

# **Stock Market Simulator & Prediction**

*Project submitted to*

*Shri Ramdeobaba College of Engineering & Management, Nagpur*

*in partial fulfillment of requirement for the award of*

*degree of*

**Bachelor of Engineering**

*In*

**COMPUTER SCIENCE AND ENGINEERING**

**(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)**

*By*

**Ms. Aashi Khanna**

**Ms. Ketaki Tank**

**Mr. Atharva**

**Rewatkar**

**Mr. Vedant Padole**

*Guide*

**Dr. Shailendra S Aote**



**Computer Science and Engineering**

**Shri Ramdeobaba College of Engineering & Management, Nagpur**

**440013**

(An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur  
University Nagpur)

**December 2022**

**SHRI RAMDEOBABA COLLEGE OF ENGINEERING & MANAGEMENT, NAGPUR**

(An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur  
University Nagpur)

Department of Computer Science and Engineering

**CERTIFICATE**

This is to certify that the project on **“Stock Market Simulator & Prediction”** is a bonafide work of

1. Ms. Aashi Khanna
2. Ms. Ketaki Tank
3. Mr. Atharva Rewatkar
4. Mr. Vedant Padole

submitted to the Rashtrasant Tukdoji Maharaj Nagpur University, Nagpur in partial fulfillment of the award of a Degree of Bachelor of Engineering, in Computer Science and Engineering (Artificial Intelligence and Machine Learning). It has been carried out at the Department Computer Science and Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur during the academic year 2022-23.

Date: 17-12-22

Place: Nagpur

Dr. Shailendra S Aote

Project guide

Dr. Avinash Agrawal

H.O. D

Department of Computer

Science and Engineering

## **DECLARATION**

I, hereby declare that the project titled “**Stock Market Simulator & Prediction**” submitted herein, has been carried out in the Department of Computer Science and Engineering of Shri Ramdeobaba College of Engineering & Management, Nagpur. The work is original and has not been submitted earlier as a whole or part for the award of any degree / diploma at this or any other institution / University

Date: 17-12-22

Place: Nagpur

**Ms. Aashi Khanna**  
**(Roll no.: 02)**

**Ms. Ketaki Tank**  
**(Roll no.: 14)**

**Mr. Atharva Rewatkar**  
**(Roll no.: 32)**

**Mr. Vedant Padole**  
**(Roll no.: 66)**

## **ACKNOWLEDGEMENT**

We would like to express our deep and sincere gratitude to our guide **Dr. Shailendra S Aote**, Professor of Computer Science and Engineering Department, RCOEM, for giving us the opportunity to work on this project and providing valuable guidance throughout the project. It was a great privilege and honor to work under her guidance. We are extremely grateful for the experience we had in this project with her.

We express our sincere gratitude to **Dr. Avinash Agrawal**, Head of the Department of Computer Science Department, RCOEM for his guidance. Talent wins games, but teamwork and intelligence win championships. We would like to take this opportunity to express our deep gratitude to all those who extended their support and guided us to complete this project.

**Ms. Aashi Khanna**

**Ms. Ketaki Tank**

**Mr. Atharva Rewatkar**

**Mr. Vedant Padole**

## **ABSTRACT**

In the digital age, financial independence among the masses can be achieved by the act of investing early on in life. The most popular investment choice in today's world are the stock markets. The majority of people turn to the performance of a country's stock market as the best indicator of how well that economy is doing. Stock markets cover all industries across all sectors of the economy. This means they serve as a barometer of what cycle the economy is in and the hopes and fears of the population who generate growth and wealth.

Although the perks of financial investments are widely known by the youth and the general populace, there exists widespread deficiency in the awareness level of people about the intricacies and working of financial instruments like stock markets, budgeting, equity, etc.

Thus, to create higher awareness levels among the student populace and reduce fears about investment risks in the stock market, a simulation of the working of the market is the need of the hour. This project involves the creation of one such platform, StockSim.

