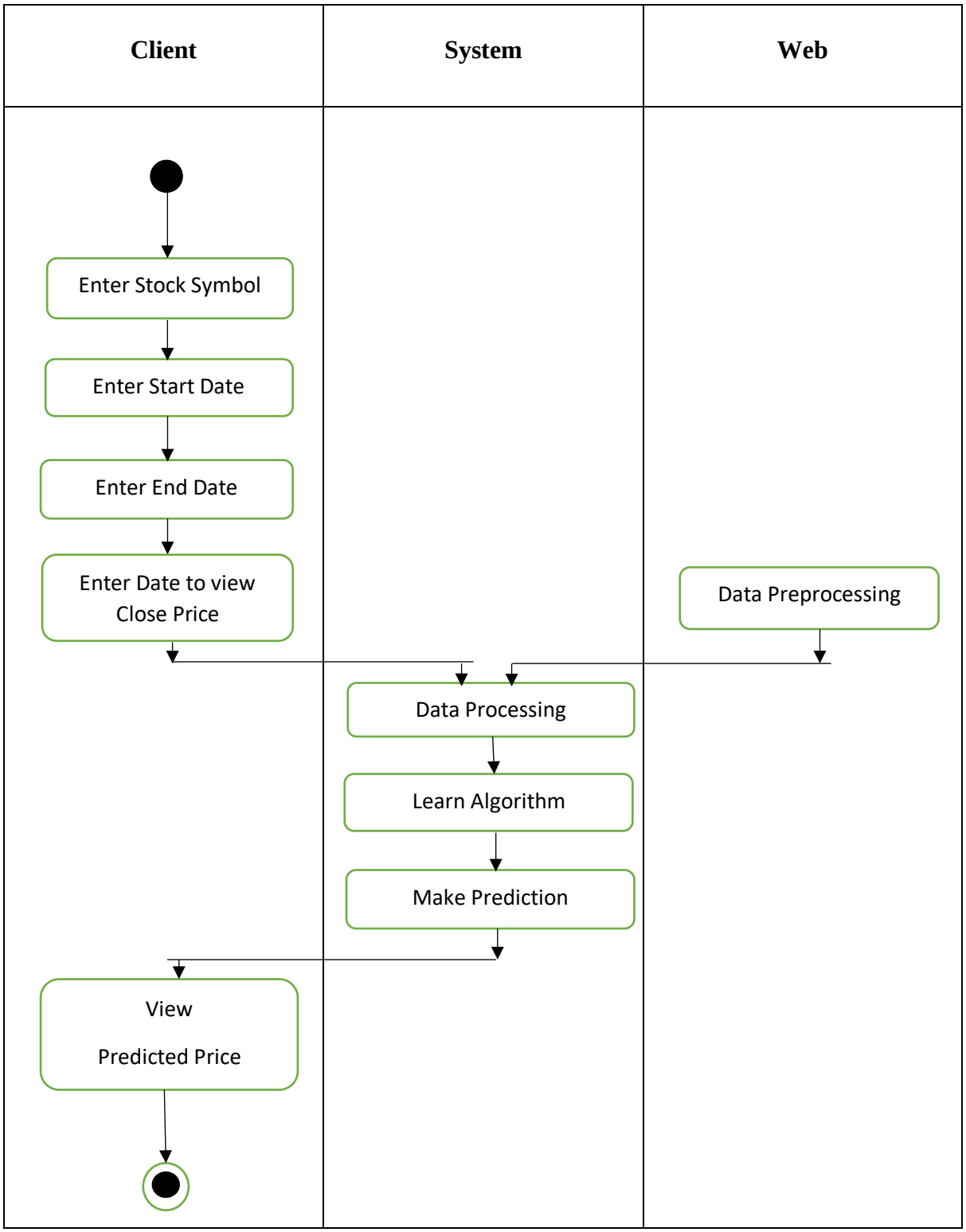


Fig 3.13: Activity Diagram

3.3.2 Class Diagram:

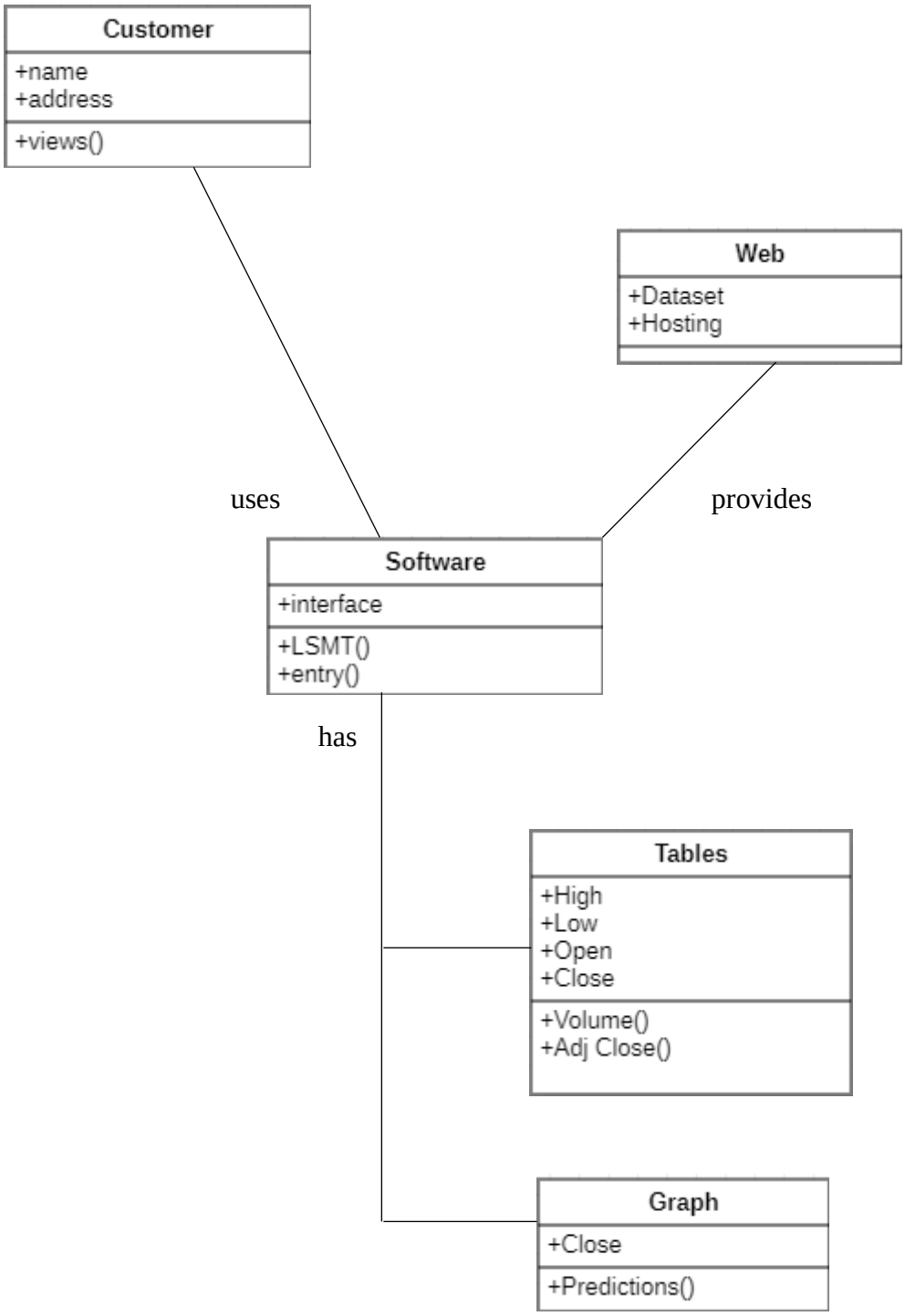


Fig 3.14: Class Diagram

3.3.3 Collaboration Diagram:

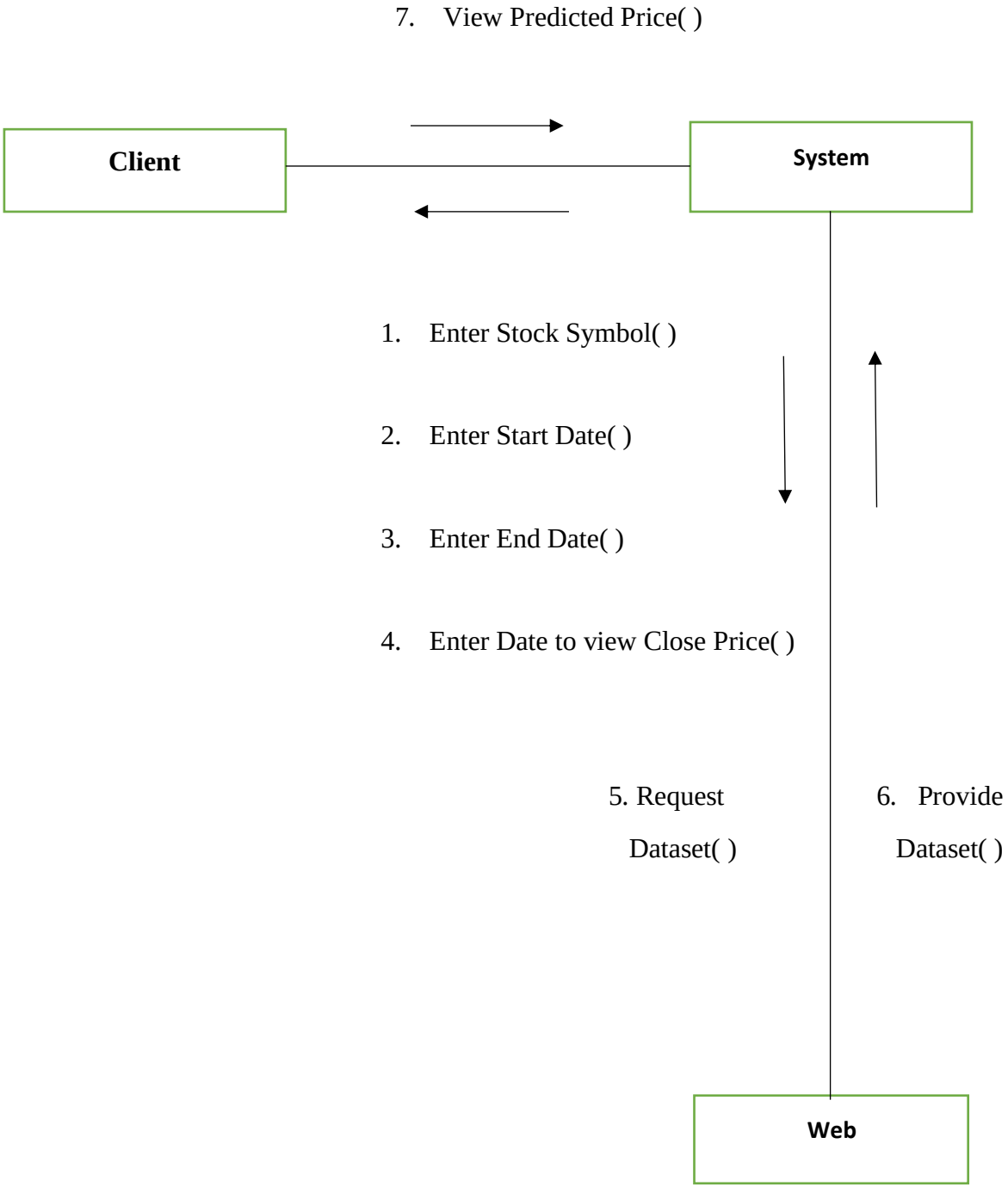


Fig 3.15: Collaborative Diagram

3.3.4 Component Diagram:

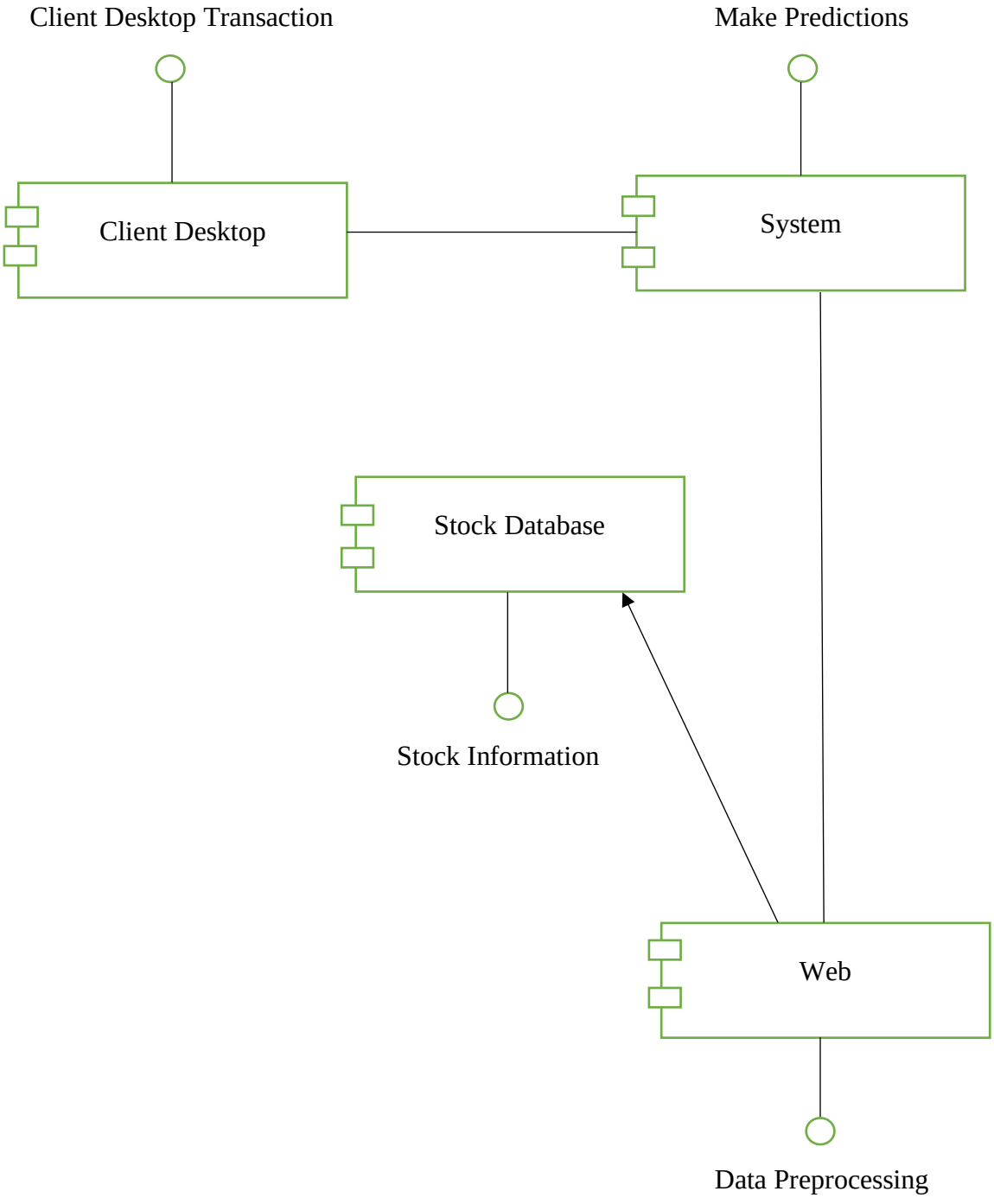


Fig 3.16: Component Diagram

3.3.5 Data Flow Diagram:

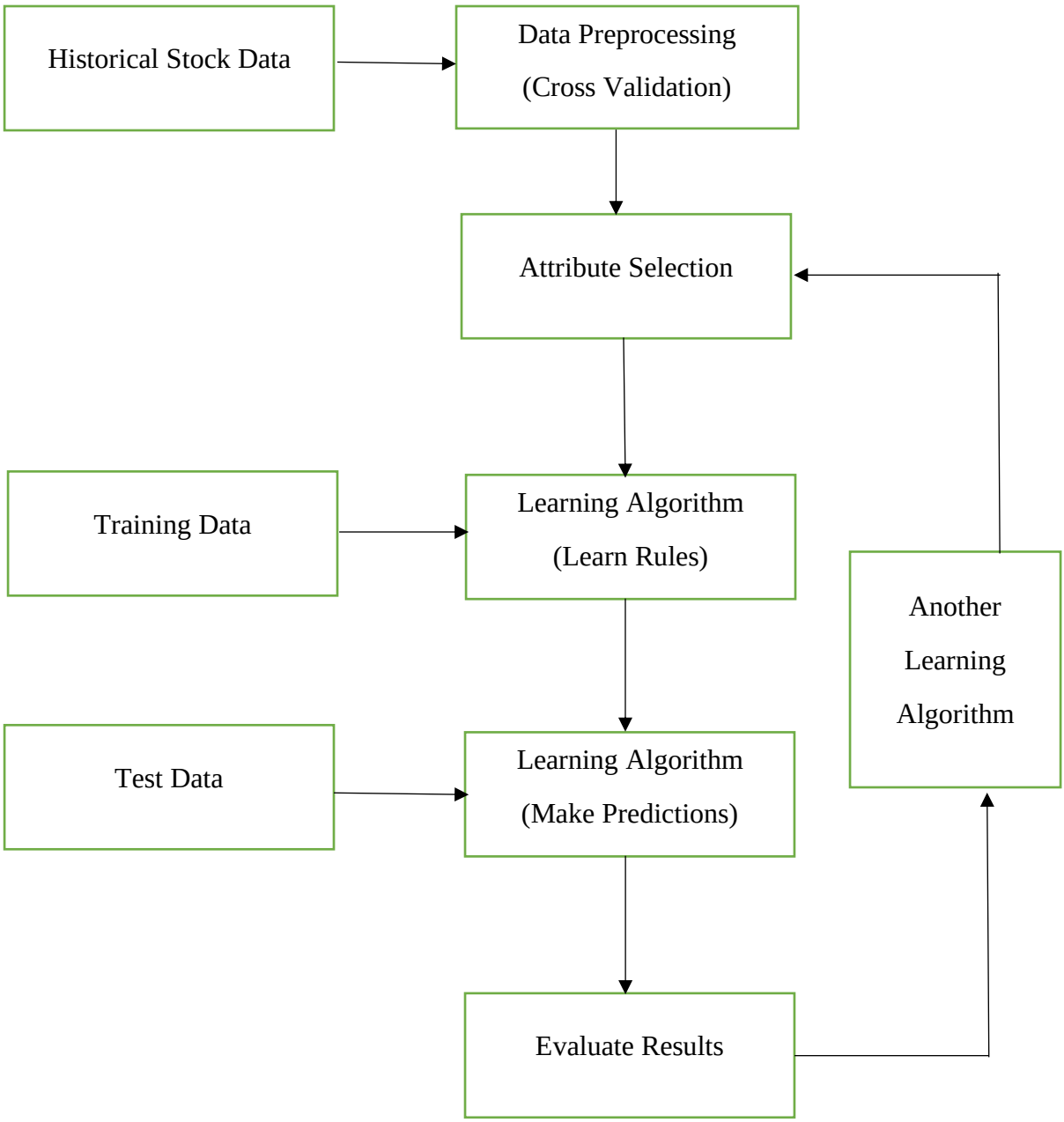


Fig 3.17: Data Flow Diagram

3.3.6 Deployment Diagram:

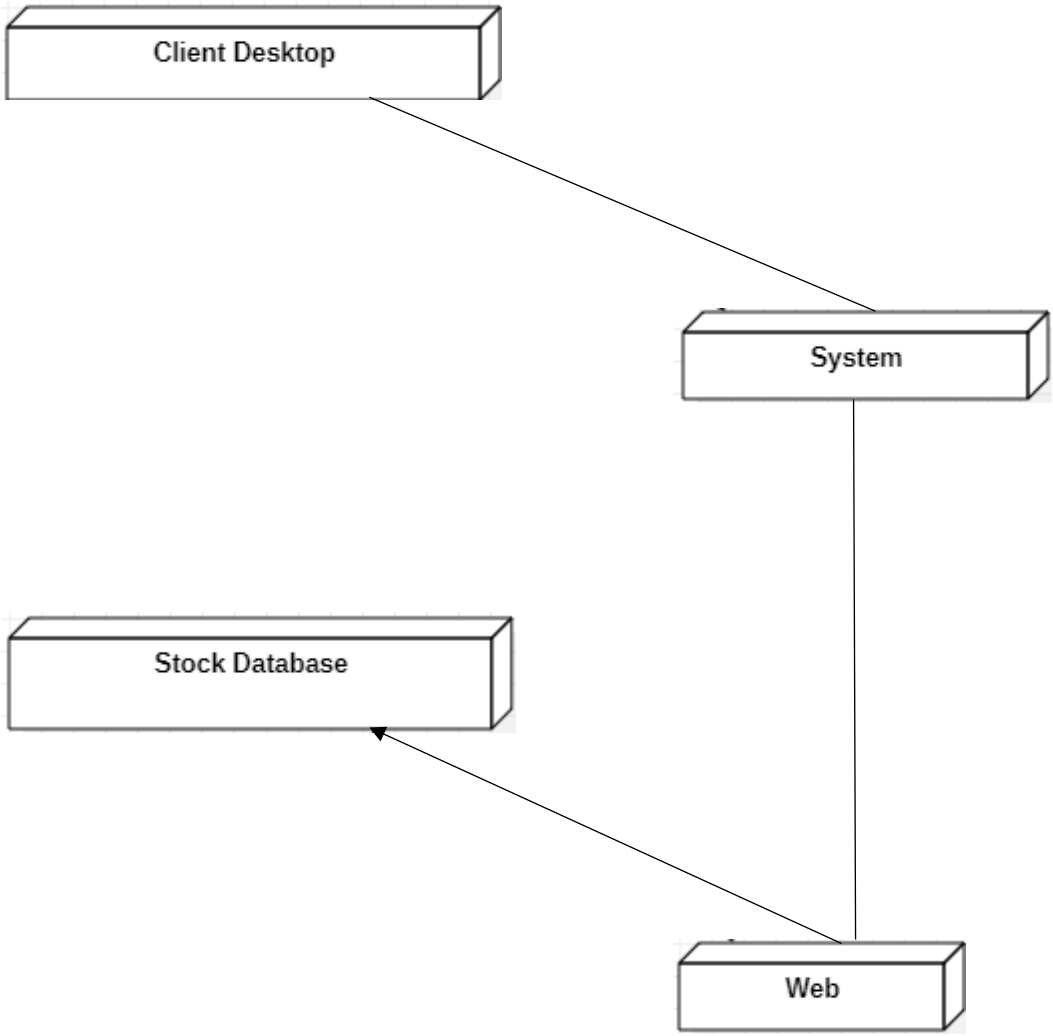


Fig 3.18: Deployment Diagram

3.3.7 ER Diagram:

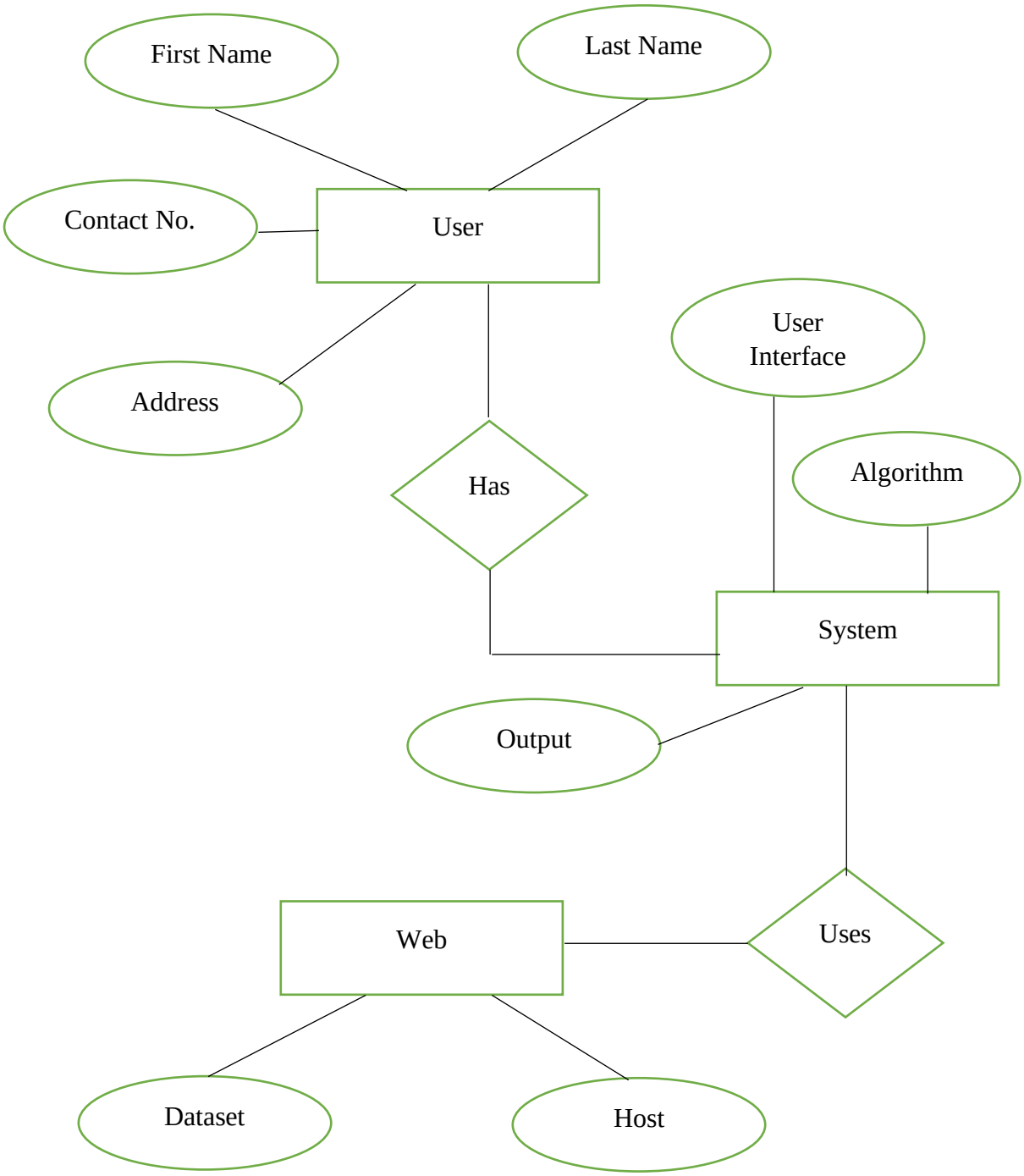


Fig 3.19: ER Diagram

3.3.8 Sequence Diagram:

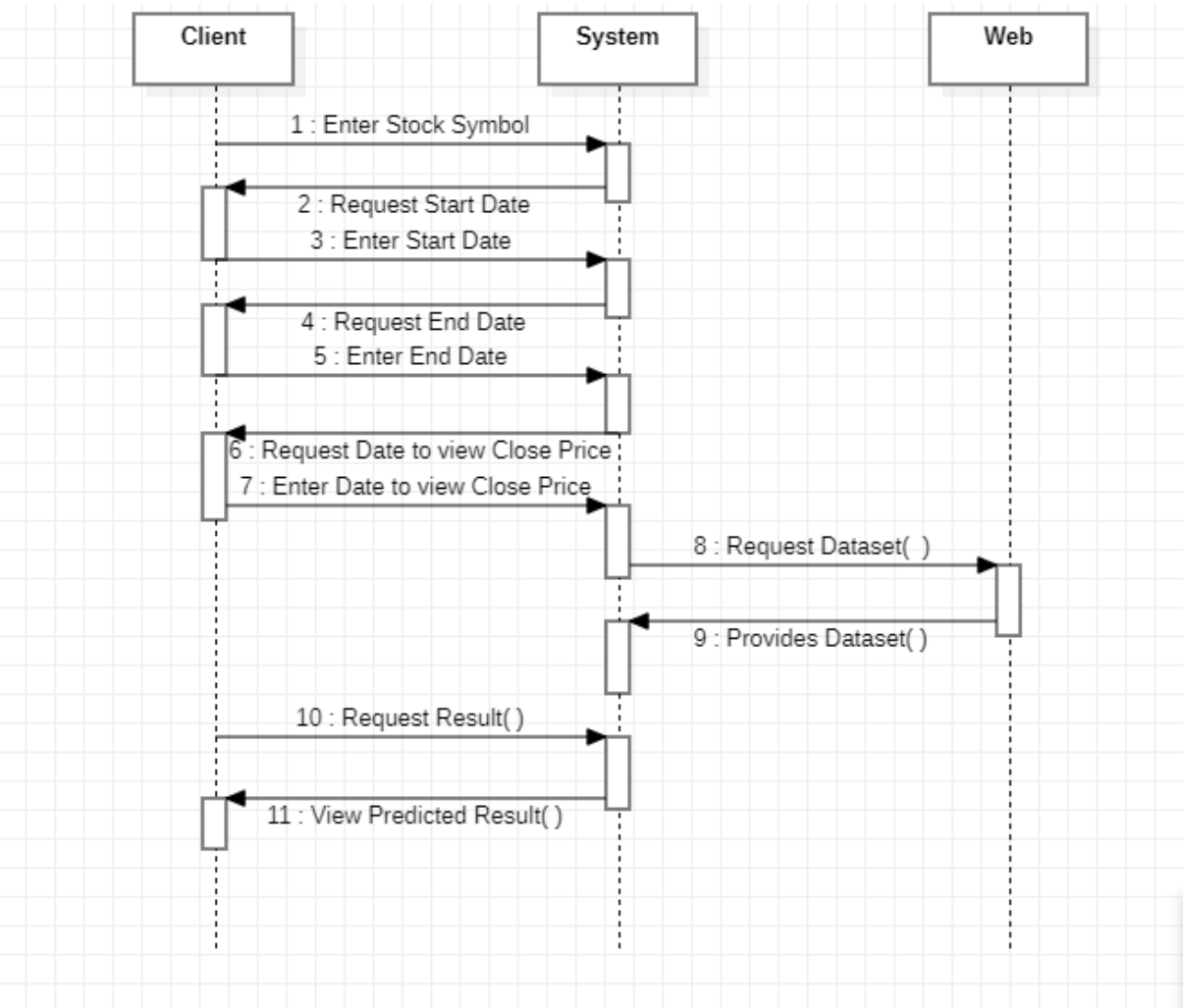


Fig 3.20: Sequence Diagram

3.3.9 State Transition Diagram:

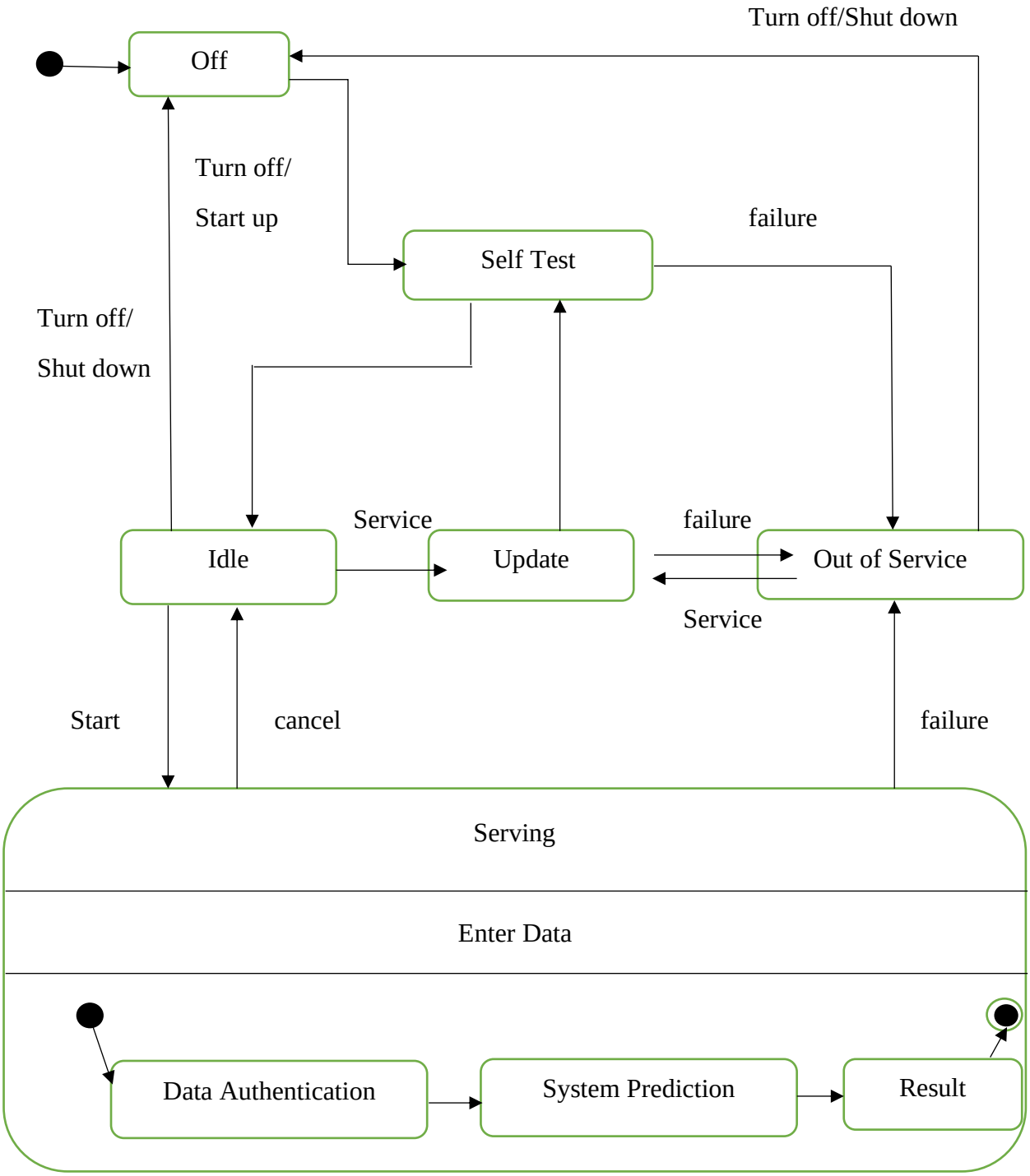


Fig 3.21: State Transition Diagram

3.3.10 Use-Case Diagram:

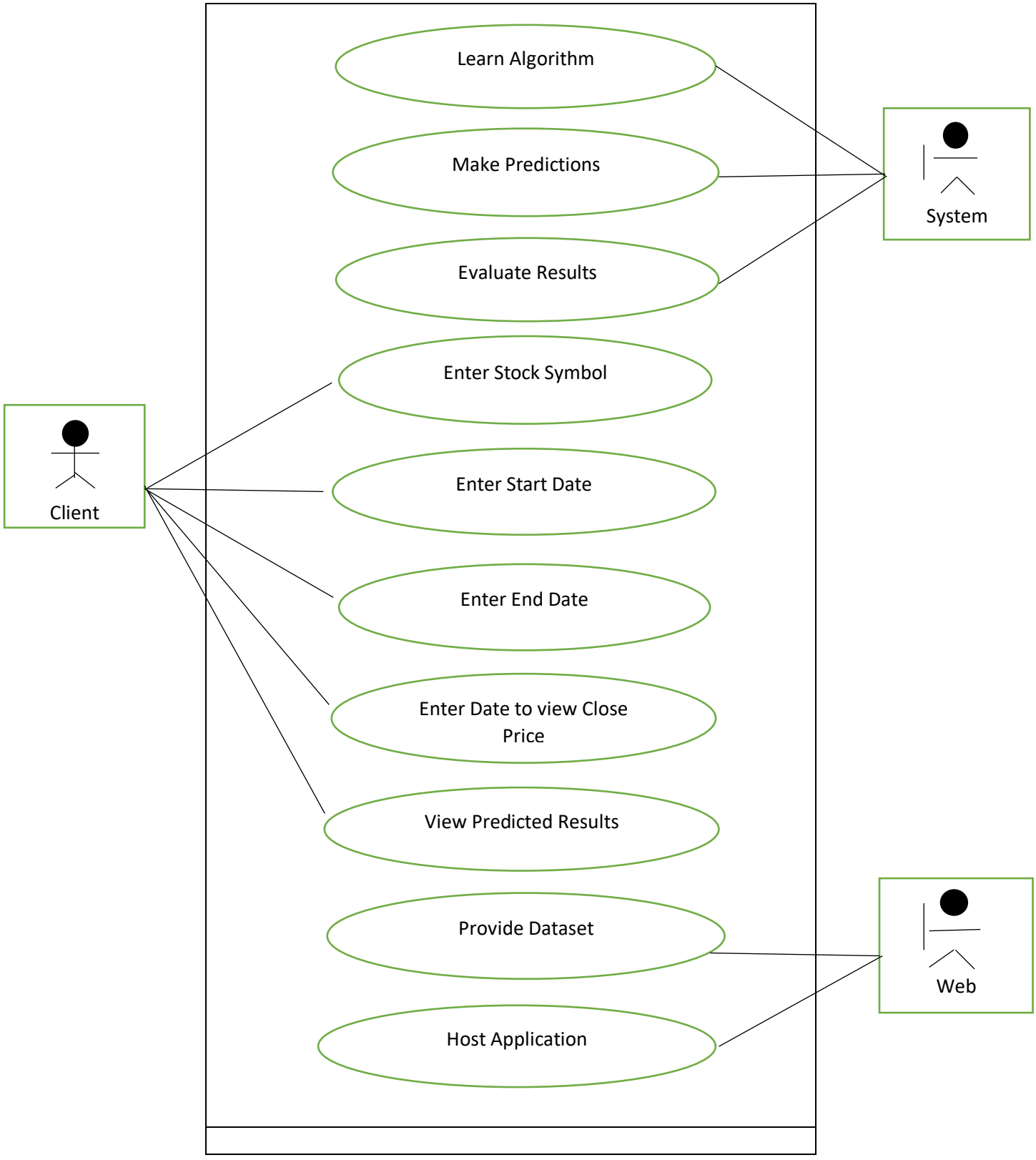


Fig 3.22: Use Case Diagram

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1 Coding

```
import streamlit as st
import pandas as pd
import numpy as np
import yfinance as yf
import plotly.express as px
import math
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import Dense, LSTM
import matplotlib.pyplot as plt
import datetime as dt

st.title('STOCK PREDICTION')

ticker = st.sidebar.text_input('Enter Stock Symbol')
start_date = st.sidebar.date_input('Enter Start Date')
end_date = st.sidebar.date_input('Enter End Date')

if ticker == '':
    st.write("Select Stock")
else:
    st.write(" ## Prediction for ", ticker)

data = yf.download(ticker, start=start_date, end=end_date)
data

fig = px.line(data, x = data.index, y = data['Adj Close'], title = ticker)
st.plotly_chart(fig)


#Create a new dataframe with only the 'Close column'
col_close = data.filter(['Close'])

#Convert the dataframe to a numpy array
dataset = col_close.values

#Get the number of rows to train the model on
training_data_len = math.ceil( len(dataset) * .8 )

#Scale the data
scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(dataset)
```

```
#Create the scaled training data set
train_data = scaled_data[0:training_data_len , :]

#Split the data into x_train and y_train data sets
