A
Project Report
On
"STOCKIFY-A WEB APP"

Prepared by

PRAKSHAL BHANDARI (20DCS007)

AARYA BHATT (20DCS008)

DEVANSHI BHATT (20DCS009)

Under the guidance of

PROF. URVI BHATT

Assistant Professor

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

4th Semester Software Group Project-II (CE255)

Submitted at

CSE

DEPSTAR

At: Changa, Dist: Anand – 388421

April 2022

CERTIFICATE

This is to certify that the report entitled "**STOCKIFY (A WEB APP)**" is a bonafide work carried out by **Mr.Prakshal Bhandari (20DCS007),Mr.Aarya Bhatt (20DCS008),Ms.Devanshi Bhatt (20DCS009)** under the guidance and supervision of **Assistant Prof.Urvi Bhatt** for the subject **CE255-Software Group Project-II** (CSE) of 4th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Prof. Urvi Bhatt
Assistant Professor
Computer Science and Engineering
DEPSTAR, Changa, Gujarat.

Dr. Parth Goel
Head of Department – Computer Science and Engineering, DEPSTAR
CHARUSAT, Changa, Gujarat.

Dr. Amit P. Ganatra
Principal, DEPSTAR
Dean, FTE
CHARUSAT, Changa, Gujarat.

DECLARATION BY THE CANDIDATE

I hereby declare that the project report entitled “Stockify (A WEB APP)” submitted by me to Devang Patel Institute of Advance Technology and Research, Changa in partial fulfilment of the requirement for the award of the degree of B.Tech in Computer Science and Engineering, from Devang Patel Institute of Advance Technology and Research (DEPSTAR), is a record of bonafide CE255 Software Group Project-II (project work) carried out by us under the guidance of Prof. Urvi bhatt. I further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Prakshal Bhandari(20DCS007)
Aarya Bhatt (20DCS008)
Devanshi Bhatt (20DCS009)

Prof. Urvi Bhatt
Assistant Professor
Devang Patel Institute of Advance Technology and Research (DEPSTAR),
CHARUSAT-Changa.

ACKNOWLEDGEMENT

We, the developers of "STOCKIFY (A Web App)", with immense pleasure and commitment would like to present the project assignment. The development of this project has given us a wide opportunity to think, implement and interact with various aspects of management skills as well as the different technologies.

Every work that one completes successfully stands on the constant encouragement, goodwill and support of the people around. We hereby avail this opportunity to express our gratitude to the number of people who extended their valuable time, full support and cooperation in developing the project.

We express a deep sense of gratitude towards our Head of the CSE Department, Dr. Parth Goel and project guide Prof. Urvi Bhatt for the support during the whole session of study and development. It is because of them that we were prompted to do hard work, adopting new technologies. We are thankful for all my teachers and friends who have been always helping to encourage me throughout the year. We have no valuable words to express our thanks, but our heart is still full of the favors we received from every person.

Yours Sincerely,

Prakshal Bhandari(20DCS007)

Aarya Bhatt (20DCS008)

Devanshi Bhatt (20DCS009)

ABSTRACT

In a financially volatile market, as the stock market, it is important to have a very precise prediction of a future trend. Because of the financial crisis and scoring profits, it is mandatory to have a secure prediction of the values of the stocks. Stock market price data is generated in huge volume and it changes every second. Stock market is a complex and challenging system where people will either gain money or lose their entire life savings. Predicting how the stock market will perform is a hard task to do. There are other factors involved in the prediction, such as physical and psychological factors, rational and irrational behavior, and so on. All these factors combine to make share prices dynamic and volatile. This makes it very difficult to predict stock prices with high accuracy. Stock Price Prediction using machine learning helps you discover the future value of company stock and other financial assets traded on an exchange. The stock market appears in the news every day. You hear about it every time it reaches a new high or a new low. In this report, we will see if there is a possibility of devising a project which will predict stock price with a less percentage of error.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1.1 PROJECT OVERVIEW

The financial market is a dynamic and composite system where people can buy and sell currencies, stocks, equities and derivatives over virtual platforms supported by brokers. The stock market allows investors to own shares of public companies through trading either by exchange or over the counter markets. This market has given investors the chance of gaining money and having a prosperous life through investing small initial amounts of money, low risk compared to the risk of opening new business or the need of high salary career. Stock markets are affected by many factors causing the uncertainty and high volatility in the market. Although humans can take orders and submit them to the market, automated trading systems (ATS) that are operated by the implementation of computer programs can perform better and with higher momentum in submitting orders than any human. However, to evaluate and control the performance of ATSs, the implementation of risk strategies and safety measures applied based on human judgements are required. Many factors are incorporated and considered when developing an ATS, for instance, trading strategy to be adopted, complex mathematical functions that reflect the state of a specific stock, machine learning algorithms that enable the prediction of the future stock value, and specific news related to the stock being analysed.

Time-series prediction is a common technique widely used in many real-world applications such as weather forecasting and financial market prediction. It uses the continuous data in a period of time to predict the result in the next time unit. Many timeseries prediction algorithms have shown their effectiveness in practice. The most common algorithms now are based on Recurrent Neural Networks (RNN), as well as its special type - Long-short Term Memory (LSTM) and Gated Recurrent Unit (GRU). Stock market is a typical area that presents time-series data and many researchers study on it and proposed various models.

Researchers have been studying different methods to effectively predict the stock market price. Useful prediction systems allow traders to get better insights about data such as: future trends. Also, investors have a major benefit since the analysis give future conditions of the market. One such method is to use machine learning algorithms for forecasting. This project’s objective is to

improve the quality of output of stock market predicted by using stock value. A number of researchers have come up with various ways to solve this problem, mainly there are traditional methods so far, such as artificial neural network is a way to get hidden patterns and classify the data which is used in predicting stock market. This project proposes a different method for prognosting stock market prices. It does not fit the data to a specific model; rather we are identifying the latent dynamics existing in the data using machine learning architectures. In this work we use Machine learning architectures Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) and Hybrid approach of LSTM + CNN for the price forecasting of NSE listed companies and differentiating their performance. On a long term basis, sling window approach has been applied and the performance was assessed by using root mean square error.