StockSim is a real-time stock market simulator web application, which creates a parallel with the working of a real world stock trading application. The purpose of the user-friendly web app is to help the unacquainted population get more comfortable with stock trading, gain an insight about the working of the system and also understand which stocks to buy and when to buy them. To cater to the latter, stockSim has an added feature of stock prediction using Machine Learning models. It also involves various other paramount aspects like user authentication, personal portfolio, real time stock prices and market charts to create a safer, easier, and grasping user experience.

The report/thesis focuses on the research, motivation, development, learning and results of the web application, StockSim.

# TABLE OF CONTENTS

1. Introduction	
1.1 Motivation .....	10
1.2 Problem Statement .....	11
1.3 Objectives .....	12
2. Review of Literature .....	13
3. Fundamentals	
3.1 Front End .....	14
3.2 Back End .....	16
3.3 Machine Learning .....	17
4. Development	
4.1 Components of the website .....	21
4.2 Backend development .....	21
4.3 Prediction .....	22
5. Result	
5.1 Working of application .....	25
5.2 Process of prediction .....	31
6. Summary and Conclusion .....	33
7. Reference Material .....	35

# **1. Introduction**

## **1.1 Motivation**

A stock market is the aggregation of buyers and sellers of stocks (also called shares); these may include securities listed on a stock exchange as well as those only traded privately. It is a reliable barometer to measure the economic condition of a country. It plays an integral role in a financial system. It offers attractive opportunities of investment in various securities. These attractive opportunities encourage people to save more and invest in securities of the corporate sector rather than investing in unproductive assets such as gold, silver etc.

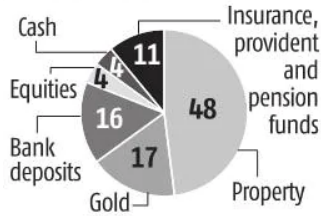
The investors must possess the required awareness about risk–return profile of different securities, concept of how stock market functions, precautions, which should be taken while dealing with market intermediaries and the practices as well as their rights and obligations. A number of researchers have contended stock market awareness to be an important component of financial literacy. More and more financial markets are becoming more sophisticated and new products are continuously offered. Moreover, changes in retirement benefit schemes and increase in life expectancy have increased the thoughts of consumers towards better and efficient savings management (OECD, 2005).

For many young adults, it seems easier to put off any investing decisions until their financial situation becomes, at least theoretically, more stable. Twenty-somethings, however, are actually in a prime position to enter the investing world. Since investing has a fairly lengthy learning curve, young adults are at an advantage because they have years to study the markets and refine their investing strategies.

## NEW INVESTORS WELCOME

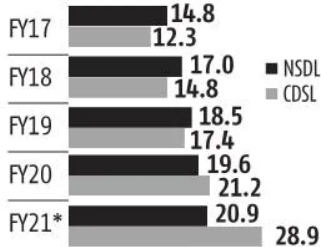
Stocks formed small part of Indian household wealth, but a record number of demat accounts were opened last year

**Indian household assets as of December 2020**



Sources: Reserve Bank of India, Jefferies estimates, Central Depository Services Ltd, National Securities Depository Ltd

**Investors demat accounts (in million)**



\*Dec 2020; Compiled by BS Research Bureau  
Source: CDSL result presentation

Fig. 1 shows millions of new demat accounts were opened post Covid, in the year 2021, indicating higher percentage of stock market beginners in the population

Risks are a part and parcel of investing. When you're new at investing, taking risks feels like walking on eggshells because you're never too sure. But it's the habit and continuous practice which helps the beginners assess their risks and take actions accordingly. But when you're a student, taking risks is far more threatening due to existing student debts and low salaries. Thus, there arises a need for the youth to learn about the market, visualize the risks, and take calculated actions without the fear of losing their meager savings.

### 1.2 - Problem Statement

Having realized the benefits of trading in stocks, how it boosts personal finances and the country's economy, it's vital for the youth of any nation to practice investing from an early age. However, risks involved in the stock market are far more scary for a student than an adult with a consistent income profile. The younger generation is a tech-savvy one, able to study, research, and apply online investing tools and techniques. Moreover, as we saw earlier, they need an online, risk free, stock market platform to provide countless learning opportunities, more than anything. We also need to assess which stock to buy and when to buy it, so that we can earn higher profits and minimize loss.