x_train = []
y_train = []
for i in range(60, len(train_data)):
    x_train.append(train_data[i-60:i, 0])
    y_train.append(train_data[i,0])

#Convert the x_train and y_train to numpy arrays
x_train, y_train = np.array(x_train), np.array(y_train)

#Reshape the data
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
x_train.shape

#Build the LSTM model
model = Sequential()
model.add(LSTM(50, return_sequences = True, input_shape= (x_train.shape[1],
1)))
model.add(LSTM(50, return_sequences = False))
model.add(Dense(25))
model.add(Dense(1))

#Compile the model
model.compile(optimizer='adam', loss='mean_squared_error')

#Train the model
model.fit(x_train, y_train, batch_size=1, epochs=1)

#Create the testing dataset

#Create the new array containing scaled values from index 1802 to 2003
test_data = scaled_data[training_data_len - 60: , :]

#Create the data sets x_test and y_test
x_test = []
y_test = dataset[training_data_len:, :]
for i in range(60, len(test_data)):
    x_test.append(test_data[i-60:i, 0])

#Convert the data to a numpy array
x_test = np.array(x_test)

#Reshape the data
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1 ))
```

```
#Get the models prdicted price values
predictions = model.predict(x_test)
predictions = scaler.inverse_transform(predictions)

#Get the root mean squarred error (RMSE)
rmse = np.sqrt( np.mean( predictions - y_test )**2 )

#Plot the data
train = data[:training_data_len]
valid = data[training_data_len:]
valid['Predictions'] = predictions

#Visualize the data plt.figure(figsize=(16,8))
plt.title('Model')
plt.xlabel('Date', fontsize=18)
plt.ylabel('Close Price USD ($)', fontsize=18)
plt.plot(train['Close'])
plt.plot(valid[['Close', 'Predictions']])
plt.legend(['Train','Val','Predictions'], loc= 'lower right')
plt.show()
st.write(' ## Close Price and Predicted Close Graph')
predict_chart_data = pd.DataFrame(valid, columns=['Close', 'Predictions'])
st.line_chart(predict_chart_data)

#Show the valid and predicted prices
st.write(' ## Close Price and Predicted Close ')
st.write(valid)

#Get the quote to find the predicted CLOSE price
p_end = st.sidebar.date_input("Enter Date to find the Predicted Close:",
