STOCK MARKET PRICE PREDICTION

A Mini Project report

Submitted in the partial fulfillment of the requirements for

the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Rayapudi Naveen
19761A05B1

Under the guidance of
Guide name with Designation
Dr.K.Vijay Bhaskar Reddy
Associate Professor

Department of Computer Science and Engineering
Lakireddy Bali Reddy College of Engineering (Autonomous)
Accredited by NAAC & NBA (Under Tier - I)
Affiliated to JNTUK, Kakinada; ISO 9001:2015 Certified 2021-2022

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(AUTONOMOUS)
Accredited by NAAC & NBA (Under Tier - I) ISO 9001:2015 Certified Institution
Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada
L.B. REDDY NAGAR, MYLAVARAM, KRISHNA DIST., A.P.-521 230.
http://cse.lbrce.ac.in, cselbreddy@gmail.com, Phone: 08659-222933, Fax: 08659-222931

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Certificate

This is to certify that the project entitled “STOCK MARKET PRICE PREDICTION” is being submitted by **Rayapudi Naveen** in partial fulfillment for the award of B.Tech in Computer Science & Engineering to the Jawaharlal Nehru Technological University Kakinada is a record of bonafide work carried out by her under our guidance.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma.

Project guide	**Head of the department**
Dr.K.Vijay Bhaskar Reddy	Dr.D.Veeraiah
Associate Professor	Professor

External Examiner

ACKNOWLEDGEMENT

I would like to thank **Dr.K.Vijay Bhaskar Reddy,** Associate Professor**,** CSE department for the encouraging and support in carrying out this project.

I would like to thank Mini-Project In-charge Dr.**A.Raja Gopal**, Sr.Assistant Professor**, CSE department** for the encouraging and support in carrying out this project.

I also take the privilege to record my thanks to **Dr. D.Veeraiah,** Head of the Department of CSE whose encouragement, cooperation and valuable support crown our success.

I express my thanks to the support given by management in completing my thesis. I also express my sincere gratitude & deep sense of respect to the Principal, **Dr.K.Appa Rao** for making us available all the required assistance and his support and inspiration to carry out this project in the Institute.

I am thankful to the teaching and non-teaching staff of CSE department for their direct as well as indirect help in my project.

I am elated to avail my selves to this opportunity to express my deep sense of gratitude to my parents.

STOCK MARKET PRICE PREDICTION

Abstract

► In Stock Market Price Prediction, the aim is to predict the future value of the financial stocks of a company. The recent trend in stock market prediction technologies is the use of machine learning which makes predictions based on the values of current stock market indices by training on their previous values. Machine learning itself employs different models to make prediction easier and authentic. The paper focuses on the use of Regression and LSTM based Machine learning to predict stock values. Factors considered are open, close, low, high and volume. The database connectivity is planned using the Sqlit3 Connection‖ methodology. The standards of security and data protective mechanism have been given a big choice for proper usage. The application takes care of different modules and their associated reports, which are produced as per the applicable strategies and standards that are put forwarded by the administrative staff.

CONTENTS

1.INTRODUCTION

Stock market prediction and analysis are some of the most difficult jobs to complete. There are numerous causes for this, including market volatility and a variety of other dependent and independent variables that influence the value of a certain stock in the market. These variables make it extremely difficult for any stock market expert to anticipate the rise and fall of the market with great precision. However, with the introduction of Machine Learning and its strong algorithms, the most recent market research and Stock Market Prediction advancements have begun to include such approaches in analyzing stock market data. summary, Machine learning Algorithms are widely utilized by many organizations in Stock market prediction. This article will walk through a simple implementation of analyzing and forecasting the stock prices of a Popular Worldwide Online Retail Store in Python using various Machine Learning Algorithms.

1.2 AIM

In stock market the act of trying to determine the future value of a company stock or financial both traded on an exchange. The successful prediction of a stock's future price could yield significant profit. The efficient market hypothasis suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information thus are inherently unpredictable. Others disagree and those with this viewpoint possess myriad methods and technologies which purportedly allow them to gain future price information.

1.3 EXISTING SYSTEM

In existing system, we tend to propose that a company's performance, in terms of its stock worth movement, is foreseen by internal communication patterns. to get early warning signals, we tend to believe that it's vital for patterns in company communication networks to be detected earlier for the pre- diction of serious stock worth movement to avoid attainable adversities that organization could face within the securities market in order that stakeholders' interests is protected the maximum amount as attainable. Despite the potential importance of such data regarding corporate communication, very little work has been tired this vital direction. We attempt to bridge these research gaps by employing a data-mining method to examine the linkage between a firm's communication data and its share price. As Enron Corporation's e-mail messages constitute the only corpus available to the public, we make use of Enron's e-mail corpus as the training and testing data for our proposed algorithm.

1.4 PROPOSED SYSTEM

In this proposed system, we focus on predicting the stock values using machine learning algorithms like Random Forest and Support Vector Machines. We proposed the system "Stock market price prediction" we have predicted the stock market price using the random forest algorithm. In this proposed system, we were able to train the machine from the various data points from the past to make a future prediction. We took data from the previous year stocks to train the model. We majorly used two machine-learning libraries to solve the problem. The first one was numpy, which was used to clean and manipulate the data, and getting it into a form ready for analysis. The other was scikit, which was used for real analysis and prediction. The data set we used was from the previous years stock markets collected from the public database available online, 80 % of data was used to train the machine and the rest 20 % to test the data. The basic approach of the supervised learning model is to learn the patterns and relationships in the data from the training set and then reproduce them for the test data. We used the python pandas library for data processing which combined different datasets into a data frame. The tuned up dataframe allowed us to prepare the data for feature extraction. The dataframe features were date and the closing price for a particular day. We used all these features to train the machine on random forest model and predicted the object variable, which is the price for a given day. We also quantified the accuracy by using the predictions for the test set and the actual values. The proposed system touches different areas of research including data pre-processing, random forest, and so on.

2.WORKING

2.1 REQUIREMENTS SPECIFICATION

SOFTWARE REQUIREMENTS :

- Operating System : Window XP, Windows 7
- Python 3.5 in idle python is used for data pre-processing, model training and prediction.

HARDWARE REQUIREMENTS :

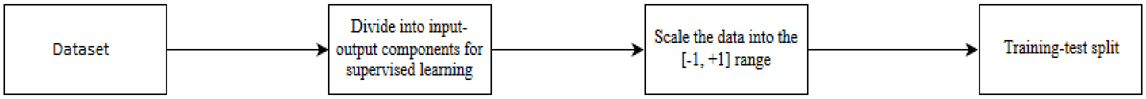
- Hard Disk Drive : 40 GB
- Processor : Intel i5,2.4 GHz
- RAM : 4 GB

TECHNOLOGIES USED:

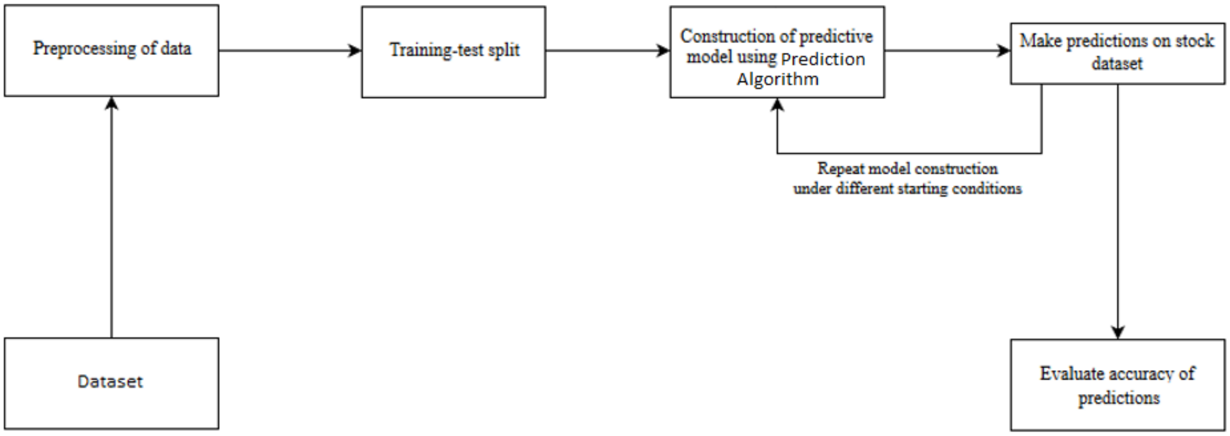
- Tensor flow
- Yahoo finance
- CSS
- Keras
- HTML
- Sqlite3
- streamlit