1.2 MOTIVATION FOR WORK

Businesses primarily run over customer's satisfaction, customer reviews about their products. Shifts in sentiment on social media have been shown to correlate with shifts in stock markets. Identifying customer grievances thereby resolving them leads to customer satisfaction as well as trustworthiness of an organization. Hence there is a necessity of an un biased automated system to classify customer reviews regarding any problem. In today's environment where we're justifiably suffering from data overload (although this does not mean better or deeper insights), companies might have mountains of customer feedback collected; but for mere humans, it's still impossible to analyse it manually without any sort of error or bias. Oftentimes, companies with the best intentions find themselves in an insights vacuum. You know you need insights to inform your decision making and you know that you're lacking them, but don't know how best to get them. Sentiment analysis provides some answers into what the most important issues are, from the perspective of customers, at least. Because sentiment analysis can be automated, decisions can be made based on a significant amount of data rather than plain intuition.

1.3 PROBLEM STATEMENT

Time Series forecasting & modelling plays an important role in data analysis. Time series analysis is a specialized branch of statistics used extensively in fields such as Econometrics & Operation Research. Time Series is being widely used in analytics & data science. Stock prices are volatile in nature and price depends on various factors. The main aim of this project is to predict stock prices using Long short-term memory (LSTM).

1.4 PROPOSED SYSTEM

- To generalize the application of existing system, our work uses the system to estimate other stocks in similar emerging markets and mature markets.
- The system can be extended to analyze multivariate time series data and import raw database directly so that profit can be maximized even when corporate stock market has lower value.
- It is considered to increase usability of system.

1.5 SCOPE OF PROJECT

Analysis of stocks using data mining will be useful for new investors to invest in stock market based on the various factors considered by the software.
Stock market includes daily activities like sensex calculation, exchange of shares. The exchange provides an efficient and transparent market for trading in equity, debt instruments and derivatives.

Our software will be analyzing sensex based on company's stock value. The stock values of company depend on many factors, some of them are:

1. Demand and Supply:Demand and Supply of shares of a company is a major reason price change in stocks. When Demand Increase and Supply is less, price rises. and vice versa.
2. Corporate results: This will be regarding to the profits or progress of the company over a span of time say 3 months.

3. Popularity: Main Strength in hands of share buyer. Popularity of a company can effect on buyers. Like if any good news of a company, may result in rise of stock price. And a bad news may break dreams.

The stock value depends on other factors as well, but we are taking into consideration only these main factors.

1.6 TOOLS AND TECHNOLOGIES

PYTHON

Python was the language of choice for this project. This was an easy decision for the multiple reasons.

1. Python as a language has an enormous community behind it. Any problems that might be encountered can be easily solved with a trip to Stack Overflow. Python is among the most popular languages on the site which makes it very likely there will be a direct answer to any query.

2. Python has an abundance of powerful tools ready for scientific computing. Packages such as Numpy, Pandas, and SciPy are freely available and well documented. Packages such as these can dramatically reduce, and simplify the code needed to write a given program. This makes iteration quick.

3. Python as a language is forgiving and allows for programs that look like pseudo code. This is useful when pseudocode given in academic papers needs to be implemented and tested. Using Python, this step is usually reasonably trivial.

However, Python is not without its flaws. The language is dynamically typed and packages are notorious for Duck Typing. This can be frustrating when a package method returns something that, for example, looks like an array rather than being an actual array. Coupled with the fact that standard Python documentation does not explicitly state the return type of a method, this can lead to a lot of trials and error testing that would not otherwise

happen in a strongly typed language. This is an issue that makes learning to use a new Python package or library more difficult than it otherwise could be.

The major libraries used in our project are: -

1. Pystan: PyStan is a Python interface to Stan, a package for Bayesian inference. Stan® is a state-of-the-art platform for statistical modeling and high-performance statistical computation. There should be no difference in Stan performance, but PyStan is slower for really large data. RStan has more bells and whistles.

2. Fbprophet: Prophet is a forecasting procedure implemented in R and Python. It is fast and provides completely automated forecasts that can be tuned by hand by data scientists and analysts. The Prophet algorithm is an additive model, which means that it detects the following trend and seasonality from the data first, then combine them together to get the forecasted values. Overall Trend. Yearly, Weekly, Daily Seasonality.

3. Ephem: PyEphem is a library that implements astronomical algorithms for the Python programming language. It is free under the LGPL. The library is written in Python and C, with code for calculating positions of bodies taken from XEphem by permission. It provides an ephem Python package for performing high-precision astronomy computations. The underlying numeric routines are coded in C and are the same ones that drive the popular XEphem astronomy application, whose author, Elwood Charles Downey, generously gave permission for their use in PyEphem. The name ephem is short for the word ephemeris, which is the traditional term for a table giving the position of a planet, asteroid, or comet for a series of dates.

4. Plotly: The plotly Python library is an interactive, open-source plotting library that supports over 40 unique chart types covering a wide range of statistical, financial, geographic, scientific, and 3-dimensional use-cases. Plotly has several advantages over matplotlib. One of the main advantages is that only a few lines of codes are necessary to create aesthetically pleasing, interactive plots. The interactivity also offers a number of advantages over static matplotlib plots: Saves time when initially exploring your dataset.

5. Streamlit: Streamlit is an open-source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib etc.

6. Time: The Python time module provides many ways of representing time in code, such as objects, numbers, and strings. It also provides functionality other than representing time, like waiting during code execution and measuring the efficiency of your code.

7. Socket: Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while the other socket reaches out to the other to form a connection.

Streamlit

Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science. In just a few minutes you can build and deploy powerful data apps. Overall, Streamlit is an easy-to-learn framework that allows data science teams to create free predicitve analytics web applications in as little as a few hours. The Streamlit gallery shows many open-source projects that have used it for analytics and machine learning. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib etc. With Streamlit, no callbacks are needed since widgets are treated as variables. Data caching simplifies and speeds up computation pipelines. Streamlit watches for changes on updates of the linked Git repository and the application will be deployed automatically in the shared link.

Streamlit vs. Flask

Streamlit is a data dashboarding tool, while Flask is a web framework. Serving pages to users is an important but small component of data dashboards. Flask

doesn’t have any data visualization, manipulation, or analytical capabilities (though since it’s a general Python library, it can work well with other libraries that perform these tasks). Streamlit is an all-in-one tool that encompasses web serving as well as data analysis.

Use Streamlit if you want a structured data dashboard with many of the components you’ll need already included. Use Streamlit if you want to build a data dashboard with common components and don’t want to reinvent the wheel.