dt.date(2022, 1,16))
data = yf.download(ticker, start=start_date, end=end_date)

#Create a new dataframe
new_df = data.filter(['Close'])
#Get the last 60 days closing price values and convert the dataframe to an
array
last_60_days = new_df[-60:].values
#Scale the data to be values between 0 and 1
last_60_days_scaled = scaler.transform(last_60_days)
#Create an empty list
X_test = []
#Append the past 60 days
X_test.append(last_60_days_scaled)
#Convert the X_test data set to a numpy array
X_test = np.array(X_test)
#Reshape the data
X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
```

```
#Get the predicted scaled price
pred_price = model.predict(X_test)
#undo the scaling
pred_price = scaler.inverse_transform(pred_price)
labels = ['Predicted Price']
pred_price = pd.DataFrame(pred_price, columns=labels)
st.write("Predicted price for",p_end,":",pred_price)

#Creating Tabs
pricing_data, news = st.tabs(["Pricing Data", "News"])

with pricing_data:
    st.header('Stock Prices')
    data_change = data
    data_change['% Change'] = data['Adj Close'] / data['Adj Close'].shift(1) -
1
    data_change.dropna(inplace = True)
    st.write(data_change)
    annual_return = data_change['% Change'].mean()*252*100
    st.write('Annual Return', annual_return, '%')

from stocknews import StockNews
with news:
    st.header(f'Latest News for {ticker}')
    sn = StockNews(ticker, save_news = False)
    df_news = sn.read_rss()
    for i in range(10):
        st.subheader(f'News {i+1}')
        st.write(df_news['published'][i])
        st.write(df_news['title'][i])
        st.write(df_news['summary'][i])
        title_sentiment = df_news['sentiment_title'][i]
        st.write(f'Title Sentiment {title_sentiment}')
        news_sentiment = df_news['sentiment_summary'][i]
        st.write(f'News Sentiment {news_sentiment}')
```

4.2 Test Cases:

4.2.1 Test Case 1:

Test Case ID: SP001

Name: Import Stock Dataset.

Test Case Description: Machine Learning Algorithm works on dataset. It requires a training dataset and a testing dataset. The dataset required for this project is imported from yahoo dataset directory.

Initial State: Dataset not imported.

Input:

```
import pandas_datareader as web
```

Fig 4.1: Database input command.

Expected Output: Stock Dataset Imported.

Actual Output: Stock Dataset Actively working.

	Open	High	Low	Close	Adj Close	Volume
2022-06-22T00:00:00	110.5565	113.3465	110.3825	111.4875	111.4875	30774000
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000

Fig 4.2: Dataset

Remark: The Actual Output and the Expected Output are the same.

4.2.2 Test Case 2:

Test Case ID: SP002

Name: Creating User Interface.

Test Case Description: A User Interface is a must for the user to interact with the software. A User Interface is a mediator between the backend and the frontend.

Initial State: User Interface not implemented.

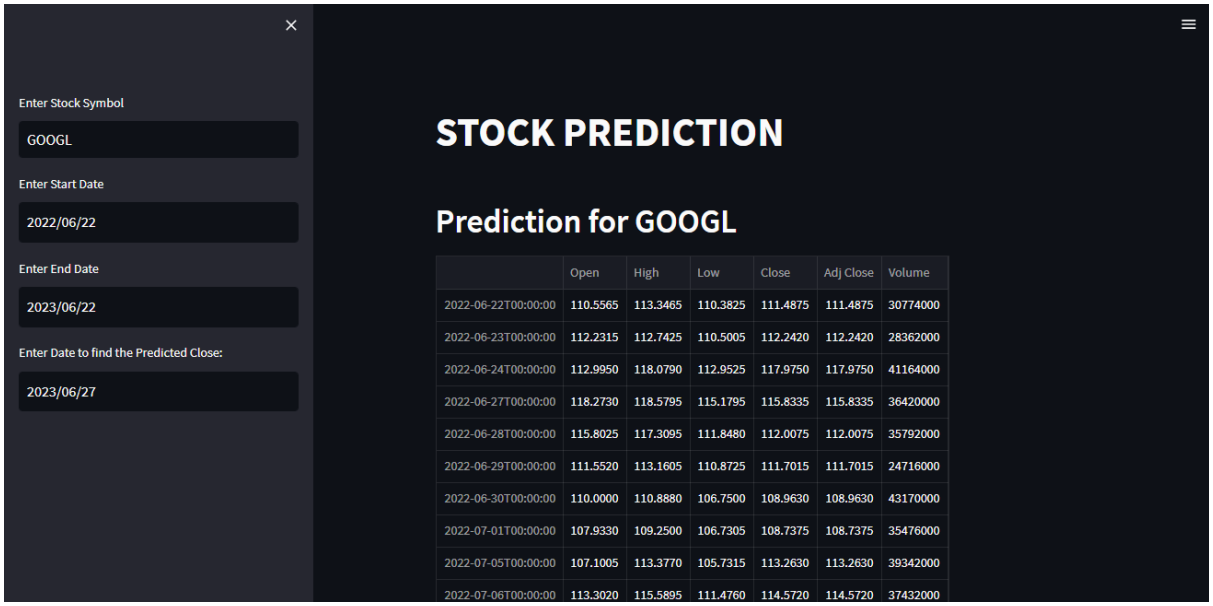
Input:

```
#Import the required libraries
import datetime as dt
import math
import pandas_datareader as web
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import Dense, LSTM
import matplotlib.pyplot as plt
import streamlit as st
```

Fig 4.3: Input Libraries.

Expected Output: A proper user-friendly interface.

Actual Output: Working User-interface.



	Open	High	Low	Close	Adj Close	Volume
2022-06-22T00:00:00	110.5565	113.3465	110.3825	111.4875	111.4875	30774000
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000

Fig 4.4: Streamlit Interface.

Remark: The Actual Output and the Expected Output are the same.

4.2.3 Test Case 3:

Test Case: SP003.

Name: Entering all the Requirements.

Test Case Description: This project requires 4 entries to be filled by the user.

Initial State: Active.

Input: 1. Enter Stock Symbol.

2. Enter Start Date.

3. Enter End Date.

4. Enter the date for Predicted Close.

Expected Output: System takes in all the four inputs.

Actual Output: All the requirements are satisfied producing desired output.

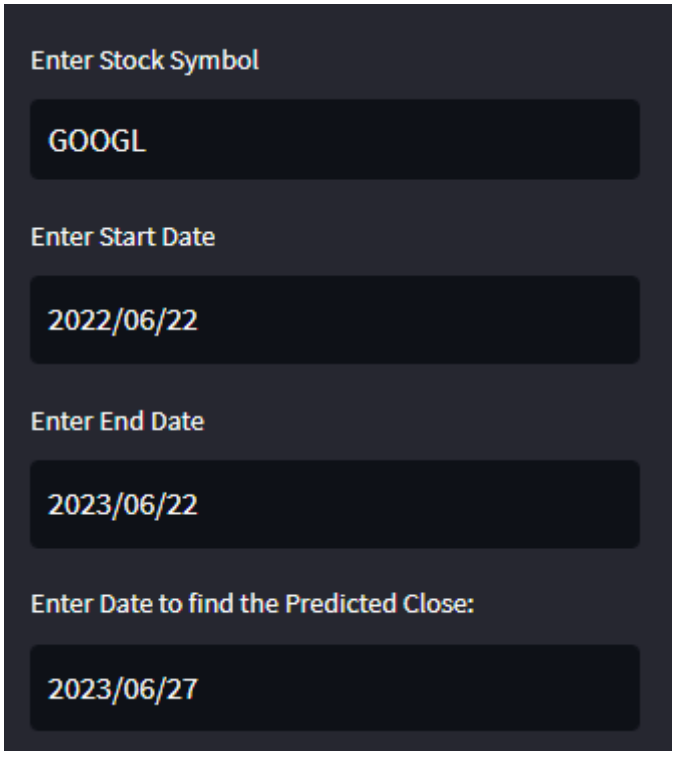


Fig. 4.5: Data Entries.

Remark: The Actual Output and the Expected Output are the same.

4.2.4 Test Case 4:

Test Case ID: SP004

Name: Closing Value Table.

Test Case Description: Depending on the token that is entered by the user the system will show a table with all the stock values for that Company.

Initial State: Select a Token.