2.2 SYSTEM ARCHITECTURE:

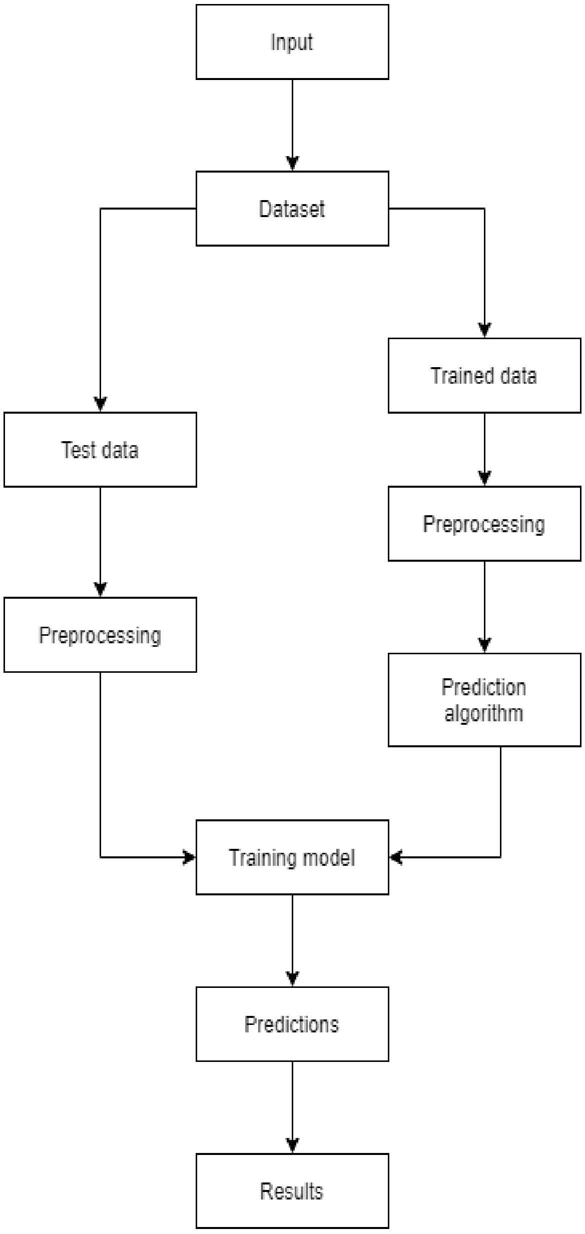
1) Preprocessing of data


Dataset
Divide into input-output components for supervised learning
Scale the data into the [-1, +1] range
Training-test split

2) Overall Architecture


Preprocessing of data
Training-test split
Construction of predictive model using Prediction Algorithm
Make predictions on stock dataset
Repeat model construction under different starting conditions
Dataset
Evaluate accuracy of predictions

2.3 PROJECT DESIGN



2.4 UML Diagrams

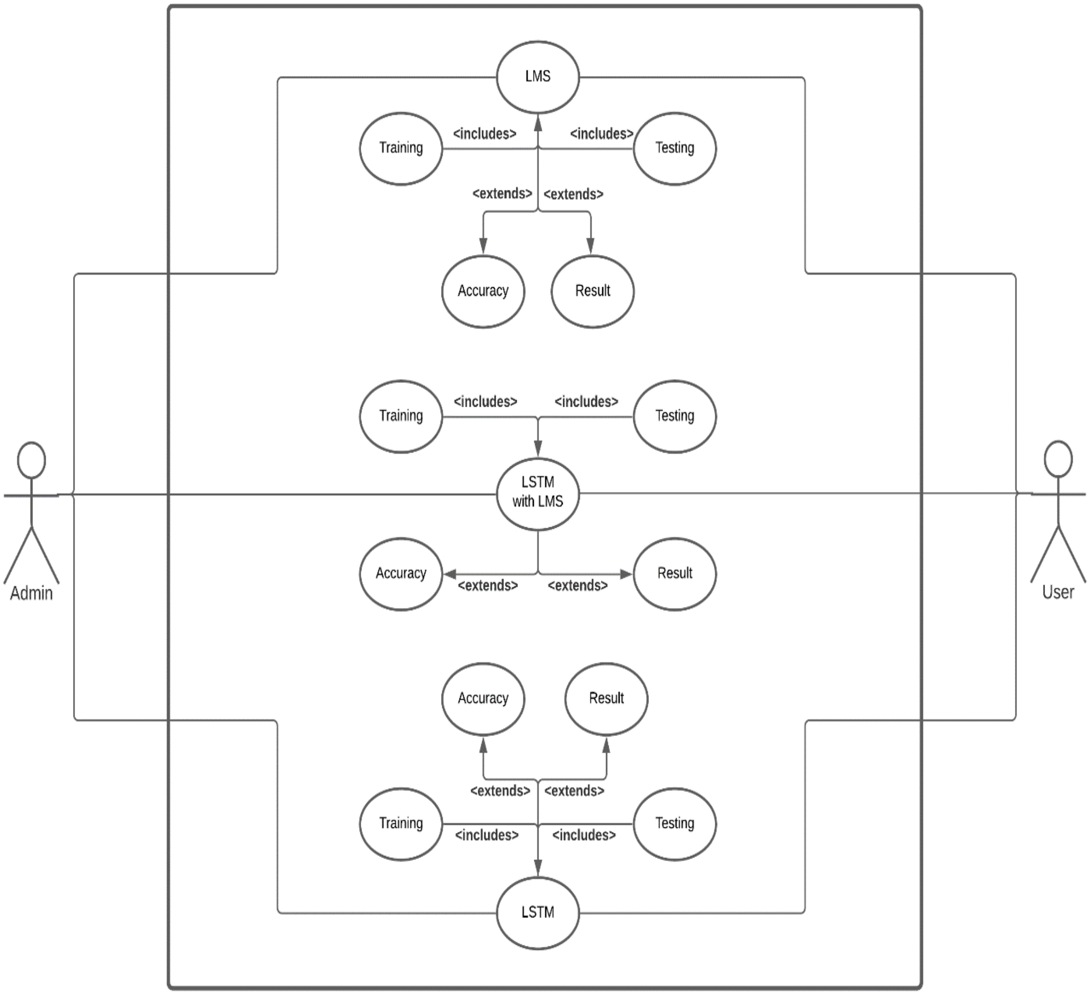
USECASE DIAGRAM:

Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

• The purpose is to show the interactions between the use case and actor.

• To represent the system requirements from user's perspective.

• An actor could be the end-user of the system or an external system.