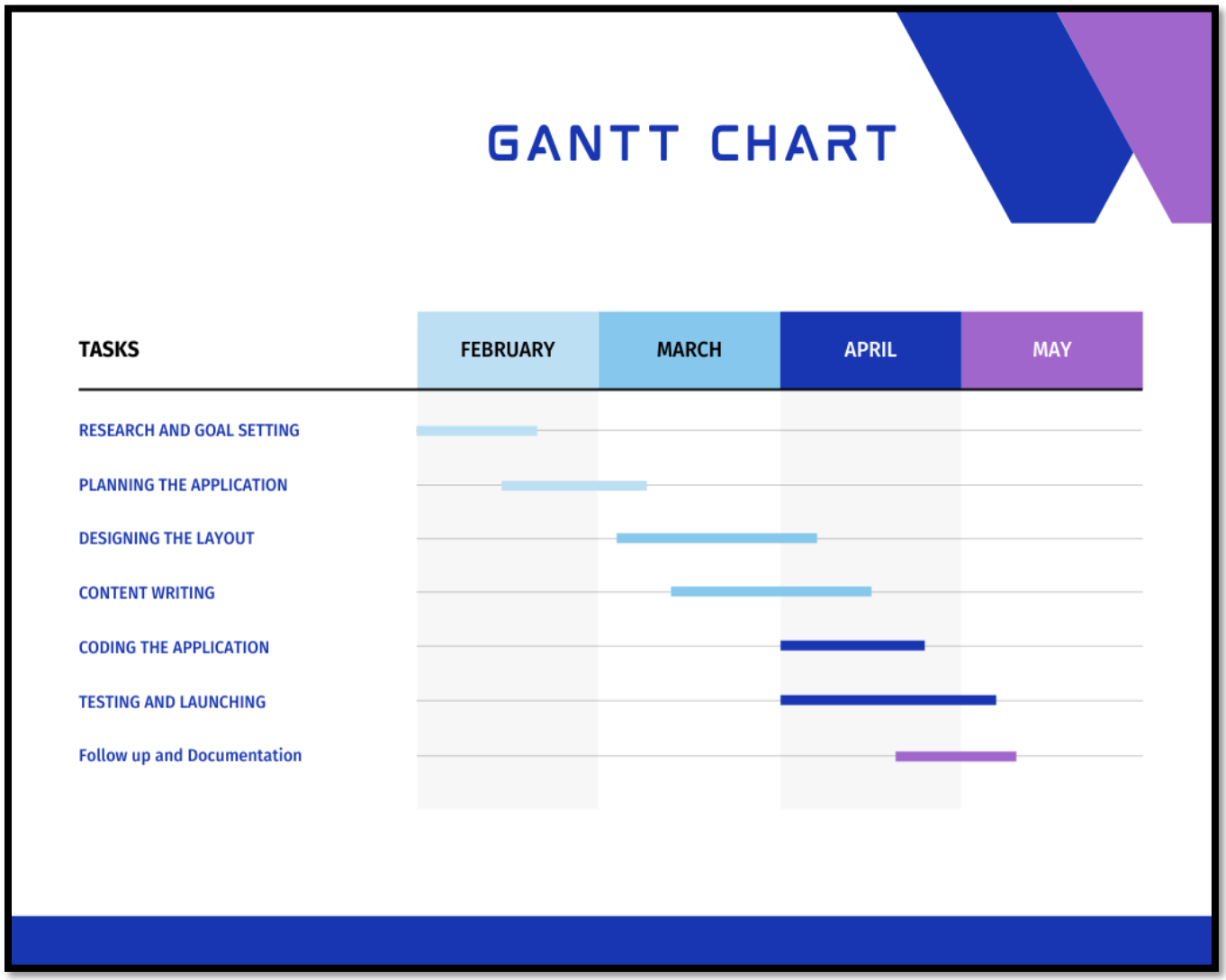
Use Flask if you want to build a highly customized solution from the ground up and you have the engineering capacity.

CHAPTER 2 PROJECT MANAGEMENT

2.1 PROJECT WORK SCHEDULING

2.1.1 GANTT CHART

Fig 2.1 Gantt chart



2.1.2 PROJECT PLANNING PROCESS

Roles and Responsibilties

- Prakshal :- Time and Socket, yfinance
- Aarya :- Streamline, Base64
- Devanshi :- Plotly, fbprophet

Fig 2.2 Planning Process

CHAPTER 3 SYSTEM REQUIREMENTS STUDY

3.1 HARDWARE AND SOFTWARE REQUIREMENT

3.1.1 HARDWARE SPECIFICATIONS

- Processor: Minimum 2GHz or more.
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
- Hard Drive: Minimum 32 GB; Recommended 64 GB or more.
- Memory (RAM): Minimum 4 GB or above
- MAC-OS and LINUX-OS.

3.1.2 SOFTWARE SPECIFICATIONS

- Operating system: Linux- Ubuntu 16.04 to 17.10, or Windows 7 to 10.
- You have to install Python 3.6 and related packages.
- Pystan
- Ephem
- Fbprophet
- Plotly
- Streamlit
- Time
- Socket
- Base64