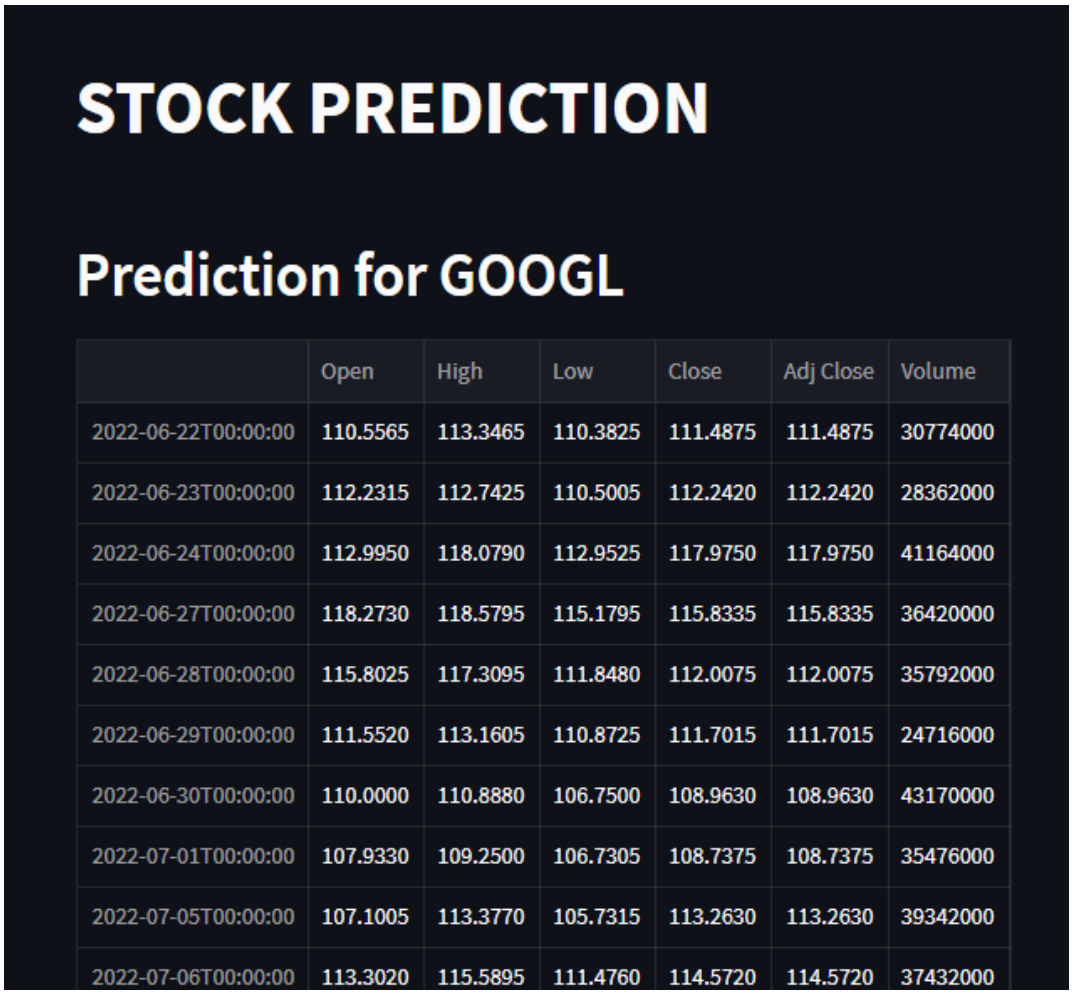
Input: 1. Enter Stock Symbol.

2. Enter Start Date.

3. Enter End Date.

Expected Output: A table with all the Stock values from the required timeframe.

Actual Output: Streamlit displays a Closing Price Table for the selected Company.



	Open	High	Low	Close	Adj Close	Volume
2022-06-22T00:00:00	110.5565	113.3465	110.3825	111.4875	111.4875	30774000
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000

Fig – 4.6: Closing Value Table.

Remark: The Actual Output and the Expected Output are the same.

4.2.5 Test Case 5:

Test Case ID: SP005

Name: Closing Graph.

Test Case Description: The page will display a graph with the closing price for the stock selected.

Initial State: Select a Token.

Input: 1. Enter Stock Symbol.

2. Enter Start Date.

3. Enter End Date.

Expected Output: A Graph with the closing price from the required timeframe.

Actual Output: Streamlit displays a Closing Price Graph for the selected Company.

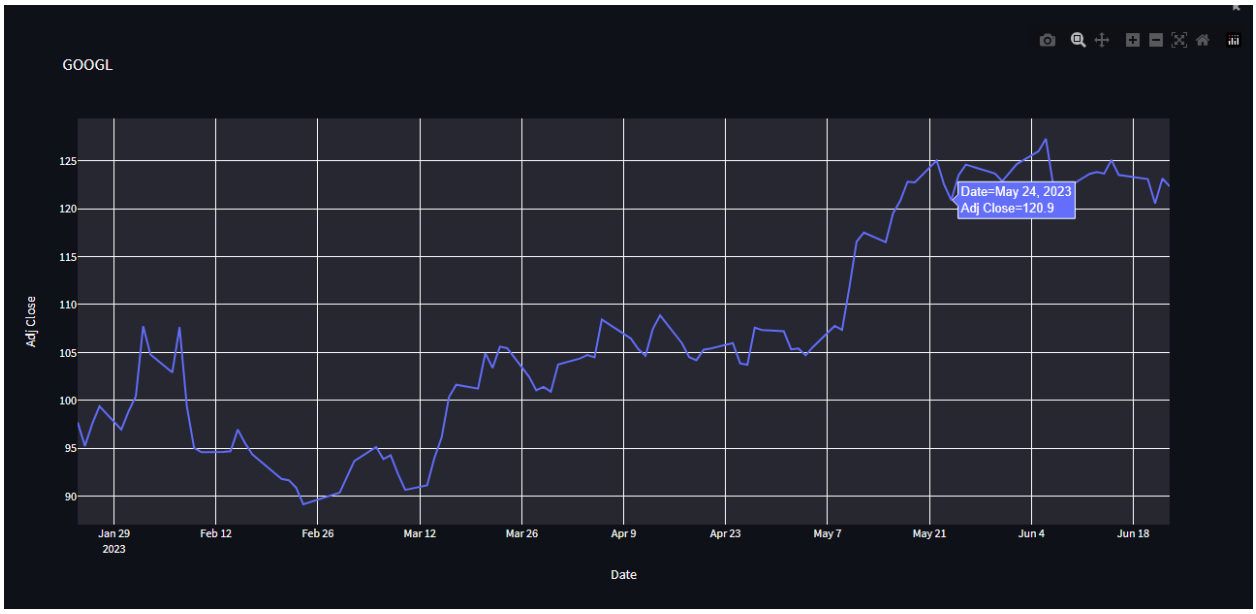


Fig – 4.7: Closing Price Graph.

Remark: The Actual Output and the Expected Output are the same.

4.2.6 Test Case 6:

Test Case ID: SP006

Name: Closing Price and Predicted Close Table.

Test Case Description: Depending on the token that is entered by the user the system will show a table with the comparison of Closing Price and Predicted Close for that Company.

Initial State: Select a Token.

Input: 1. Enter Stock Symbol.

2. Enter Start Date.

3. Enter End Date.

Expected Output: A Table with the closing price and Predicted Closing Price.

Actual Output: Streamlit displays a Closing Price and Predicted Close Table.

Close Price and Predicted Close

	Open	High	Low	Close	Adj Close	Volume	Predictions
2023-04-11T00:00:00	106.5500	106.7300	104.6800	105.3500	105.3500	26311800	103.14820098876953
2023-04-12T00:00:00	106.5800	106.7500	104.3400	104.6400	104.6400	24370300	103.40604400634766
2023-04-13T00:00:00	105.8400	107.4900	105.8400	107.4300	107.4300	24843600	103.51271057128906
2023-04-14T00:00:00	106.8900	108.9400	106.8400	108.8700	108.8700	26578000	103.7257080078125
2023-04-17T00:00:00	104.6600	106.1600	104.5200	105.9700	105.9700	37571200	104.07794189453125
2023-04-18T00:00:00	106.4900	106.5400	104.0700	104.5000	104.5000	26596400	104.2590560913086
2023-04-19T00:00:00	103.5800	104.9800	103.0700	104.1800	104.1800	20905700	104.22032165527344
2023-04-20T00:00:00	103.9100	106.2500	103.8700	105.2900	105.2900	27820800	104.03350067138672
2023-04-21T00:00:00	105.4700	106.0000	104.7800	105.4100	105.4100	25800100	103.85714721679688
2023-04-24T00:00:00	105.4900	106.6300	104.7000	105.9700	105.9700	23542800	103.70703887939453

Fig – 4.8: Closing Price and Predicted Close Table.

Remark: The Expected Output and the Actual Output are the same.

4.2.7 Test Case 7:

Test Case ID: SP007

Name: Close Price and Predicted Close Graph.

Test Case Description: The page will display a graph with the Closing Price and Predicted Closing Price for the stock selected.

Initial State: Select a Token.

Input: 1. Enter Stock Symbol.

2. Enter Start Date.

3. Enter End Date.

Expected Output: A Graph with the Closing Price and Predicted Closing Price from the required timeframe.

Actual Output: Streamlit displays a Closing Price and Predicted Close Graph.

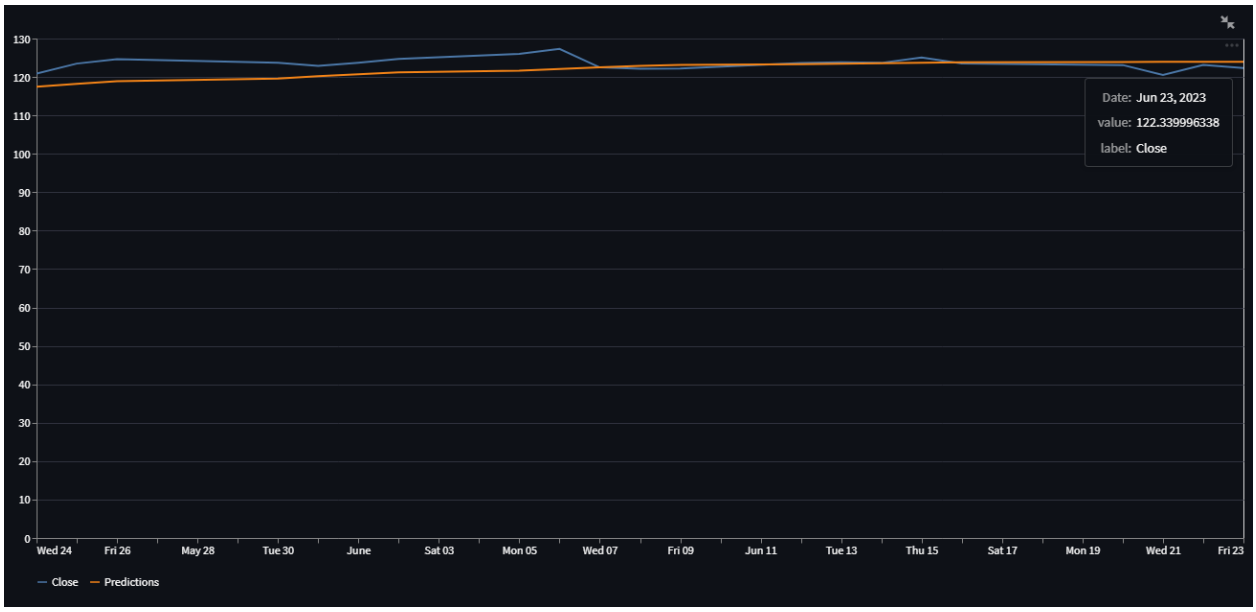


Fig – 4.9: Closing Price and Predicted Close Graph.

Remark: The Expected Output and Actual Output are the same.

4.2.8 Test Case 8:

Test Case ID: SP008

Name: Predicted Value.

Test Case Description: The page will display a Predicted Value for a particular date for the stock selected.

Initial State: Select a Token.

Input:

Enter Date to find Predicted Close.

Expected Output: A predicted value will be displayed for the given date and selected company.

Actual Output: Streamlit displays a Predicted Value for the particular Date.

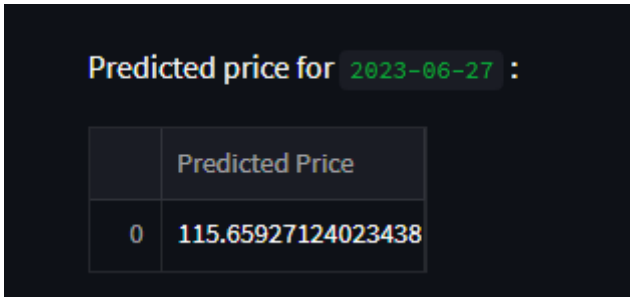


Fig – 4.10: Predicted Value

Remark: The Expected Output and the Actual Output are the same.

4.2.9 Test Case 9:

Test Case ID: SP009

Name: % Change Table and Annual Return.