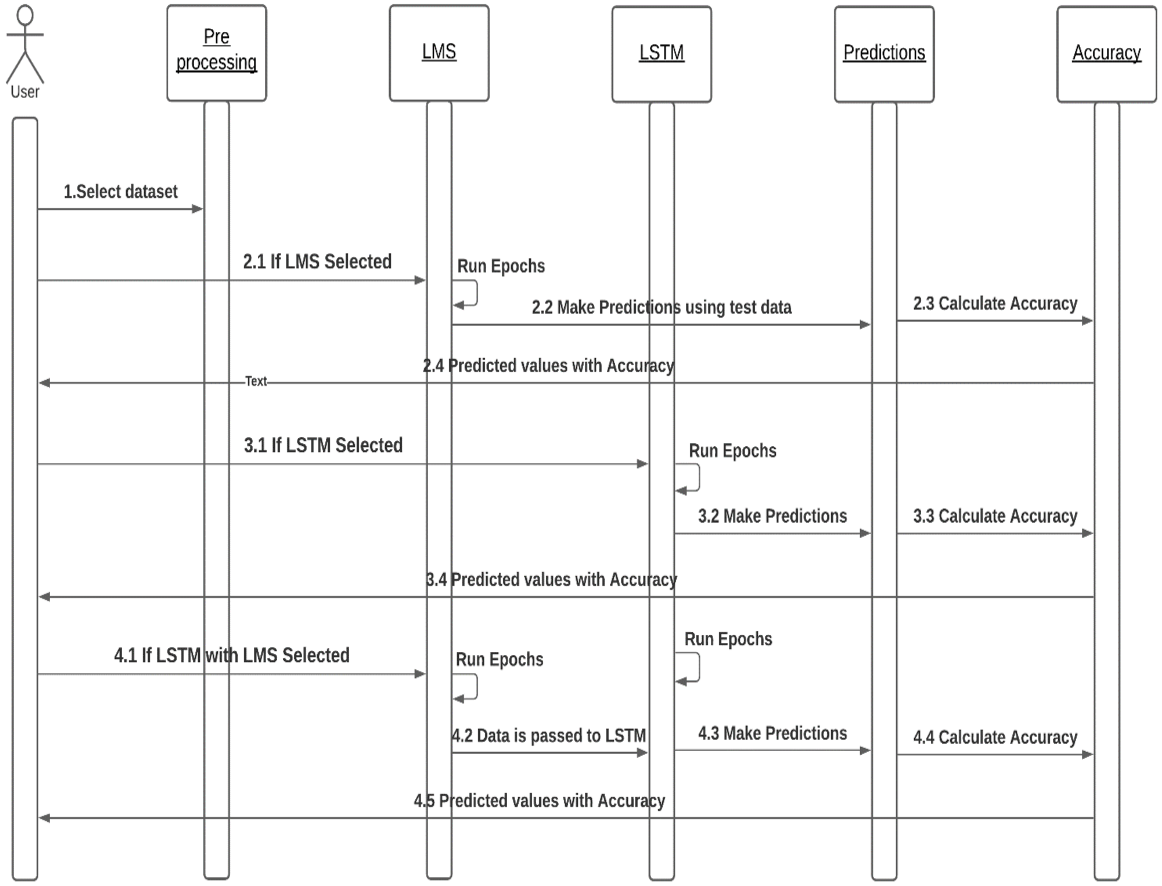


Sequence Diagram :

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

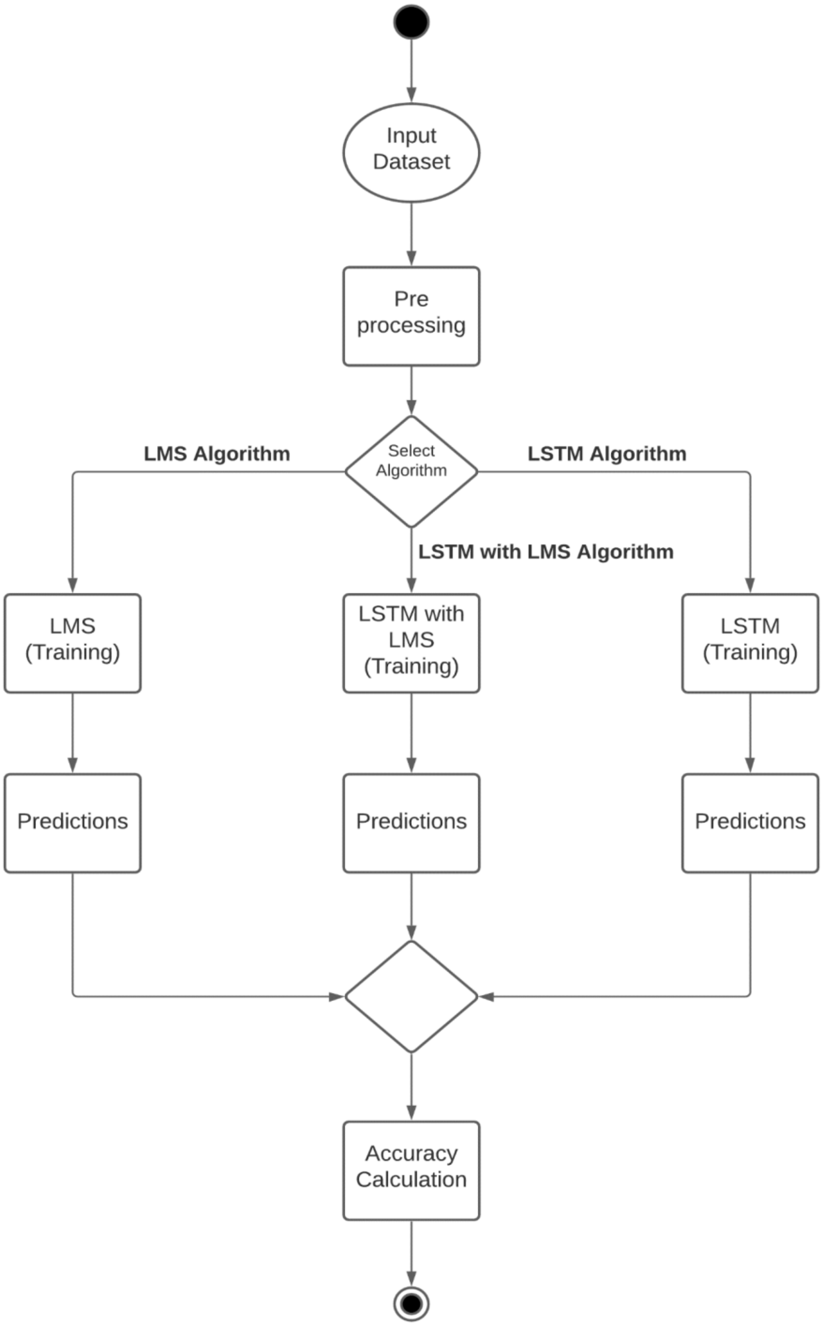
Sequence diagrams can be useful references for businesses and other organizations. Try drawing a sequence diagram to:

- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.