Hence, in an attempt to address these problems, which most students like us face, we came up with an idea to develop a risk free, stock market simulating web application, StockSim. It simulates the working of a real world stock trading platform, just without the



incorporation of actual money.

This ensures a risk-free, safer learning environment, enabling everyone to learn about the intricacies of stock trading, market trends, reading charts, portfolio maintenance.

### **1.3 Objective**

In order for StockSim to serve its true purpose and pose as a useful tool and significant contribution to the advancement of financial literacy, it needs to have the following features -

#### **1) User friendliness**

To ensure smooth learning and trading experience, the web application has to be easy to use and understand by the general user. It increases engagement and general appeal.

#### **2) User authentication**

The trading practices of any user can be hazardous to them if exposed to fraudulent public entities. Hence arises the need for a secure web browsing experience, which can be achieved by user authentication. This involves the user signing into the application using their personal email addresses.

#### **3) Personal stock portfolio**

For any trader, keeping track of their investments is vital for financial security and growth. StockSim proves to achieve this by displaying the stock portfolios of every user depending on the stocks they've bought and sold.

#### **4) Real time stock prices**

Stock prices are dynamic and observe an incline or decline every second. Thus the app needs to display the accurate and latest stock price of every stock the user searches for.

## 5) Accurate stock prediction

To aid the beginners in their learning and trading journey, StockSim has an added feature of stock prediction which allows the user an insight into which stock they can buy depending on their portfolio and the current market trends.

## 2. Review of Literature

In this mini project we are trying to implement a machine learning approach to stock price prediction. Machine learning is effectively implemented in stock price forecasting. The goal is to predict stock prices so that more informed and accurate investment decisions can be made. We propose a stock price prediction system that integrates mathematical functions, machine learning and other external factors to achieve better stock prediction accuracy and issue profitable trades.

There are two types of shares. You may know intraday trading by the commonly used term "day trading". Intraday traders hold positions in securities for at least one day and often several days to weeks or months. Now why LSTM's well LSTMs are very powerful in sequence prediction problems because they are able to store past information. This is important in our case because the previous price of a stock is crucial for predicting its future price. While predicting the actual price of a stock is an uphill climb, we can build a model to predict whether the price will go up or down.

Research work done by V Kranthi Sai Reddy Student, ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India. In the financial world, stock trading is one of the most important activities. Stock market prediction is the act of trying to determine the future value of a stock or other financial instrument traded on a financial exchange. This article explains stock prediction using machine learning. Technical and fundamental analysis, or time series analysis, is used by most stock traders when making stock predictions. The programming language used to predict the stock market using machine learning is Python. In this paper, we propose a

machine learning (ML) approach that will be trained from available stock data and gain intelligence and then use the acquired knowledge to make accurate predictions. In this context, this study uses a machine learning technique called Support Vector Machine (SVM) to predict stock prices for large and small caps and in three different markets, using prices with both daily and current frequencies.

Also recently the research work was done by a group of students of Anil Neerukonda Institute of Technology and Sciences. They used the LSTM model with Random forest and LMS filter to get their accuracy to 97 percent after 50 epochs. This article also made clear how well LSTM goes with time series forecasting. LSTM is a special network structure with three "gate" structures. Three gates are located in the LSTM unit, called the input gate, the forgotten gate, and the output gate. As information enters the LSTM network, it can be selected according to rules. Only information that conforms to the algorithm will remain, and information that does not conform will be forgotten by the gate of oblivion

### **3. Fundamentals & Technology Stack**

#### **3.1 - Frontend Technologies**

##### **1) HTML**

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as CSS and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of web pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and

other items.

## 2) CSS

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML. CSS is designed to enable the separation of content and presentation, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

## 3) JavaScript

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat.