CHAPTER 4 SYSTEM DESIGN

4.1 SCREEN LAYOUT

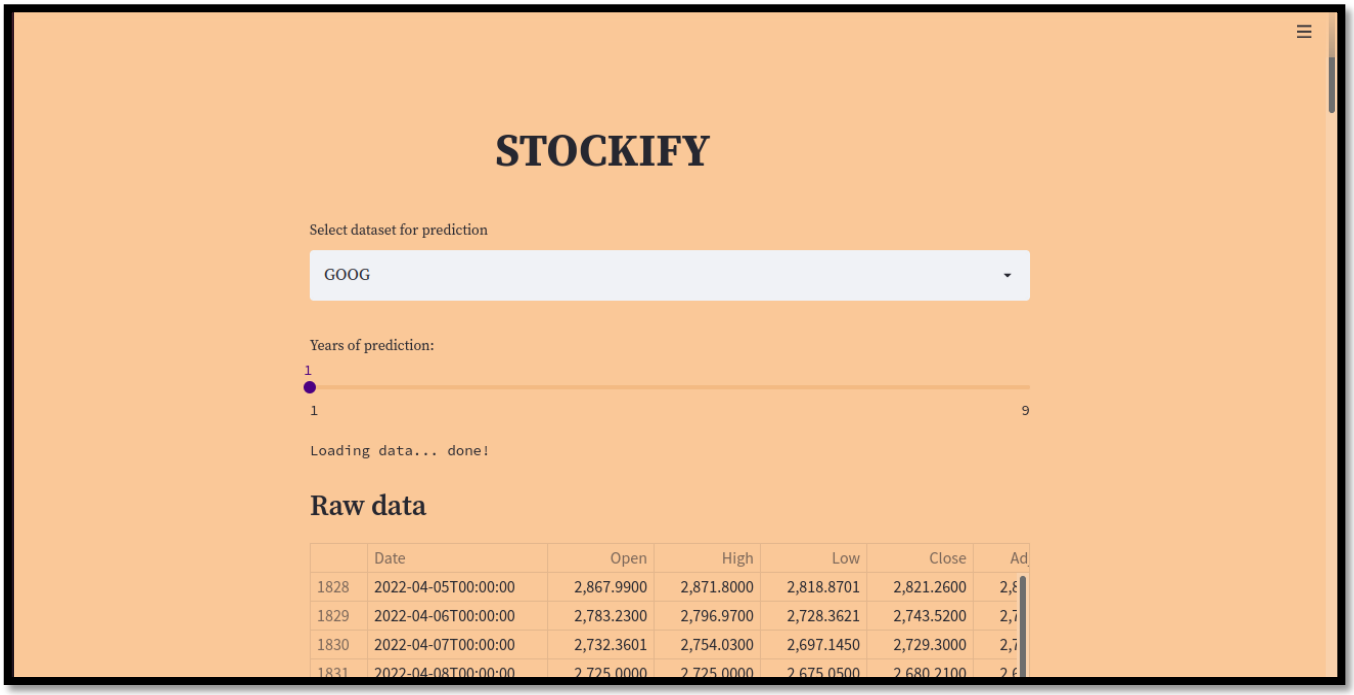


Fig 4.1 Project Implementation-1

Fig 4.2 Project Implementation-2

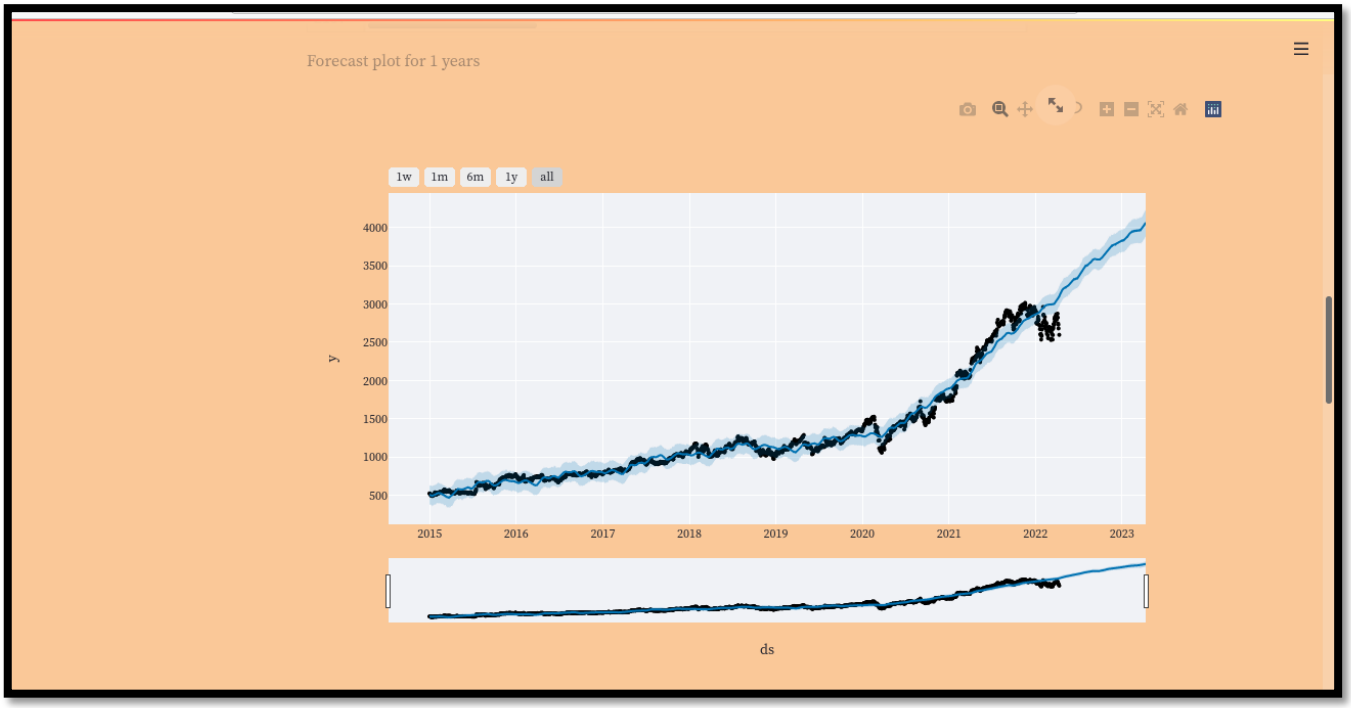


Fig 4.3 Project Implementation-3

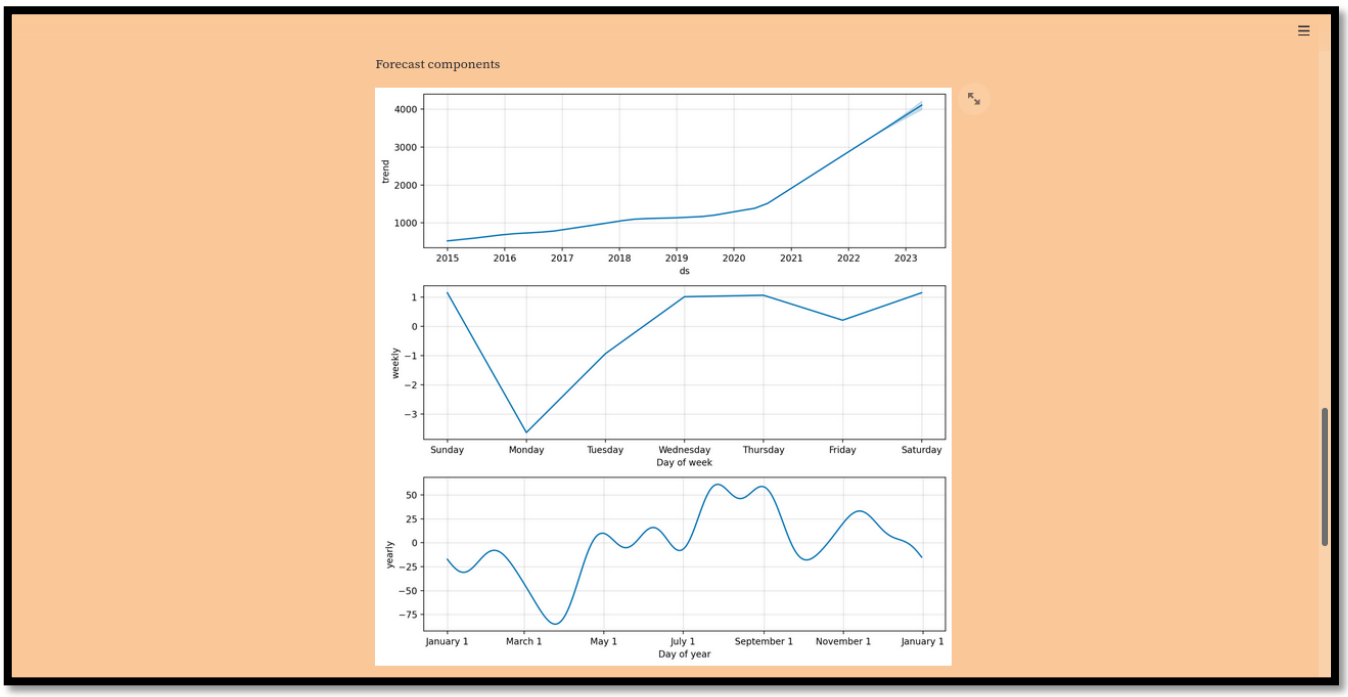


Fig 4.4 Project Implementation-4

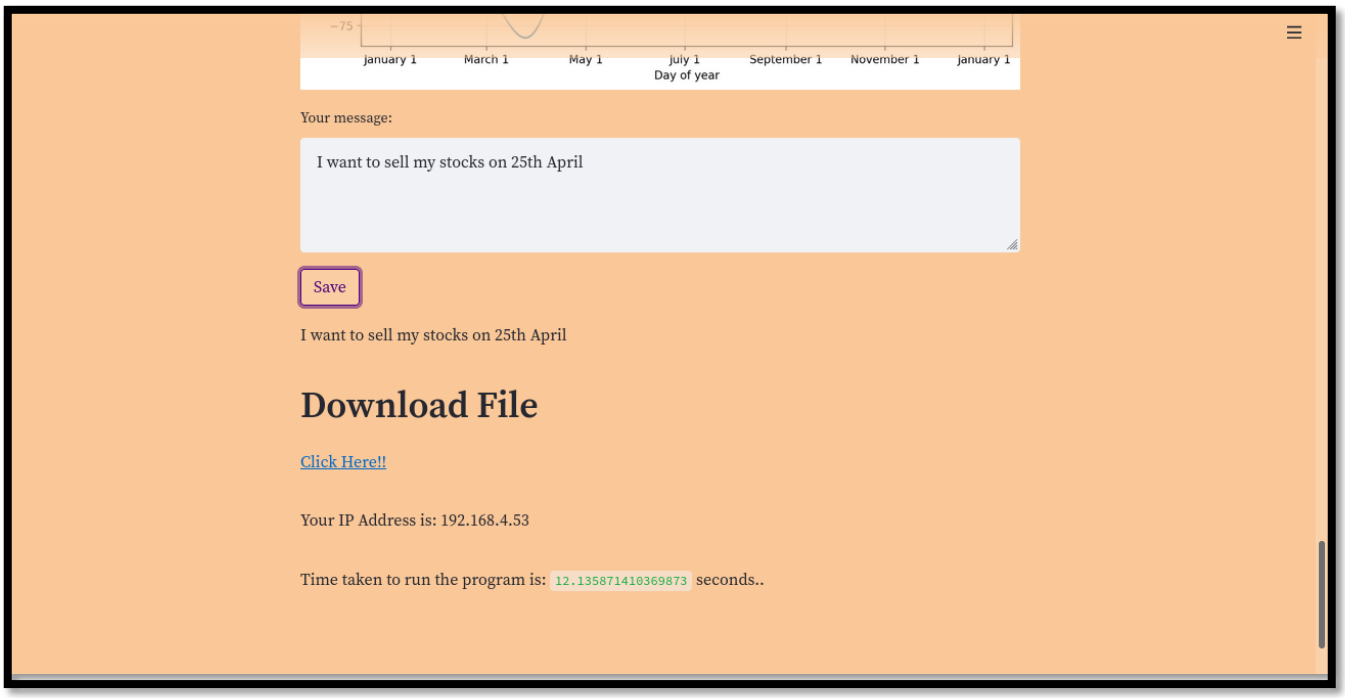


Fig 4.5 Project Implementation-5

4.2 METHOD PSEUDO CODES

```
import streamlit as st
import streamlit.components as stc
from datetime import date
import yfinance as yf
from fbprophet import Prophet
from fbprophet.plot import plot_plotly
from plotly import graph_objs as go
import time
import socket
import base64
timestr= time.strftime("%Y%m%d-%H%M%S")


start=time.time()

START = "2015-01-01"
TODAY = date.today().strftime("%Y-%m-%d")

_, col2, _ = st.columns([1, 2 , 1])

with col2:
    st.title("STOCKIFY")


stocks = ('GOOG', 'AAPL', 'MSFT','TCS','BHARTIARTL.NS','ONGC.NS','NMDC.NS')
st.text("")
selected_stock = st.selectbox('Select dataset for prediction', stocks)
st.text("")
n_years = st.slider('Years of prediction:', 1, 9)
period = n_years * 365


@st.cache
def load_data(ticker):
    data = yf.download(ticker, START, TODAY)
    data.reset_index(inplace=True)
    return data


data_load_state = st.text('Loading data...')
data = load_data(selected_stock)
data load state.text('Loading data... done!')
```