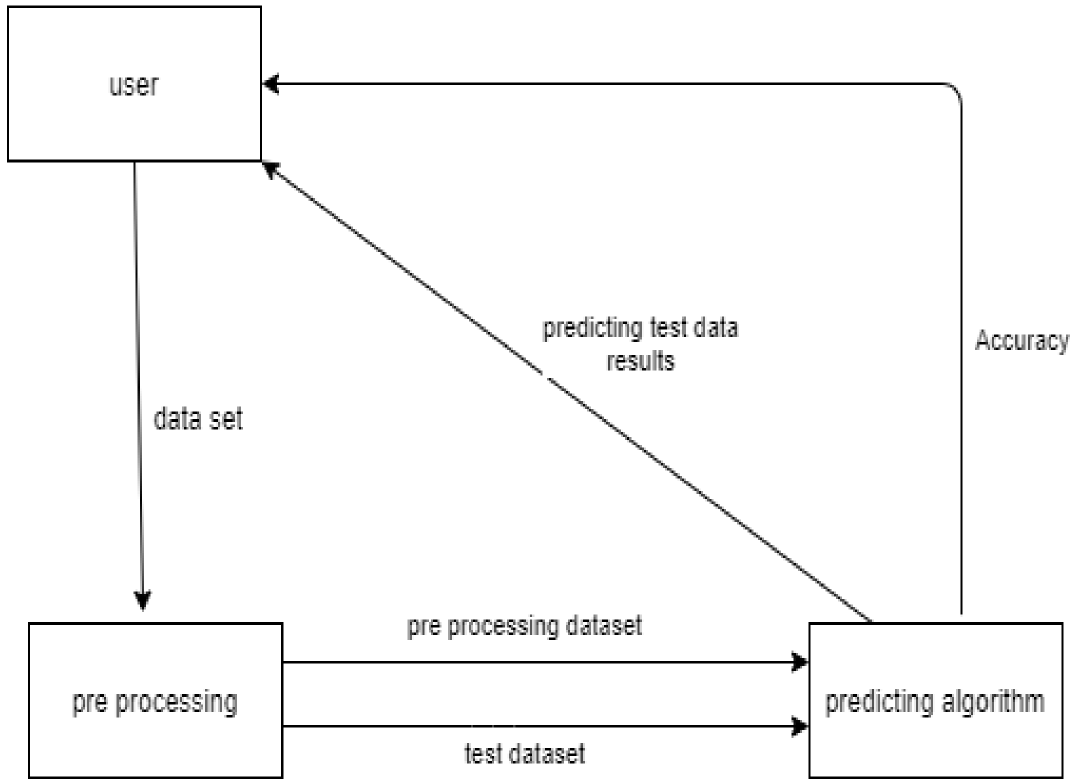
Activity Diagram :

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.



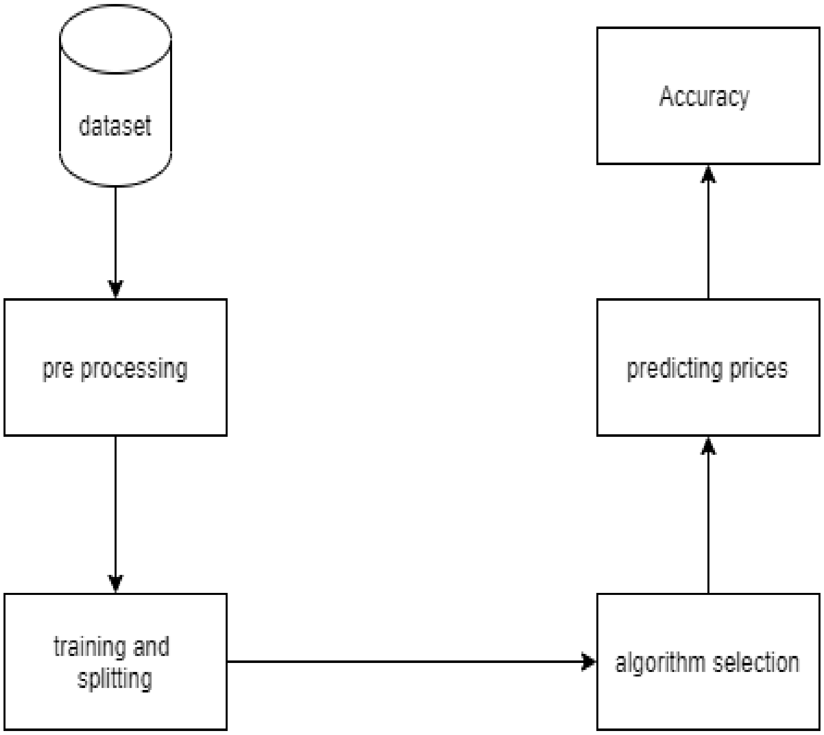
Collaboration Diagram :

The collaborations are used when it is essential to depict the relationship between the object. Both the sequence and collaboration diagrams represent the same information, but the way of portraying it quite different. The collaboration diagrams are best suited for analyzing use cases



Component Diagram :

Component diagrams are used in modeling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.



OVERVIEW OF TECHNOLOGIES USED:

HTML:

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements

Yahoo finance:

- Finance is a media property that is part of the Yahoo! network.
- It provides financial news, data and commentary including stock quotes, press releases, financial reports, and original content.
- It also offers some online tools for personal finance management.

Tensor flow:

- TensorFlow provides a collection of workflows to develop and train models using Python or JavaScript,
- It generates output predictions for the input samples.
- A tensor is a generalization of vectors and matrices to potentially higher dimension

- **Keras:**

  - Keras is used for creating deep models which can be productized on smartphones.
  - Keras is a neural network Application Programming Interface (API) for Python that is tightly integrated with TensorFlow, which is used to build machine learning models.
  - Keras' models offer a simple, user-friendly way to define a neural network, which will then be built for you by TensorFlow.

Python:

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.
It is used for:

- web development (server-side),
- software development,
- system scripting.

Sqlite3:

- SQLite3 allows us to access a database using SQL that doesn't require a separate server.
- sqlite3 module is a powerful part of the Python standard library
- SQLite3 is a very easy to use database engine.
- It lets us work with a fully featured on-disk SQL database without installing any additional software

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.
- Streamlit allows you to write an app the same way you write a python code.
- Streamlit is a free, open-source, all-python framework that enables data scientists to quickly build interactive dashboards and machine learning web apps with no front-end web development experience required.
- The best thing about Streamlit is it doesn't require any knowledge of web development.

3.MODULES

3.1 Admin Module:

This module provides administrator related functionalities. The administrator manages the entire application and maintains the profiles of applicants

3.2 User Module:

Home: In this section, users know what the web application consists of stock Market

Web application: It gives the information about certain predicted price of a company that was given input by the user

About: It gives the information about reason for implement of the web app

Symbol search: It gives the information about companies symbols that are used in web application