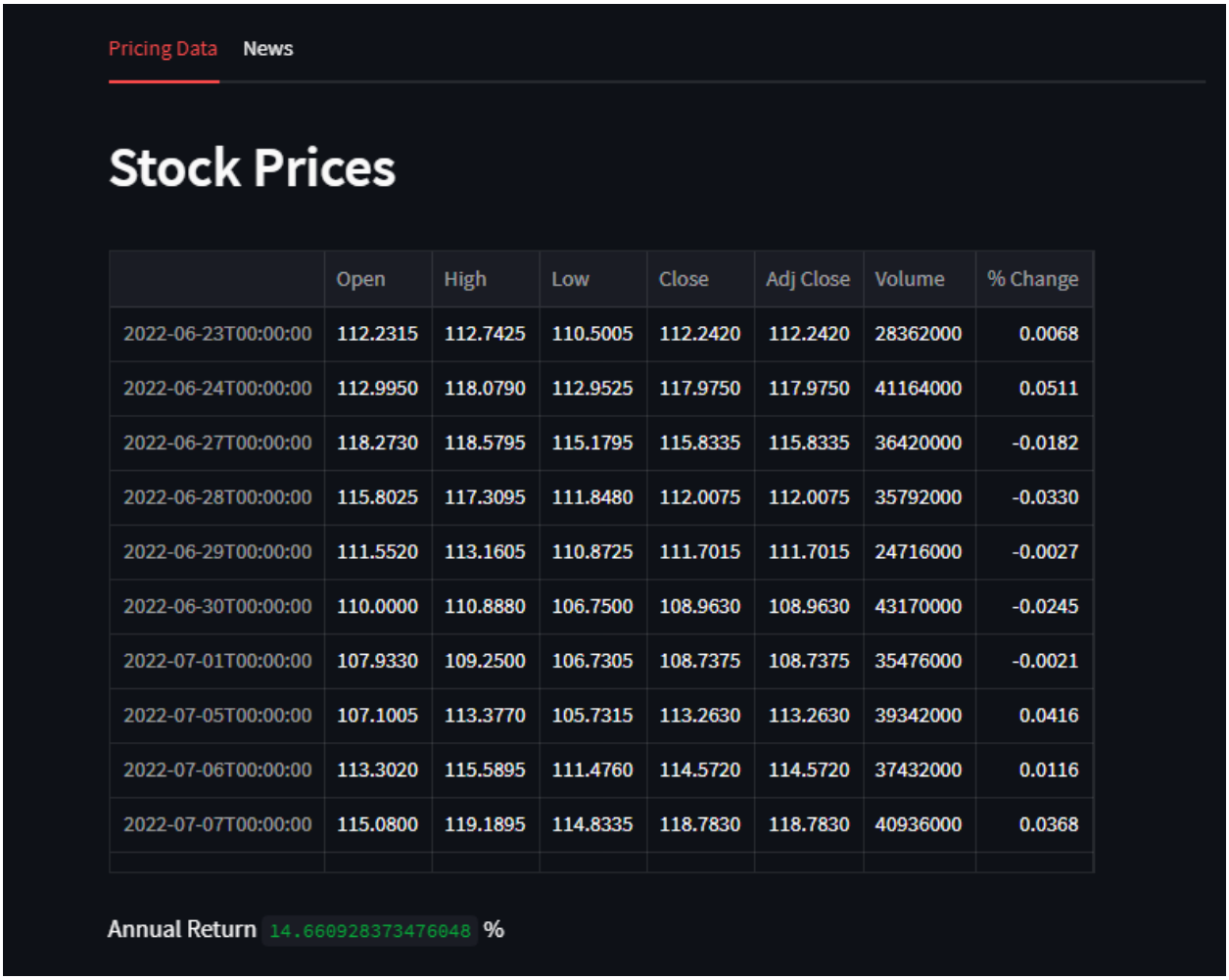
Test Case Description: The page will display table with a column for % Change in the values of the company and its Annual Returns.

Initial State: Select a Token.

Input: The Company selected.

Expected Output: A Table with % Change column and Annual Return is displayed.

Actual Output: Streamlit displays a table with change of stock value compared to the previous day and its Annual Returns.



	Open	High	Low	Close	Adj Close	Volume	% Change
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000	0.0068
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000	0.0511
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000	-0.0182
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000	-0.0330
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000	-0.0027
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000	-0.0245
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000	-0.0021
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000	0.0416
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000	0.0116
2022-07-07T00:00:00	115.0800	119.1895	114.8335	118.7830	118.7830	40936000	0.0368

Fig – 4.11: Pricing Data.

Remark: The Expected Output and the Actual Output are the same.

4.2.10 Test Case 10:

Test Case ID: SP010

Name: News Data.

Test Case Description: The page will display Top 10 recent News related to the company selected.

Initial State: Select a Token.

Input: The Company selected.

Expected Output: A page that displays Top 10 related news.

Actual Output: Streamlit displays the Top 10 News related to the company in recent times.

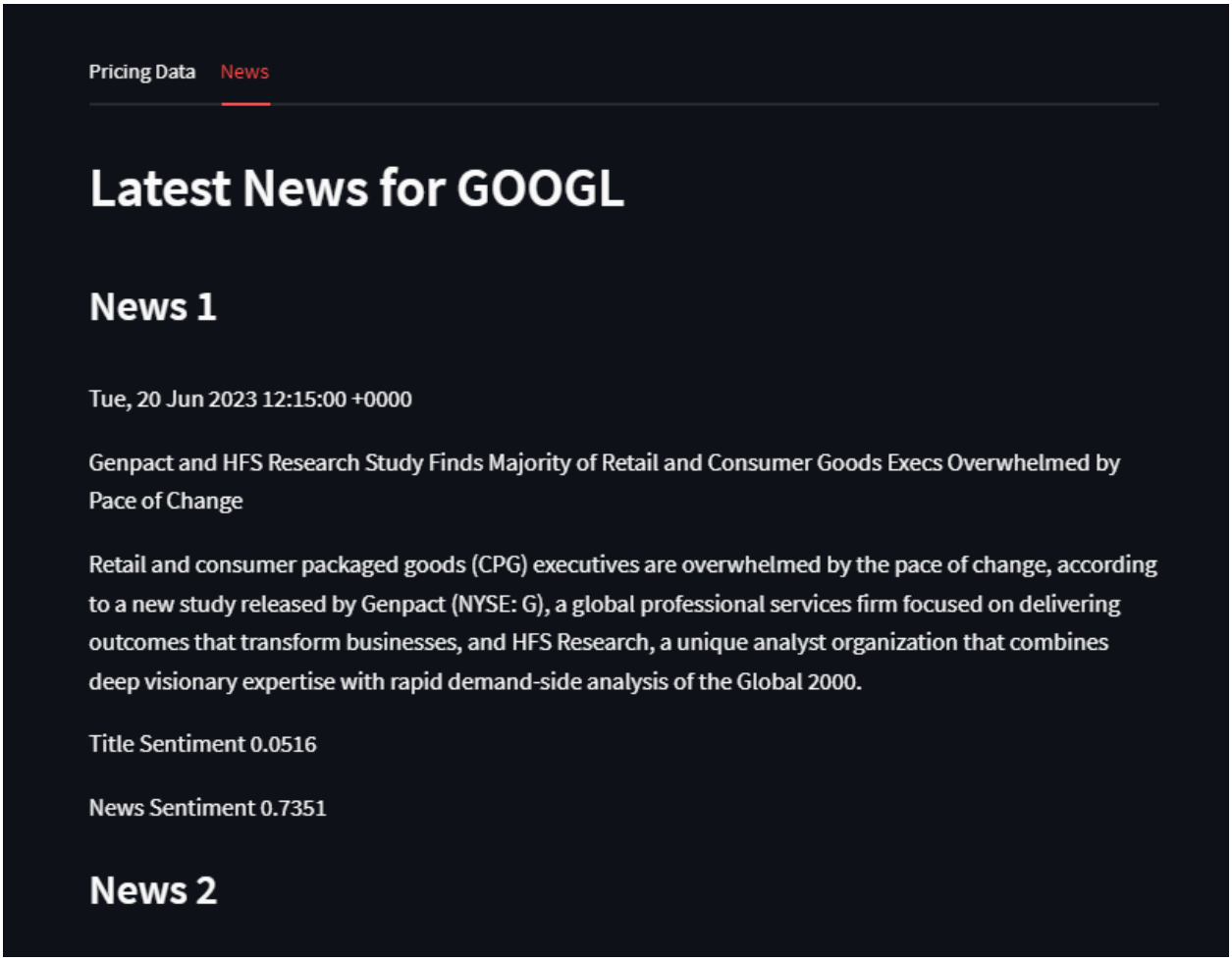


Fig – 4.12: Top 10 News.

Remark: The Expected Output and the Actual Output are the same.

4.2.11 Test Case 11:

Test Case ID: SP011.

Name: Running the Software.

Test Case Description: Once done with the frontend and backend the software needs to run in the terminal in order to launch the Streamlit page.

Initial State: Inactive

Input:

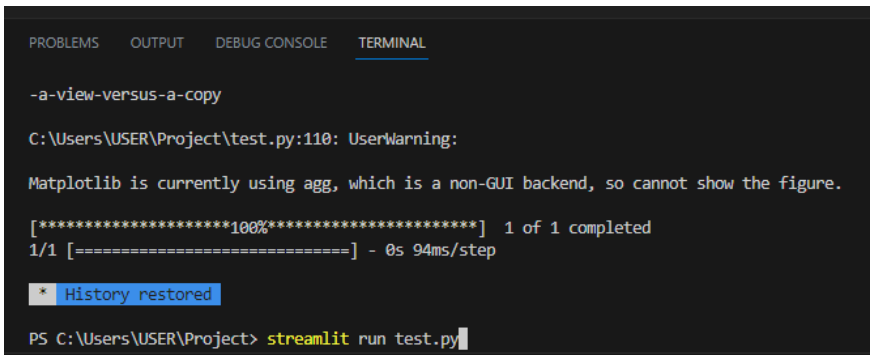


Fig. 4.13: Run Command.

Expected Output: The terminal is able to load the page.

Actual Output:

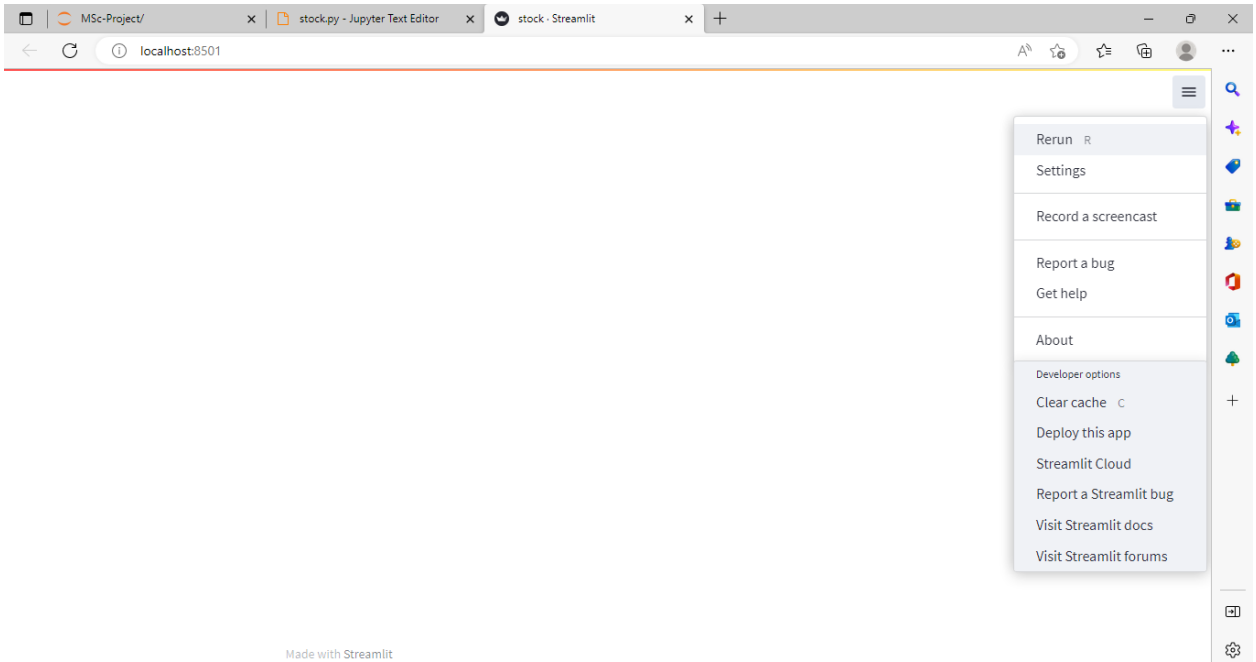
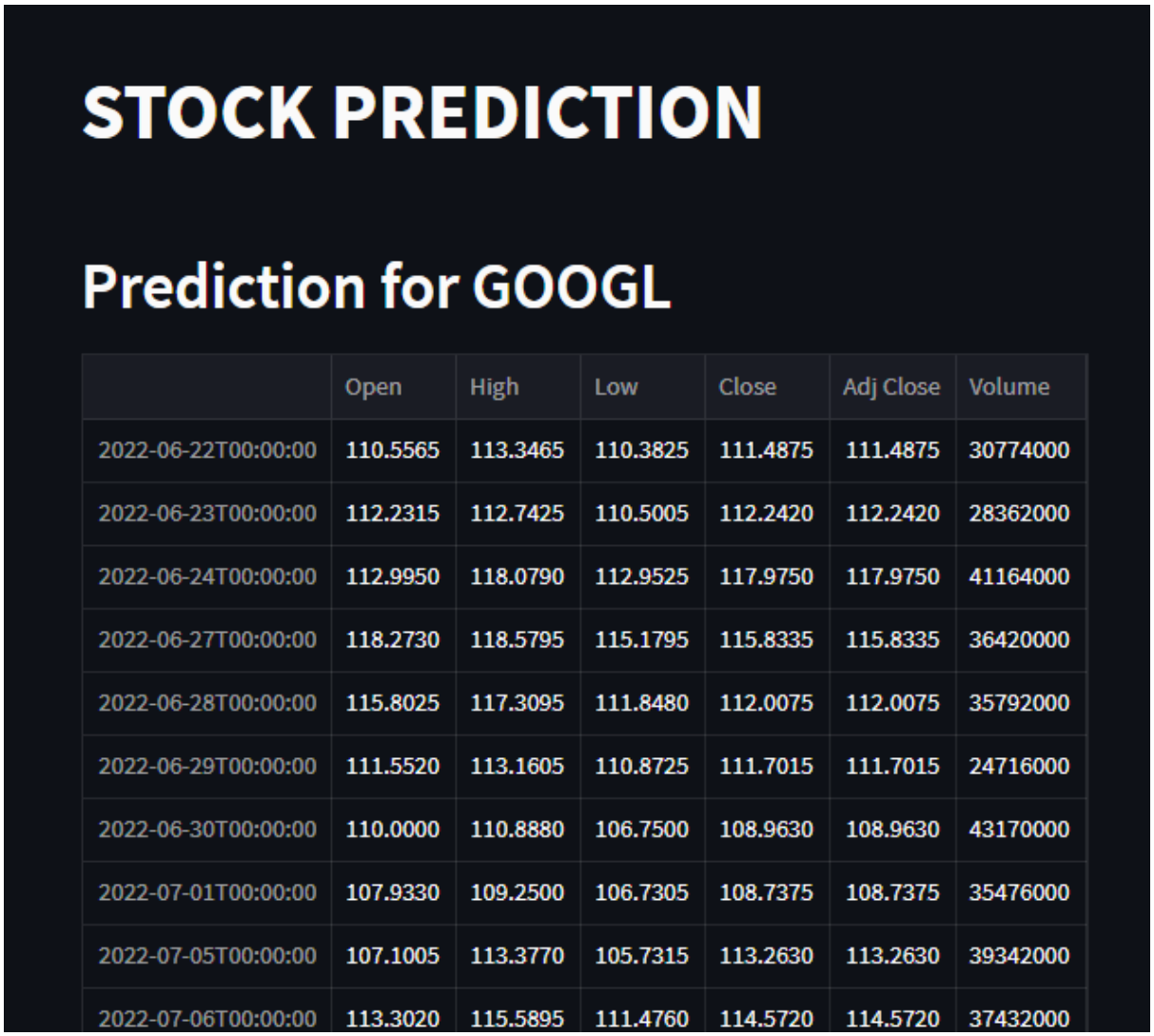