Fig 4.2.1 Method Psuedo Code-1

```
st.subheader('Raw data')
st.write(data.tail())

# Plot raw data
def plot_raw_data():
    fig = go.Figure()
    fig.add_trace(go.Scatter(x=data['Date'], y=data['Open'], name="stock_open"))
    fig.add_trace(go.Scatter(x=data['Date'], y=data['Close'], name="stock_close"))
    fig.layout.update(title_text='Time Series data with Rangeslider', xaxis_rangeslider_visible=True)
    st.plotly_chart(fig)

plot_raw_data()

# Predict forecast with Prophet.
df_train = data[['Date','Close']]
df_train = df_train.rename(columns={"Date": "ds", "Close": "y"})

m = Prophet()
m.fit(df_train)
future = m.make_future_dataframe(periods=period)
forecast = m.predict(future)

# Show and plot forecast
_, col2, _ = st.columns([1, 2 , 1])

with col2:
    st.title("Forecast data")
st.write(forecast.tail())

st.write(f'Forecast plot for {n_years} years')
fig1 = plot_plotly(m, forecast)
st.plotly_chart(fig1)
st.text("")
fig2 = m.plot_components(forecast)
st.write(fig2)
```

Fig 4.2.2 Method Psuedo Code-2

```
def text_downloader(raw_text):
    b64=base64.b64encode(raw_text.encode()).decode()
    new_filename="new_text_file){}.txt".format(timestr)
    st.markdown("## Download File ##")
    href=f'<a href="data:file/txt;base64,{b64}" download="{new_filename}"> Click Here!! </a>'
    st.markdown(href,unsafe_allow_html=True)



my_text=st.text_area("Your message:")
if st.button("Save"):
    st.write(my_text)
    text_downloader(my_text)



end=time.time()

st.text("" )
hostname = socket.gethostname()
IPAddr = socket.gethostbyname(hostname)
st.write("Your IP Address is:",IPAddr)
st.text(" ")
st.write("Time taken to run the program is:", end-start,"seconds..")

```

Fig 4.2.3 Method Psuedo Code-3

4.3 IMPLEMENTATION FLOW

- STEPS for the Project Implementation:

1. Select the stock of your choice (should be listed in drop-down menu using yfinance)
2. Select the no of years of stock prediction you want.
3. After that the data will be taken from dataset(yfinance) and table and graph will be plotted.
4. This data will be passed to fbprpohet for the stock price prediction.
5. The forecasted data will be displayed in the form of table and graph.