4. IMPLEMENTATION

4.1 CODING

SOURCE CODE:

```
import streamlit as st
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pandas_datareader as data
from keras.models import load_model
from keras.models import Sequential
from keras.layers import Dense,Dropout,LSTM
import streamlit as st
from PIL import Image
import streamlit.components.v1 as components
st.set_page_config(
                page_title="Stock Market Web App",
                page_icon="C:/idlepython/download1.jpg",
                layout="wide",
                initial_sidebar_state="expanded",


                 )
```

```
# Security
#passlib,hashlib,bcrypt,scrypt
import hashlib
def make_hashes(password):
    return hashlib.sha256(str.encode(password)).hexdigest()


def check_hashes(password,hashed_text):
    if make_hashes(password) == hashed_text:
        return hashed_text
    return False
# DB Management
import sqlite3
conn = sqlite3.connect('data.db')
c = conn.cursor()
# DB  Functions
def create_usertable():
    c.execute('CREATE TABLE IF NOT EXISTS userstable(username TEXT,password TEXT)')
def add_userdata(username,password):
    c.execute('INSERT INTO userstable(username,password) VALUES (?,?)',(username,password))
    conn.commit()


def login_user(username,password):
      c.execute('SELECT * FROM userstable WHERE username =? AND password = ?',(username,password))

      data = c.fetchall()

      return data

def view_all_users():

      c.execute('SELECT * FROM userstable')

      data = c.fetchall()

      return data

def main():



    st.sidebar.image("/idlepython/download1.jpg", use_column_width=True)

    menu = ["Home","Login","SignUp"]

    choice = st.sidebar.selectbox("Menu",menu)



    if choice == "Home":

        st.markdown('<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">', unsafe_allow_html=True)

        st.markdown("""

<nav class="navbar fixed-top navbar-expand-lg navbar-dark" style="background-color: #3498DB;">

 <a class="navbar-brand" href="#">Home</a>
```

```
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarNav">
    <ul class="navbar-nav">
      <li class="nav-item active">
        <a class="nav-link" href="https://www.nasdaq.com/market-activity/stocks/screener1" target="_blank">SymbolSearch</a>
      </li>
      <li class="nav-item active">
        <a class="nav-link" href="https://www.businessinsider.com/personal-finance/best-stock-trading-apps?IR=T" target="_blank">About</a>
      </li>
    </ul>
  </div>
</nav>
""", unsafe_allow_html=True)
st.markdown("<h1 style='text-align: center; color: white;'>Stock Market Prediction Web App </h1>", unsafe_allow_html=True)
                image=Image.open("C:/idlepython/project/348545152_8a4e55ff2e_b.jpg")
                st.image(image, use_column_width=True,width="2000")


                components.html("""
                        <h2><p style="color:white">Content Available:</p></h2>


                        <h3><p style="color:white">-->closing price and Timechart with 50 Moving
Average and with 50 Moving Average and 100 Moving Average</p><h3>
                        <h3><p style="color:white">-->Original price and Predicted price
chart</p></h3>


                        """)
                components.html("""    <p style="color:red">By default it shows google
predictions</p>""")


                st.subheader("Sample Data visualaization:")


                st.subheader("Data Information")


                image=Image.open("C:/Users/rayapudi naveen/OneDrive/Desktop/Screenshot 1.png")
                st.image(image, use_column_width=True,width="None")


                st.subheader("Closing and TimeChart")
                image=Image.open("C:/Users/rayapudi naveen/OneDrive/Desktop/Screenshot 2.png")
                st.image(image, use_column_width=True,width="None")


                st.subheader("Moving Average 50")
```

```
          image=Image.open("C:/Users/rayapudi naveen/OneDrive/Desktop/Screenshot 3.png")
        st.image(image, use_column_width=True,width="None")


        st.subheader("Original price and Predicted price")
        image=Image.open("C:/Users/rayapudi naveen/OneDrive/Desktop/Screenshot 4.png")
        st.image(image, use_column_width=True,width="None")


    elif choice == "Login":
        st.subheader("Login Section")


        username = st.sidebar.text_input("User Name")
        password = st.sidebar.text_input("Password",type='password')


        if st.sidebar.button("Login"):
                  # if password == '12345':
            create_usertable()
            hashed_pswd = make_hashes(password)


            result = login_user(username,check_hashes(password,hashed_pswd))
            if result:


                st.success("Logged In as {}" .format(username))
```

```
start='2010-01-01'
end='2022-01-01'

st.title('stock Market prediction')

image=Image.open("C:/Users/rayapudi naveen/OneDrive/Desktop/stocks_BSE_Nifty_Sensex__markets_bse_nse_nifty-1_1280-770x433.jpg")
st.image(image, use_column_width=True,width="2000")
user_input=st.text_input('enter stock tick','GOOG')
df=data.DataReader(user_input,'yahoo',start,end)
#describing data
st.subheader('Information from 2010-2022')

st.write(df.describe())
col1,col2=st.columns(2)
with col1:
#visualization
    st.subheader('Closing Price vs Time chart')
    fig=plt.figure(figsize=(14,7))
    plt.plot(df.Close)
    st.pyplot(fig)

with col2:
    st.subheader('Closing Price vs Time chart with 50MA')
    MA50=df.Close.rolling(50).mean()
    fig=plt.figure(figsize=(14,7))
    plt.plot(MA50,'r')
    plt.plot(df.Close)
    st.pyplot(fig)

co1,co2=st.columns(2)
with co1:
    st.subheader('Closing Price vs Time chart with 50MA and 100MA')
    MA50=df.Close.rolling(50).mean()
    MA100=df.Close.rolling(100).mean()
    fig=plt.figure(figsize=(14,7))
    plt.plot(MA50,'r')
    plt.plot(MA100,'g')
    plt.plot(df.Close,'b')
    st.pyplot(fig)

with co2:
        #splitting data into training and testing
        data_training = pd.DataFrame(df['Close'][0:int(len(df)*0.70)])
        data_testing = pd.DataFrame(df['Close'][int(len(df)*0.70):int(len(df))])
```

```
print(data_training.shape)

print(data_testing.shape)



from sklearn.preprocessing import MinMaxScaler

scaler=MinMaxScaler(feature_range=(0,1))


data_training_array=scaler.fit_transform(data_training)


#train

x_train=[]

y_train=[]


for i in range(100, data_training_array.shape[0]):

        x_train.append(data_training_array[i-100:i])

        y_train.append(data_training_array[i,0])



x_train,y_train=np.array(x_train),np.array(y_train)



model=Sequential()
```

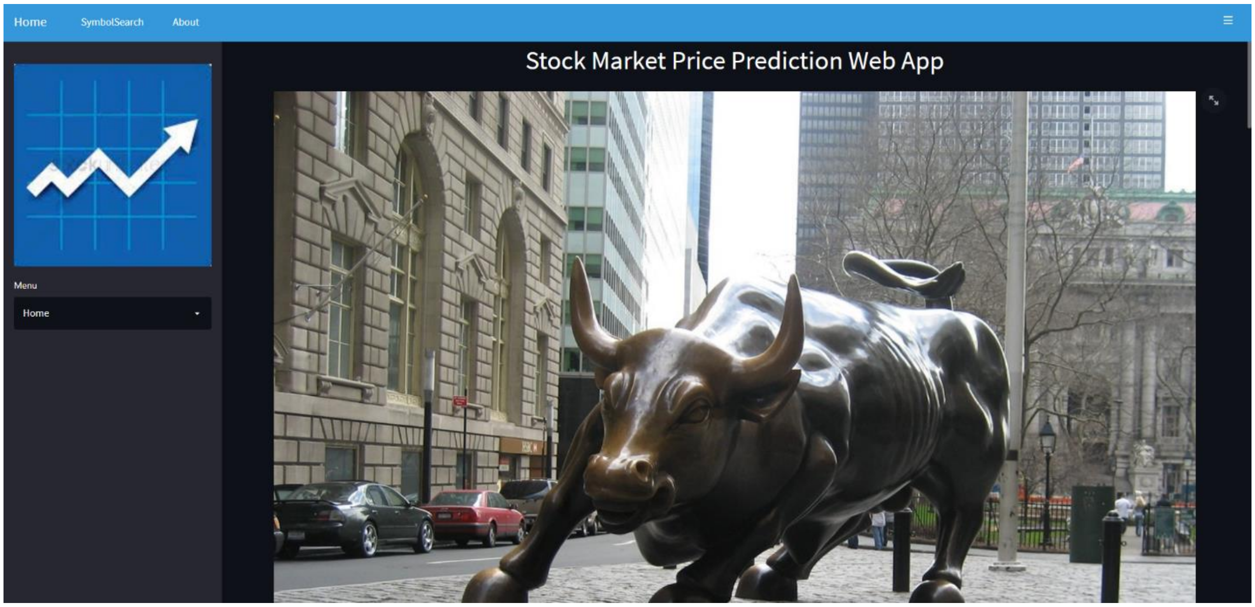
```
model.add(LSTM(units = 50
,activation='relu',return_sequences=True,input_shape=(x_train.shape[1],1)))
model.add(Dropout(0.2))


model.add(LSTM(units = 60,activation='relu',return_sequences=True))
model.add(Dropout(0.3))




model.add(LSTM(units = 80,activation='relu',return_sequences=True))
model.add(Dropout(0.4))


model.add(LSTM(units = 120,activation='relu'))
model.add(Dropout(0.5))


model.add(Dense(units =1))


model.compile(optimizer='adam',loss='mean_squared_error')
model.fit(x_train,y_train,epochs = 3)


#testing
past_100_days=data_training.tail(100)
final_df=past_100_days.append(data_testing,ignore_index=True)
```

```
input_data=scaler.fit_transform(final_df)


x_test=[]

y_test=[]


for i in range(100, data_training_array.shape[0]):

        x_test.append(data_training_array[i-100:i])

        y_test.append(data_training_array[i,0])


x_test,y_test=np.array(x_test),np.array(y_test)

y_predict=model.predict(x_test)

scaler=scaler.scale_

scale_factor =1/0.02099517

y_predict=y_predict * scale_factor

y_test=y_test * scale_factor

st.subheader('OriginelPrice vs PredictedPrice')

fig2=plt.figure(figsize=(12,6))

plt.plot(y_test,'b',label='Originel Price')

plt.plot(y_predict,'r',label='predicted Price')

plt.xlabel('Time')

plt.ylabel('Price')

plt.legend()

st.pyplot(fig2)
```

```
            else:
                st.warning("Incorrect Username/Password")
    elif choice == "SignUp”:
        st.subheader("Create New Account")
        new_user = st.text_input("Username")
        new_password = st.text_input("Password",type='password')

        if st.button("Signup"):
            create_usertable()
            add_userdata(new_user,make_hashes(new_password))
            st.success("You have successfully created a valid Account")
            st.info("Go to Login Menu to login")
if __name__ == '__main__':
      main()
```