Fig. 4.14: Streamlit Page.

Remark: The Actual Output and the Expected Output are the same.

CHAPTER 5: RESULT AND DISCUSSION



STOCK PREDICTION

Prediction for GOOGL

	Open	High	Low	Close	Adj Close	Volume
2022-06-22T00:00:00	110.5565	113.3465	110.3825	111.4875	111.4875	30774000
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000

Fig – 5.1: Closing Value Table

- The Closing Value Table gives the data for the selected stock company.
- It displays data like the Open and Close values of a particular stock for that day, the Adj Close i.e., the Adjacent Close value of the stock and the final Volume means the total transactions carried out for the particular day.

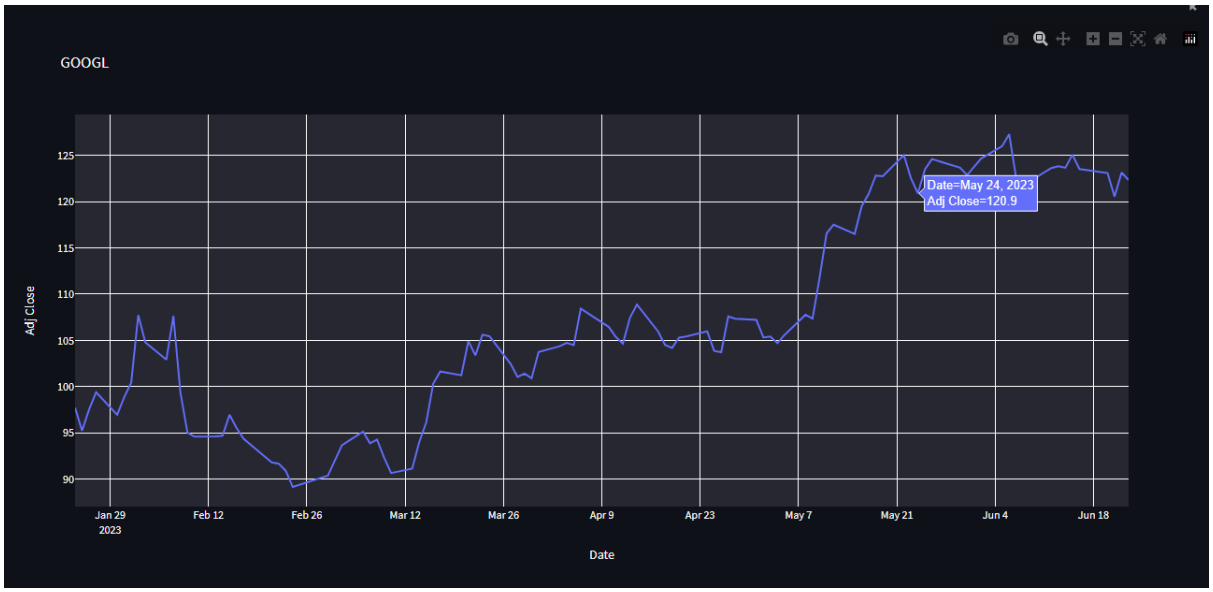
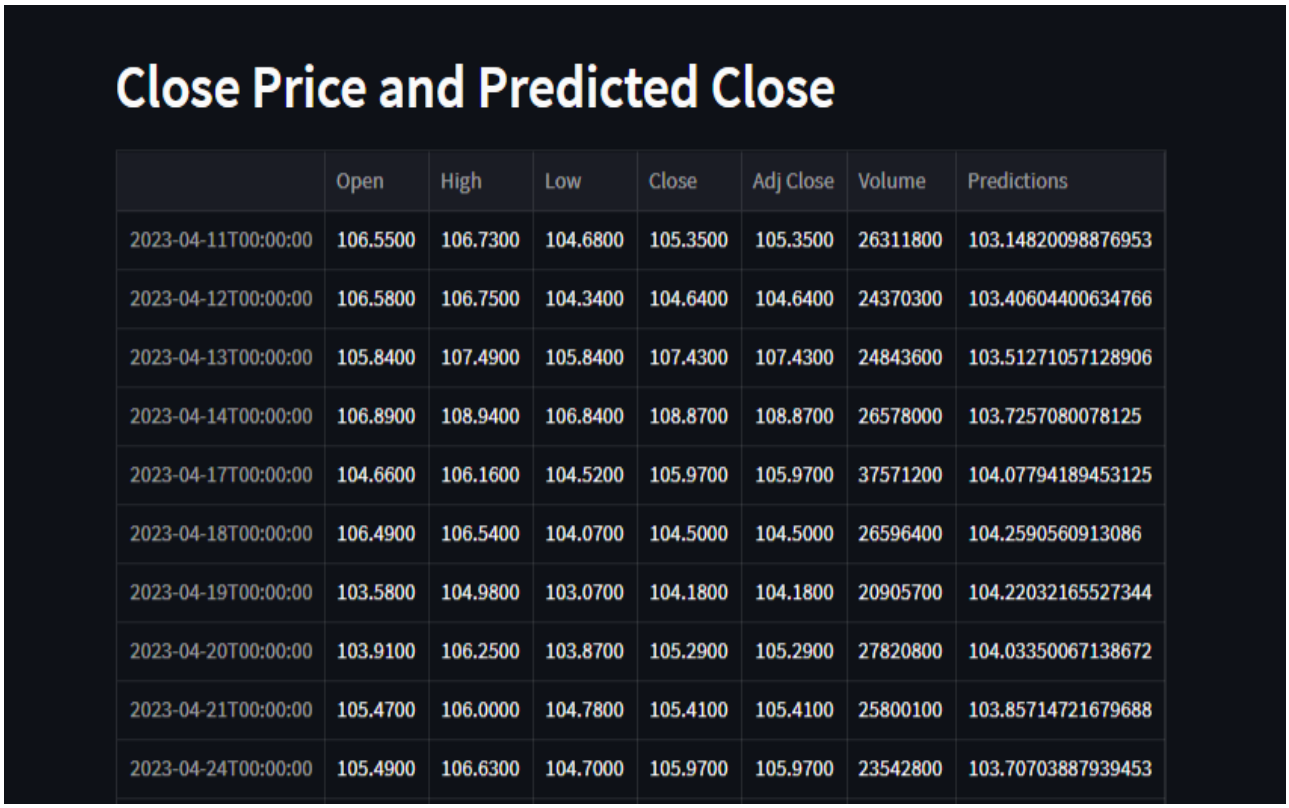


Fig – 5.2: Closing Price Graph

- The Closing Price Graph is a graphical representation of the fluctuation of the prices for a particular stock in the given time frame.

- The X – coordinate shows the dates and the Y – coordinate shows the Adjacent Close values of the stock.



Close Price and Predicted Close

	Open	High	Low	Close	Adj Close	Volume	Predictions
2023-04-11T00:00:00	106.5500	106.7300	104.6800	105.3500	105.3500	26311800	103.14820098876953
2023-04-12T00:00:00	106.5800	106.7500	104.3400	104.6400	104.6400	24370300	103.40604400634766
2023-04-13T00:00:00	105.8400	107.4900	105.8400	107.4300	107.4300	24843600	103.51271057128906
2023-04-14T00:00:00	106.8900	108.9400	106.8400	108.8700	108.8700	26578000	103.7257080078125
2023-04-17T00:00:00	104.6600	106.1600	104.5200	105.9700	105.9700	37571200	104.07794189453125
2023-04-18T00:00:00	106.4900	106.5400	104.0700	104.5000	104.5000	26596400	104.2590560913086
2023-04-19T00:00:00	103.5800	104.9800	103.0700	104.1800	104.1800	20905700	104.22032165527344
2023-04-20T00:00:00	103.9100	106.2500	103.8700	105.2900	105.2900	27820800	104.03350067138672
2023-04-21T00:00:00	105.4700	106.0000	104.7800	105.4100	105.4100	25800100	103.85714721679688
2023-04-24T00:00:00	105.4900	106.6300	104.7000	105.9700	105.9700	23542800	103.70703887939453

Fig – 5.3: Closing Price and Predicted Close Table

- The Closing Price and Predicted Close Table shows all the values like Open, Close, Adj Close, Volume along with columns like Low, High and Prediction.

- The Low column represents the lowest value for the stock for that day, similarly the High value represent the highest value for that day.

- The Prediction column represents the predicted value of the stock for that particular day.