CHAPTER 5 TEST CASES

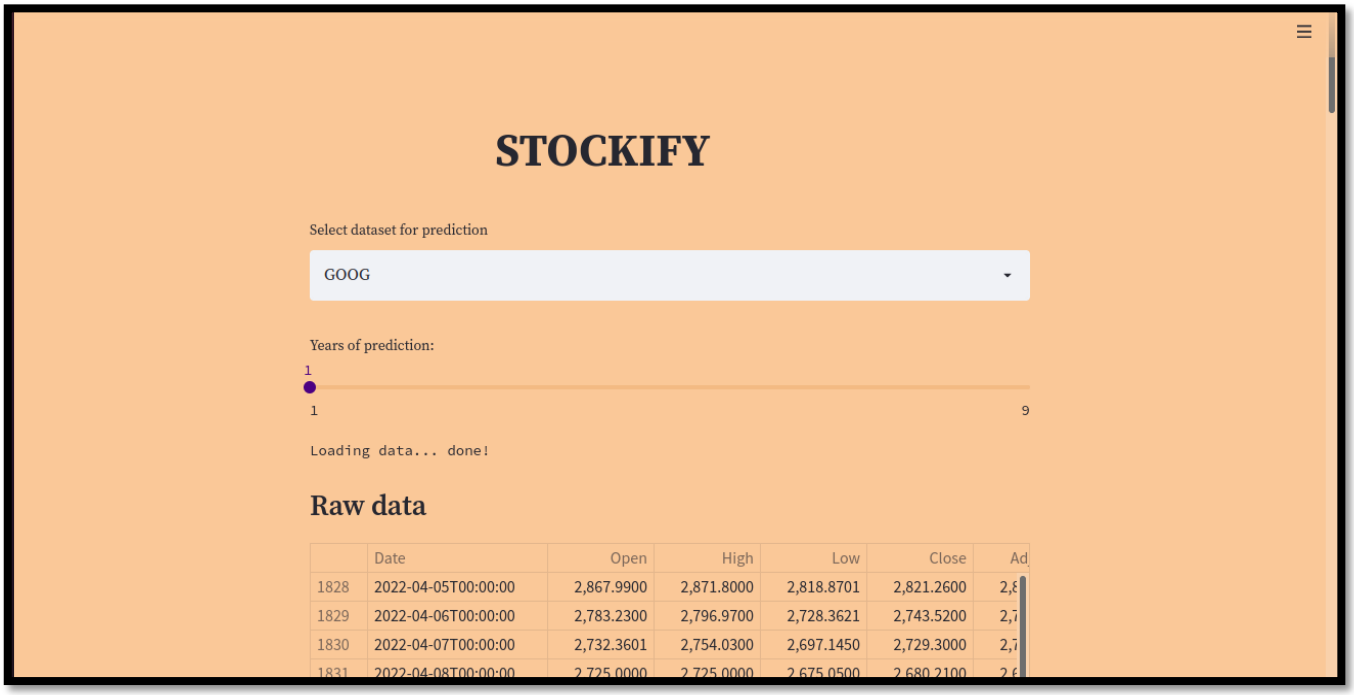


Fig 5.1 Test Case image-1

Fig 5.2 Test Case image-2

CHAPTER 6 SYSTEM DIAGRAMS

6.1 ENTITY COMPONENT DIAGRAM

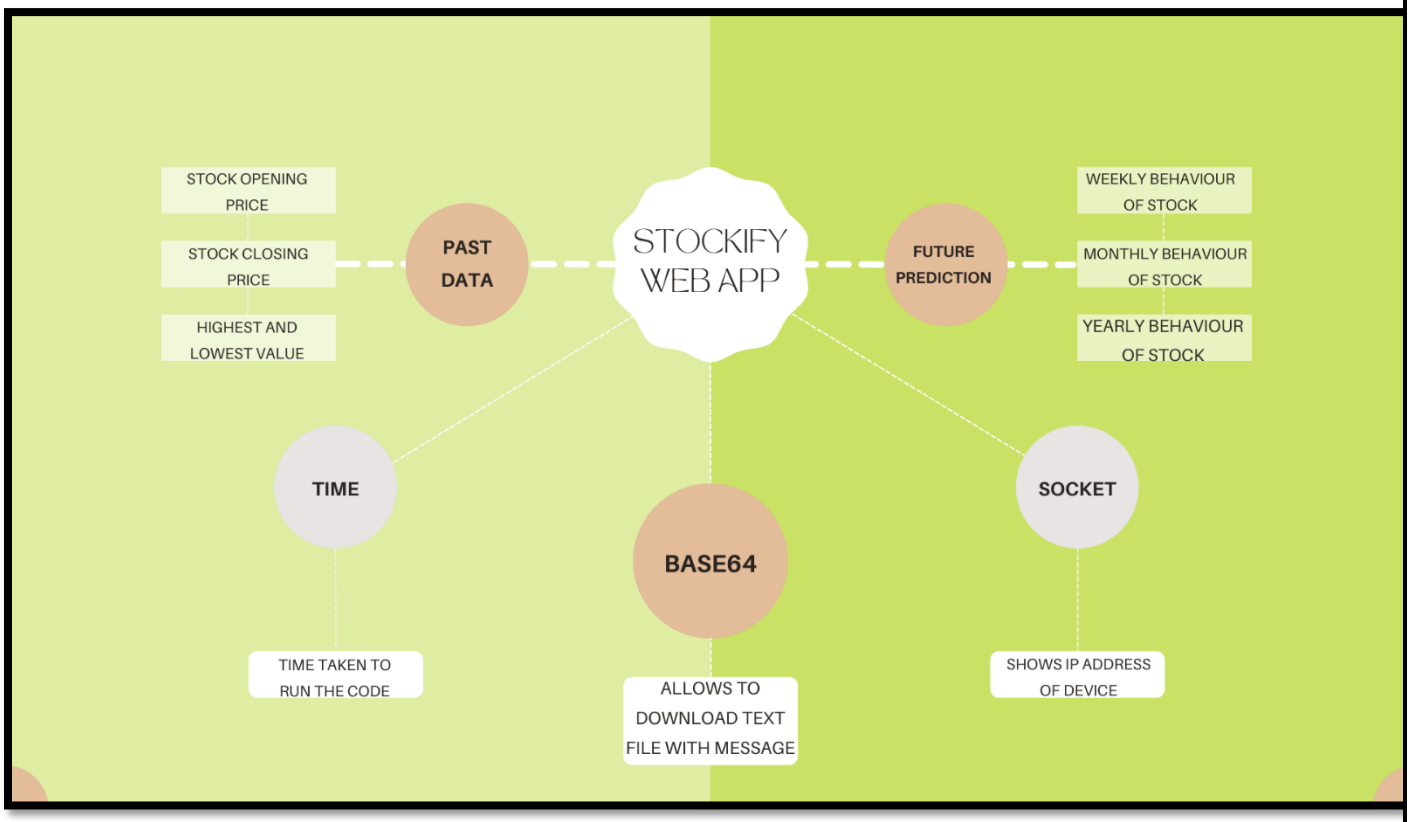


Fig 6.1 Entity Component Diagram

6.2 WORK FLOW DIAGRAM

Methodology

Stock market prediction seems a complex problem because there are many factors that have yet to be addressed and it doesn't seem statistical at first. But by proper use of machine learning techniques, one can relate previous data to the current data and train the machine to learn from it and make appropriate assumptions. Machine learning as such has many models but this paper focuses on two most important of them and made the predictions using them.

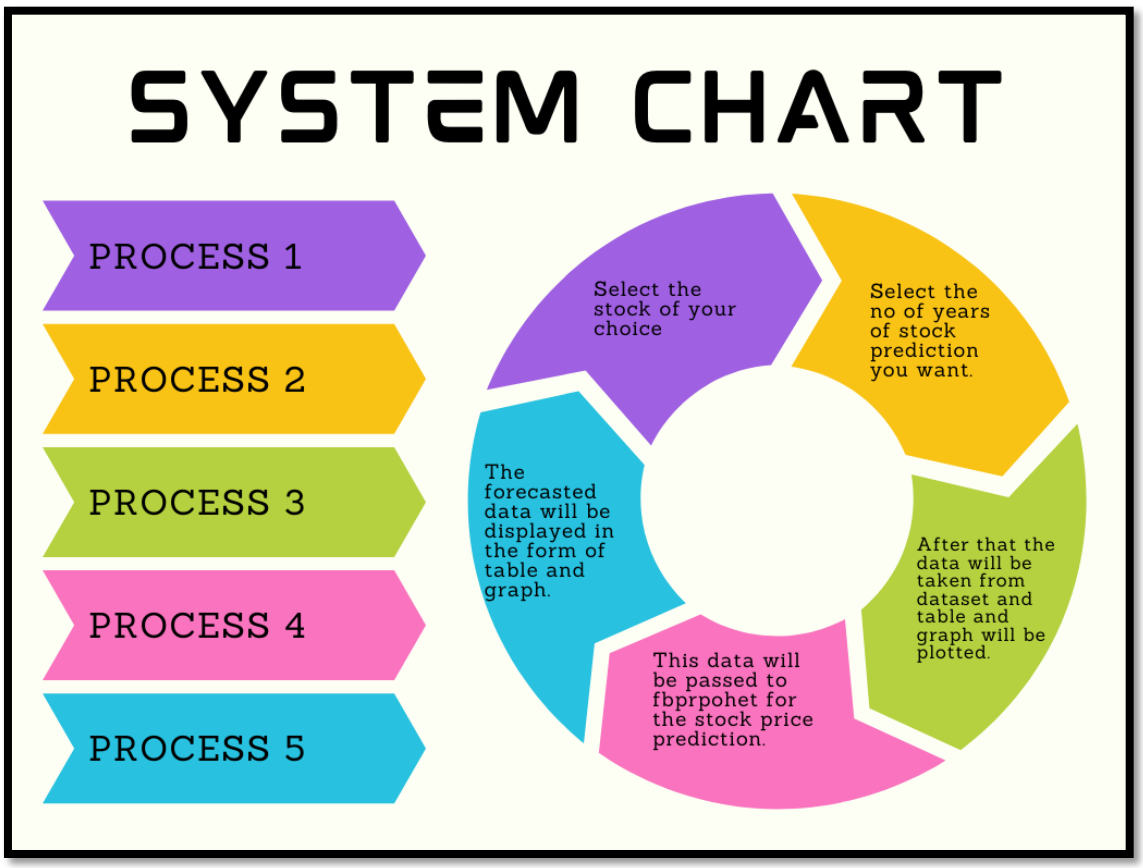


Fig 6.2 System Chart

6.3 USE CASE DIAGRAM

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

- Scenarios in which your system or application interacts with people, organizations, or external systems.
- Goals that your system or application helps those entities (known as actors) achieve.
- The scope of your system.