5.RESULT

HOME PAGE

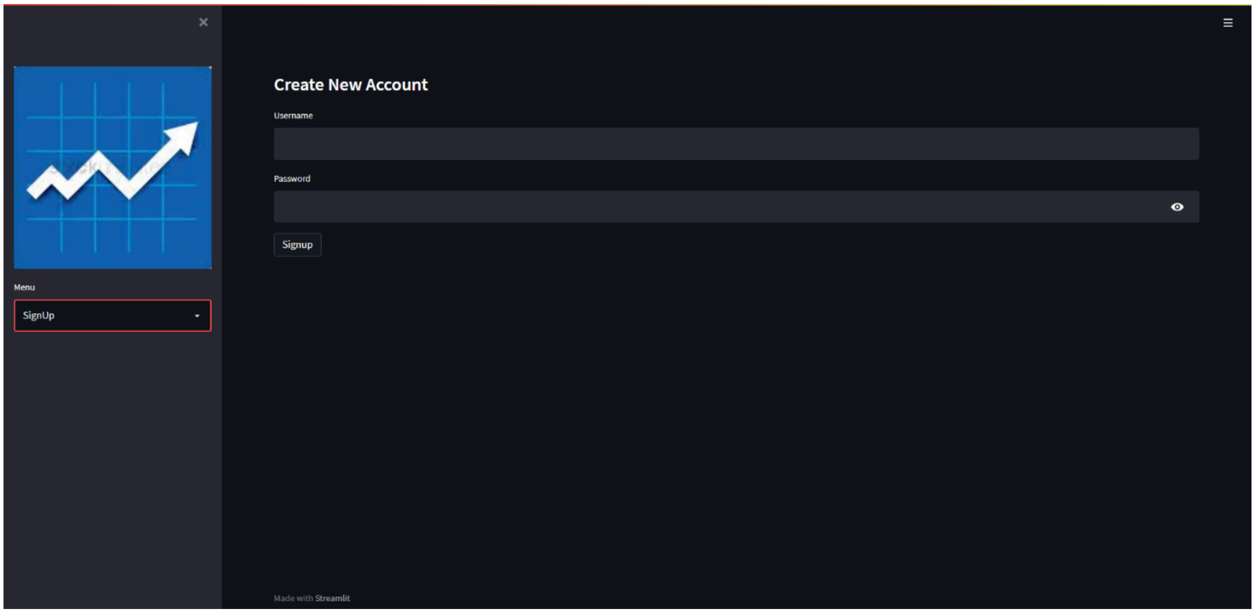


Stock Market Price Prediction Web App

Menu

Home

SINGUP SECTION


Create New Account

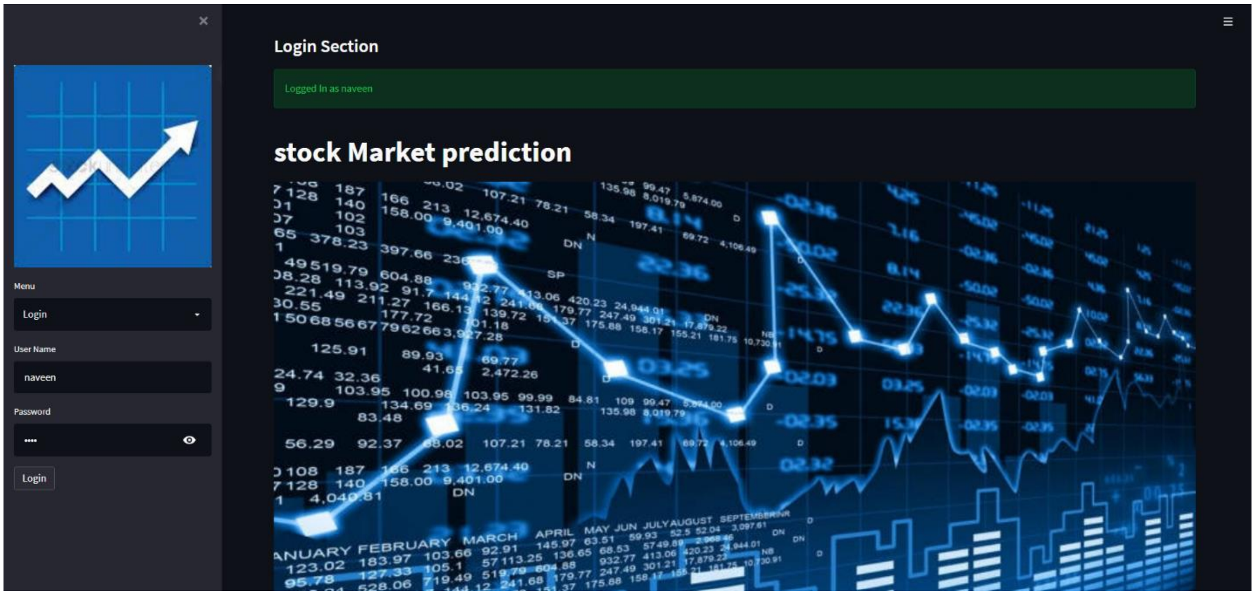
Username

Password

Menu

SignUp

LOGIN SECTION



Login Section

Logged In as naveen

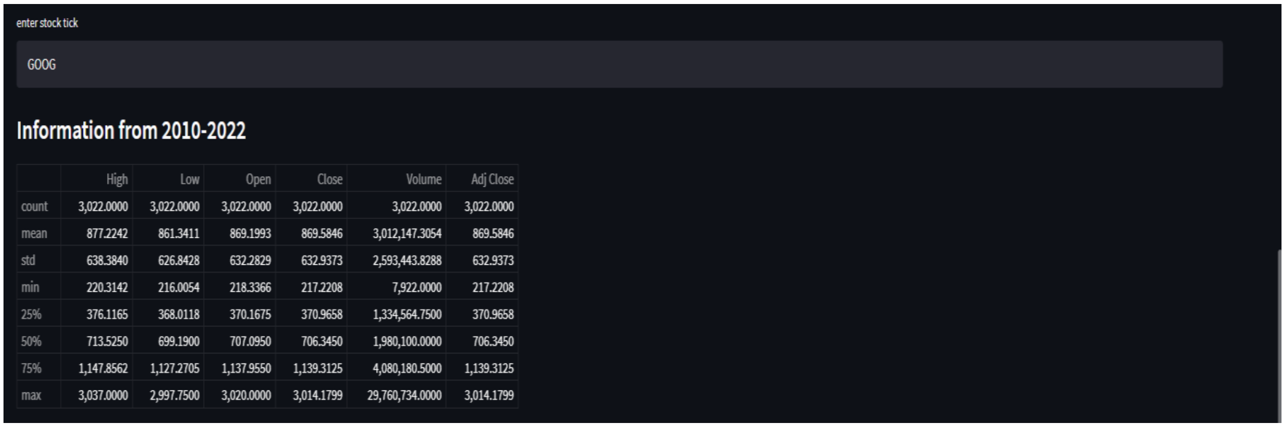
stock Market prediction

Menu: Login

User Name: naveen

Password: ••••

SEARCH BAR AND DATA INFORMATION

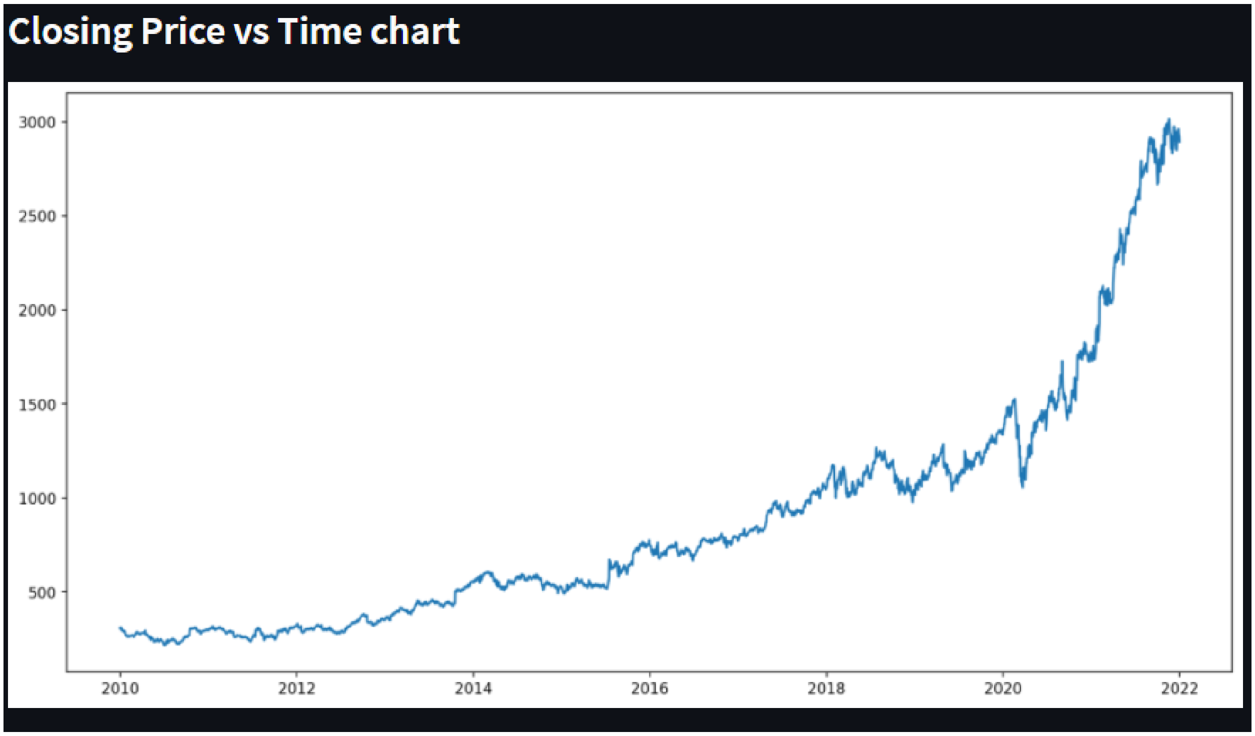


enter stock tick: GOOG

Information from 2010-2022

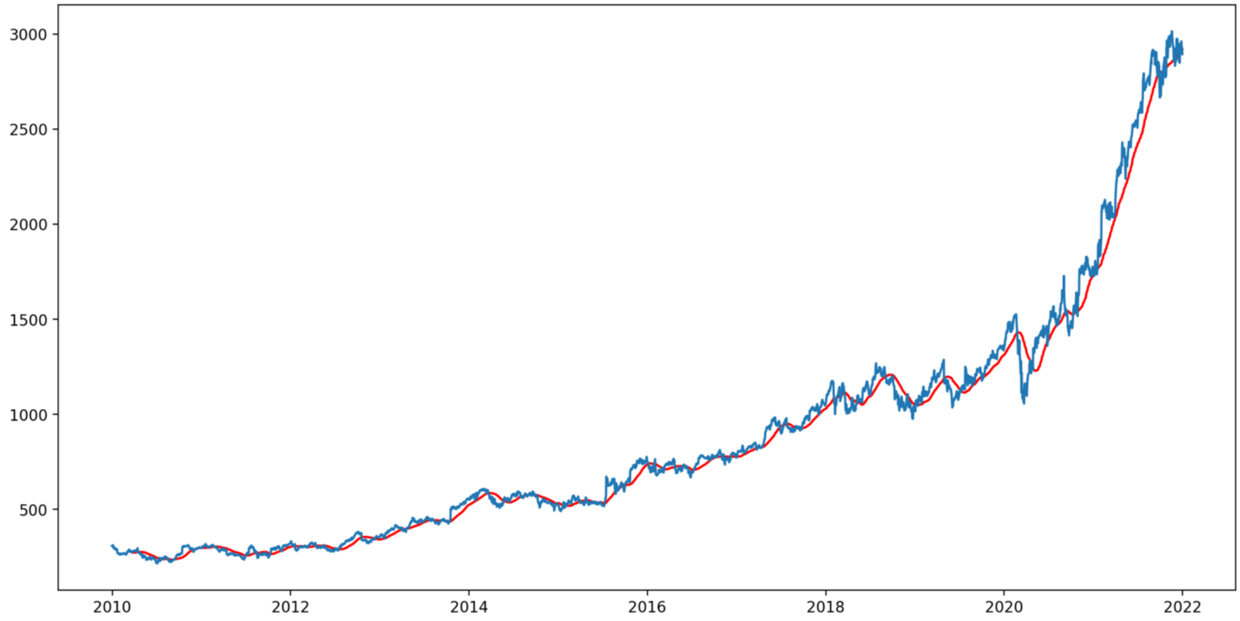
	High	Low	Open	Close	Volume	Adj Close
count	3,022.0000	3,022.0000	3,022.0000	3,022.0000	3,022.0000	3,022.0000
mean	877.2242	861.3411	869.1993	869.5846	3,012,147.3054	869.5846
std	638.3840	626.8428	632.2829	632.9373	2,593,443.8288	632.9373
min	220.3142	216.0054	218.3366	217.2208	7,922.0000	217.2208
25%	376.1165	368.0118	370.1675	370.9658	1,334,564.7500	370.9658
50%	713.5250	699.1900	707.0950	706.3450	1,980,100.0000	706.3450
75%	1,147.8562	1,127.2705	1,137.9550	1,139.3125	4,080,180.5000	1,139.3125
max	3,037.0000	2,997.7500	3,020.0000	3,014.1799	29,760,734.0000	3,014.1799