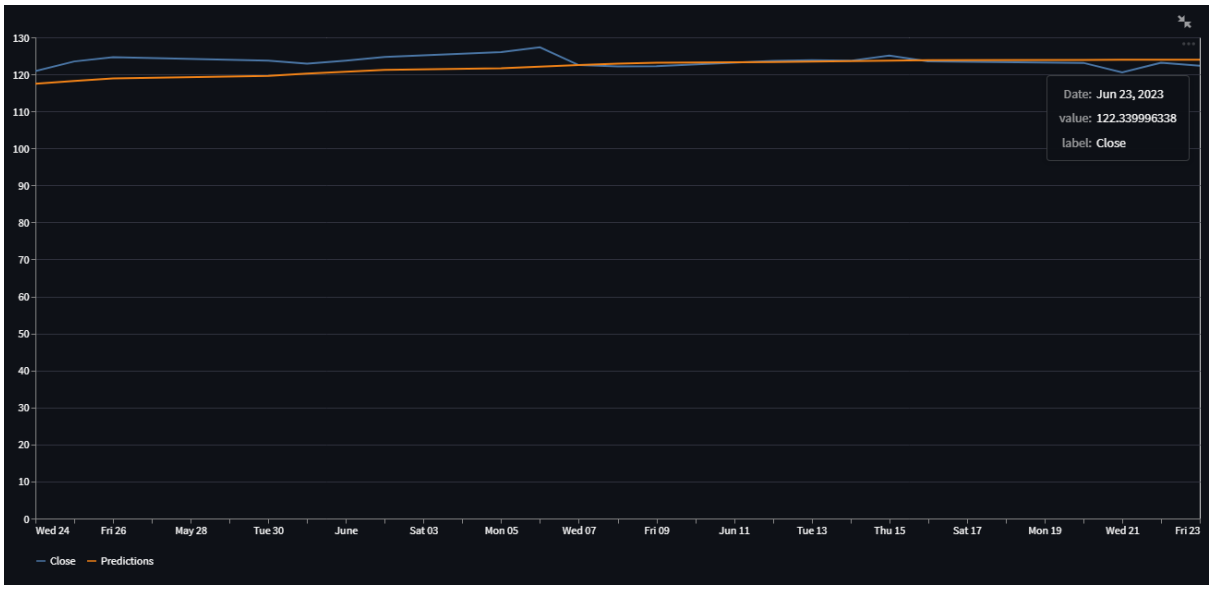
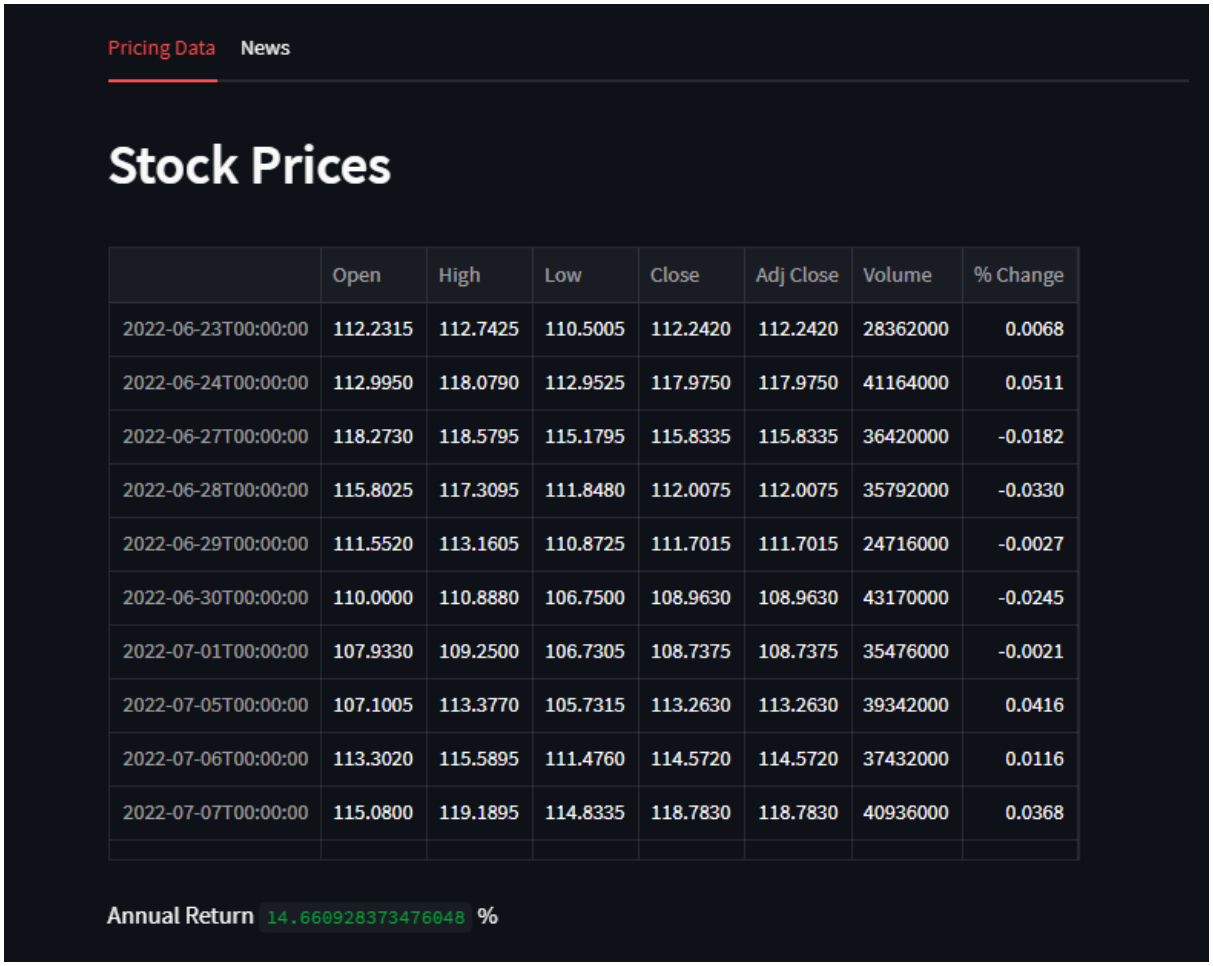


Fig 5.4 – Predicted Close Graph

- The predicted Close Graph is a graphical representation of the predicted value of the particular stock in contrast to its actual closing value.

- The blue line represents the Closing price and the orange line represents the Predicted price.

- The X – coordinate shows the dates while the Y – coordinate shows the stock price for both closing price and predicted close.



	Open	High	Low	Close	Adj Close	Volume	% Change
2022-06-23T00:00:00	112.2315	112.7425	110.5005	112.2420	112.2420	28362000	0.0068
2022-06-24T00:00:00	112.9950	118.0790	112.9525	117.9750	117.9750	41164000	0.0511
2022-06-27T00:00:00	118.2730	118.5795	115.1795	115.8335	115.8335	36420000	-0.0182
2022-06-28T00:00:00	115.8025	117.3095	111.8480	112.0075	112.0075	35792000	-0.0330
2022-06-29T00:00:00	111.5520	113.1605	110.8725	111.7015	111.7015	24716000	-0.0027
2022-06-30T00:00:00	110.0000	110.8880	106.7500	108.9630	108.9630	43170000	-0.0245
2022-07-01T00:00:00	107.9330	109.2500	106.7305	108.7375	108.7375	35476000	-0.0021
2022-07-05T00:00:00	107.1005	113.3770	105.7315	113.2630	113.2630	39342000	0.0416
2022-07-06T00:00:00	113.3020	115.5895	111.4760	114.5720	114.5720	37432000	0.0116
2022-07-07T00:00:00	115.0800	119.1895	114.8335	118.7830	118.7830	40936000	0.0368

Fig – 5.5: Pricing Data

- The Pricing Table show all the related values of the stock for that day alongside with an additional column of % Change.

- The % Change column shows the change in value of the stock in comparison to previous day. The positive value shows that the price of the stock increased compared to yesterday and so the negative value shows that the price decreased.

- Along with the table is a element that shows the Annual Return of the company i.e., the annual performance of the company's stock.

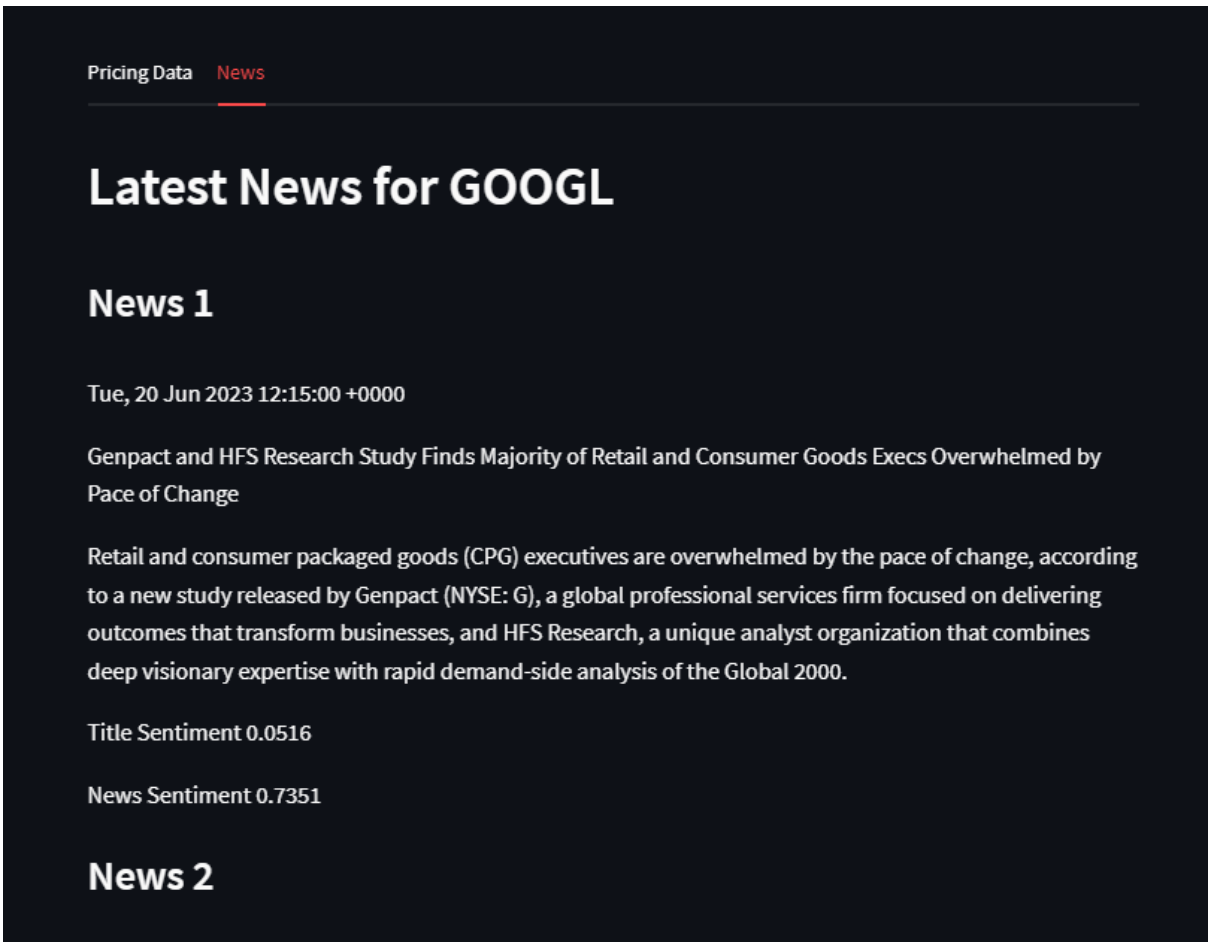


Fig – 5.6: Top 10 News

- The page also shows the Top 10 latest news related to the stock company.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion:

• In this project, I tried to develop a prediction model for the stock market based on the technical analysis using LSTM stock market data.

• LSTM model are very powerful in sequence prediction problems because they are able to store past information.

• This could guide the future investors in the stock market to make profitable investment decisions whether to buy or sell or hold a share.

• Stock markets are hard to monitor and require plenty of context when try to interpret the movement and predict prices.

• They are able to keep track of context specific temporal dependencies between stock prices for a longer period of time while performing predictions.

• Prediction models although very useful are also subjected to various changes in fields like medical, hospitality, transportation, travel, environment, etc.

6.2 Future Work:

The goal of stock prediction is to help organizations deliver maximum business value. Periodization and managing resource capacity help us to make 'the most effective use of human resources'. The project is based on a machine learning algorithm that is used as a feedforward link. The same artificial neural network can be used in various different domains to predict the future outcome like

1. **Health Care Facility:**
   A patient's response to a future procedure can be predicted based on the history of the patient.

2. **Traffic Controller:**
   Past as well as present data can be used to predict the traffic on a road on a particular day and a particular time. The same can be applied for Air Traffic Controlling.

3. **Event Sourcing:**
   The probability and the number of crowed that attend a particular event can roughly be calculated.

4. **Weather Forecasting:**
   The machine learning algorithm can best predict the weather if provided with accurate past and present data.

CHAPTER 7: REFERENCES

For Streamlit Functionality and Syntax:

- https://docs.streamlit.io/library/api-reference?highlight=altair_chart#display-data

For Machine Learning and Deep Learning Concepts:

- https://sist.sathyabama.ac.in/sist_coursematerial/uploads/SEC1609.pdf
- https://ai.stanford.edu/~nilsson/MLBOOK.pdf
- https://cs229.stanford.edu/materials/CS229-DeepLearning.pdf

For Deep Learning Algorithms:

- https://theaisummer.com/Deep-Learning-Algorithms/

For Long Short Term Memory (LSTM) Algorithm:

- https://pages.cs.wisc.edu/~shavlik/cs638/lectureNotes/Long%20Short-Term%20Memory%20Networks.pdf
- https://direct.mit.edu/neco/article-abstract/9/8/1735/6109/Long-Short-Term-Memory?redirectedFrom=fulltext