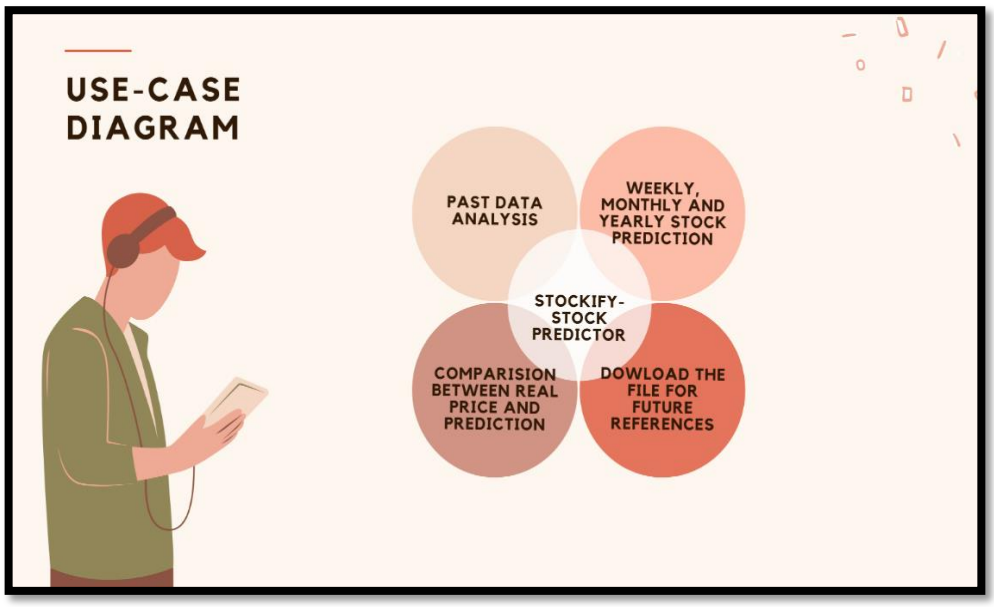


Fig 6.3 Use Case Diagram

CHAPTER 7 FUTURE ENHANCEMENT

In report, we will compare a machine learning models like LSTM model, the CNN model and also the hybrid approach of LSTM + CNN model. We have a tendency to train the model using the data of NSE listed companies to predict the stock future value. This is shows the proposed method is capable to distinctive around interrelation with the data. Also, it is evident from the results that, Hybrid approach of LSTM+CNN model is capable to identify the changes in trends. For the projected method the Hybrid approach of LSTM+CNN is known as the best model. It uses the information given at a specific instant for prediction. Even if the other two models LSTM and CNN are utilized in a lot of other time-dependent data analysis, it is not outperforming over the Hybrid approach of LSTM+CNN architecture in this case. This is often because of quick changes occur in stock market. The changes in the stock market is not always be in a regular pattern or not always follow the continuous cycle. Based on the companies and sectors, the existence of the trends and the period of their existence will differ. The analysis of this type of cycles and trends can offer a more profit to the investors. In future work, we add more stock market data and compare more model to improve accuracy of predicted stock price. In the future, for better accuracy model can be trained with more varied and detailed data. Also, other algorithms along with proposed can be used to create a new hybrid model. We want to extend this application for predicting cryptocurrency trading. We want to add sentiment analysis for better analysis.

7.1 LIMITATIONS OF THE OUR PROJECT

While most of the previous research in this field were concentrating on techniques to forecast stock price based on the historical numerical data such as past stock trends, there is not much research put into the textual analysis side of it. News and media has huge influence on human beings and the decisions we take. Also, fluctuations in the stock market are a result of the trading activities of human beings. As news articles influence our decisions and as our decisions influence the market, the news indirectly influence the stock market. Therefore, extracting information from news articles may yield better results in predicting the stock prices. News sensitive stock trend prediction, sentiment polarity analysis using a cohesion based approach , mining of text concurrent text and time series are some of the distinguished works in the domain. However, there are few issues in the above-mentioned works. First being many of these researches used Bag of Words (BoW) approach to extract information from the news reports, despite of the fact that the BoW approach cannot capture some important linguistic characteristics such as word ordering, synonyms and variant identification. Next, most works either used just the numerical information or textual information, whereas the market analysts used both. Also, previous approaches did not consider the fact that the stock prices between the companies is correlational. Finally, the approaches which used textual information to forecast, did not consider stock prices as time series. If the stock is not listed in dataset then we can't predict it. If any drastic change occurs in Stock Price due to abnormalities , we can't predict. Still our accuracy is somewhat less

CHAPTER 8 CONCLUSION

8.1 CONCLUSION

1. In this project, we are predicting closing stock price of any given organization.
2. We have applied datasets belonging to Google, Nifty50, TCS, Infosys and Reliance Stocks and achieved above 95% accuracy for these datasets.
3. It helps to raise capital for business, mobilize savings for investment, facilitates the growth of companies, and enables profit sharing.
4. It assists in creating investment opportunities for small investors, and raising capital for development projects taken up by the government. It acts as a barometer of the economy.
5. They perform several functions in markets, including efficient price discovery and efficient dealing.

CHAPTER 9 REFERENCES

9.1 REFERENCES

1. https://machinelearningmastery.com/start- here/#getstarted
2. https://www.sas.com/en_us/insights/analytics/machine-learning.html
3. https://groww.in/us-stocks/googl
4. https://www.investopedia.com/terms/d/deep- learning.asp
5. https://www.nasdaq.com/market-activity/stocks/goog
6. http://www.jstor.org/stable/2352710.
7. http://www.nber.org/papers/w0564.
8. http://www.sciencedirect.com/science/article/pii/S2214579615000027
9. http://www.sciencedirect.com/science/article/pii/S0957417415005126.