CLOSING PRICE VS TIME CHART

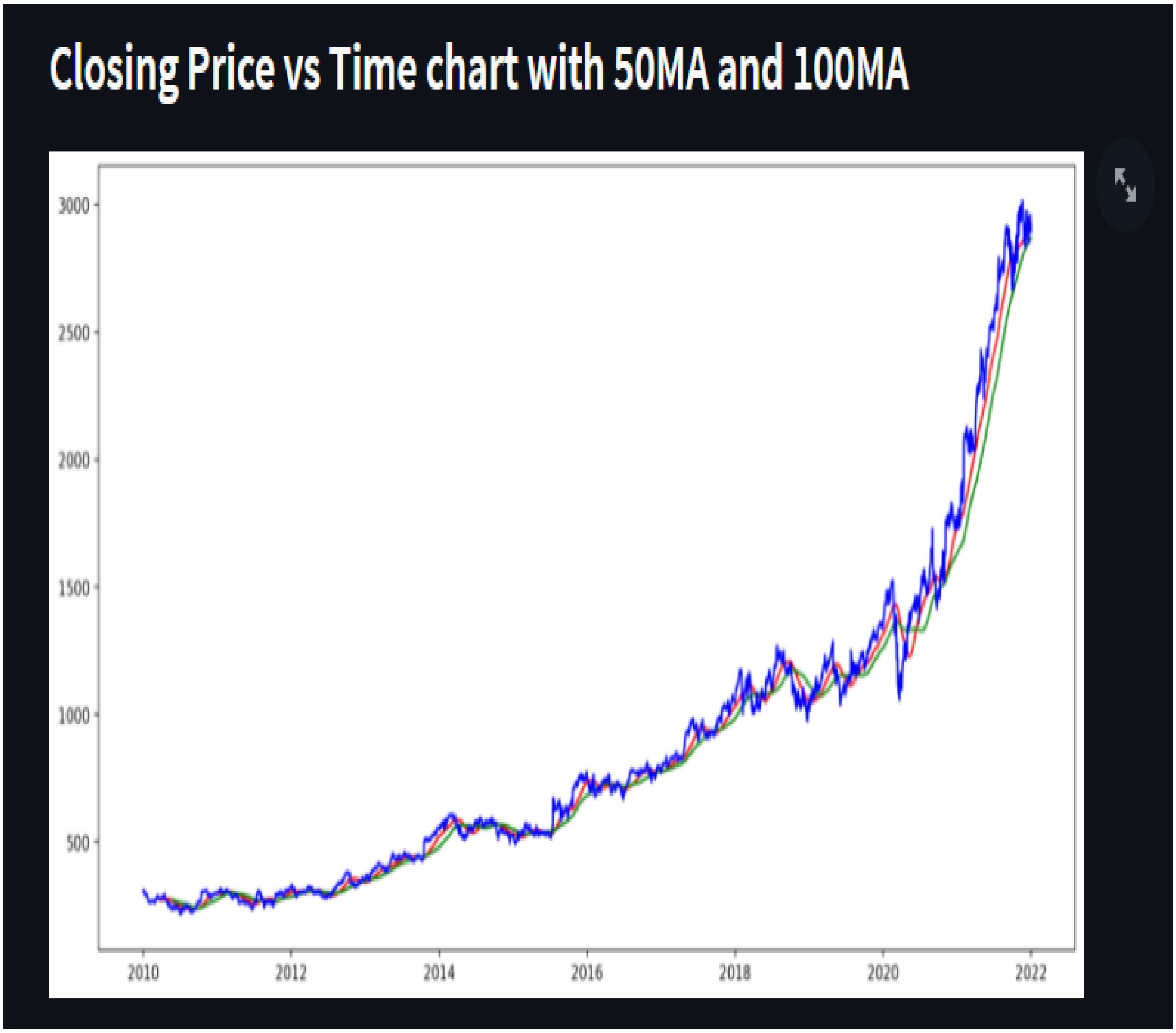


Closing Price vs Time chart

CLOSING PRICE VS TIME CHART WITH 50 MOVING

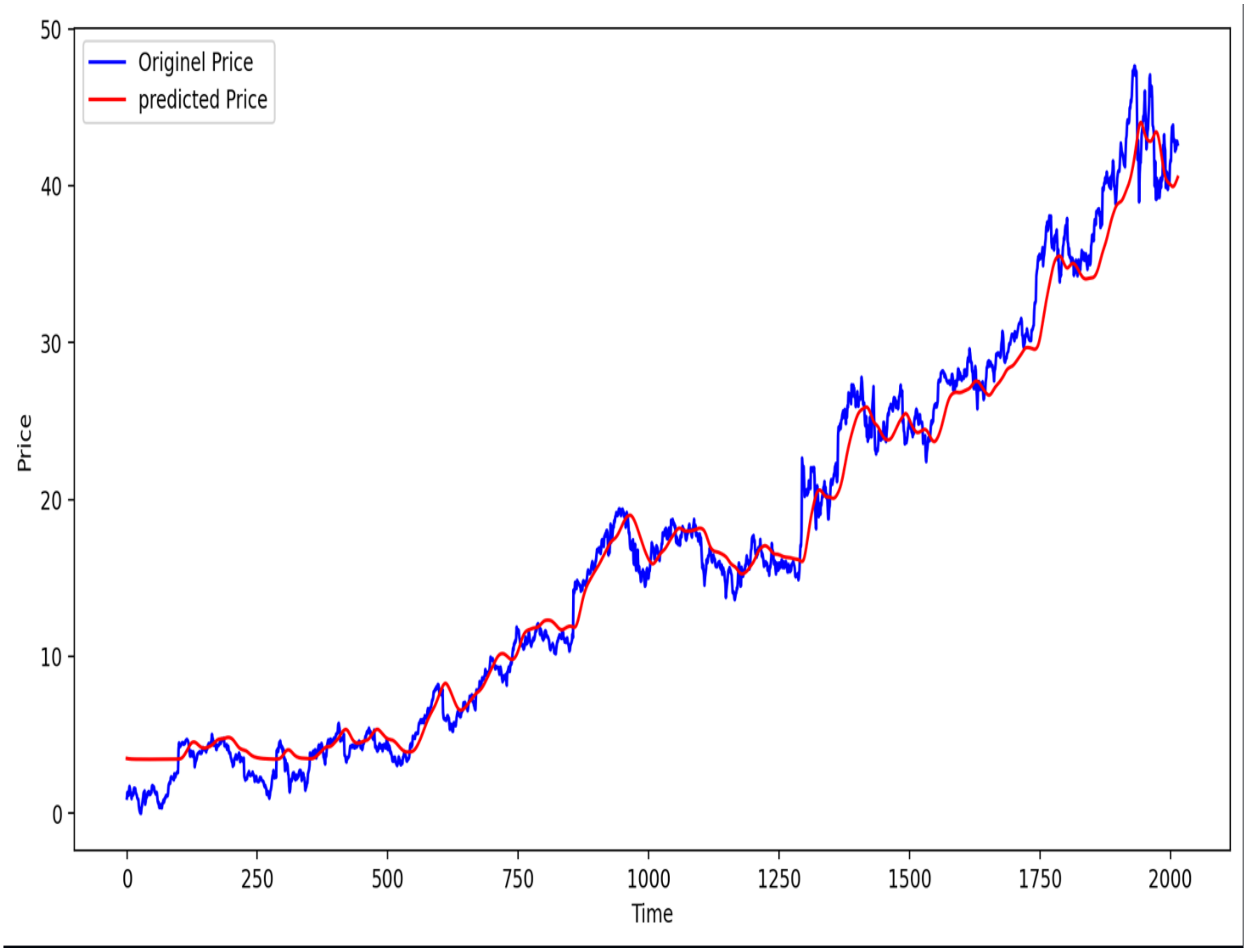


AVERAGECLOSING PRICE VS TIME CHART WITH 100 MOVING AVERAGE



Closing Price vs Time chart with 50MA and 100MA

ORIGINAL PRICE VS PREDICTED PRICE



6.CONCLUSION

In this project, we are predicting closing stock price of any given organization, we developed a web application for predicting close stock price using LSTM algorithm for prediction.Two techniques have been utilized in this paper LSTM and Regression,on the Yahoo finance dataset. Both the techniques have shown an improvement in the accuracy of predictions, there by yielding positive results. Use of recently introduced machine learning techniques in the prediction of stocks have yielded promising results and thereby marked the use of them in profitable exchange schemes.It has led to the conclusion that it is possible to predict stock market with more accuracy and efficiency using machine learning techniques. In the future, the stock market prediction system can be further improved by utilizing a much bigger data set and other models include to get more accuracy in result.

7.REFERENCES

- https://www.youtube.com/watch?v=s3CnE2tqQdo&t=3019s
- https://docs.streamlit.io/library/components/components-api
- https://www.youtube.com/watch?v=bsSBcByYaqQ&t=220s
- https://www.youtube.com/watch?v=hoPvOIJvrb8