## 4) Bootstrap

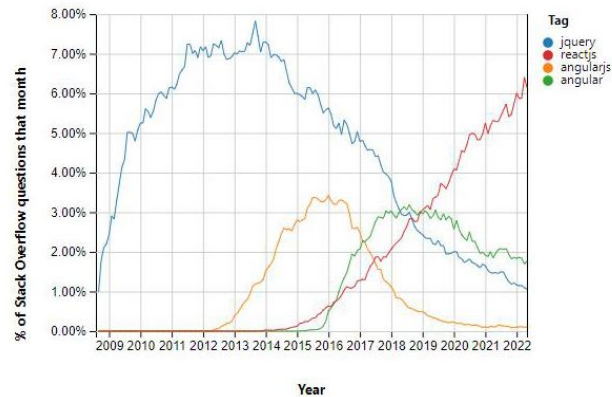
Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

## 4) ReactJS

React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It can be used as a base in the development of single-page, mobile, or server-rendered applications with frameworks like Next.js. However, React is only concerned with state management and rendering that state to the DOM, so creating React

applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

React lays the foundation for stockSim. It enables us to create reusable components that can be used on various pages once rendered. As a result, development effort is reduced while flawless performance is maintained.



Graph 1 shows increasing demand and usage of React over the years, owing to its many benefits.

### 3.2 - Backend Technologies

#### 1) Firebase

Firebase is Backend as a Service (BaaS) used to authenticate users for the project, firebase authentication offers ready-made UI libraries, simple SDKs, and backend services. It supports federated identity providers like Google, Facebook, and Twitter, as well as passwords, phone numbers, and other methods.

In the project, we have authenticated using email ID and password.

We have utilized Firebase's secure service for authentication. This token-based authentication method makes it simple to integrate with most platforms. It makes use of industry standards like OpenID Connect and OAuth 2.0, making it simple to integrate with your unique backend.

We used firestore, a versatile, scalable database for mobile, web, and server development from Firebase and Google Cloud, to securely store the user's portfolio. It supports offline support for mobile and web, maintains your data synchronized among client apps with real-time listeners, and enables you to create responsive apps that function independent of network latency or Internet connectivity. Additionally, Cloud Firestore provides easy connection with Google and Firebase.

### 3.3 - Machine Learning Technologies

#### 1) Python and libraries

The Python libraries which we will be using are as follows

Python library	Description
Pandas	Pandas is an open source library for Python. It provides powerful out-of-the-box data structures and data analysis tools. Pandas modules run on NumPy and are widely used for data science and data analysis
Numpy	NumPy is a Python library used for manipulating arrays. It also has functions for working in the areas of linear algebra, Fourier transforms, and matrices. NumPy was developed by Travis Oliphant in 2005. This is an open source project and is free to use.

Matplotlib	Matplotlib is a Python library used to create 2D graphs and plots using Python scripts. There is a module called pyplot that makes plotting easier by providing functions to control line styles, font properties, axis formatting, etc.
Tensorflow	TensorFlow is a Python library for fast numerical computations created and published by Google. It is a core library that can be used to build deep learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow.
Keras	Keras is a high-level deep learning API developed by Google for implementing neural networks. It is written in Python and is used to facilitate the implementation of neural networks. It also supports the calculation of multiple backend neural networks.
yfinance	yfinance is one of the famous modules in Python which is used to collect online data and with it we can collect yahoo financial data. With the help of the yfinance module, we obtain and collect

	company financial information (such as financial indicators, etc.)
Streamlit	Streamlit is an open source application framework in Python. It helps us build data science and machine learning web applications in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy (latex), NumPy, pandas, Matplotlib, etc.

## 2) ML Models

To train our model we will use RNN. A recurrent neural network (RNN) is a type of artificial neural network that uses sequential data or time series data. Recurrent neural networks use training data to learn. They stand out with their "memory" because they take information from previous inputs to influence the current input and exit. While traditional deep neural networks assume that inputs and outputs are independent, the output of recurrent neural networks is mutually dependent on the previous elements in the frame sequence. Recurrent neural networks use the backpropagation in time (BPTT) algorithm.

determine gradients that differ slightly from traditional backpropagation specific to sequence data. The principles of BPTT are the same as traditional backpropagation, where the model is trained by calculating errors from its output layer to its input layer. RNNs tend to run into two problems, known as exploding gradients and vanishing gradients.

Vanishing gradient:

Vanishing gradient problem where neural networks are unable to propagate back to



a better gradient.

For this purpose we have LSTM to help us overcome this problem. Simple LSTM helps to keep the magnitude of the gradient constant. The activation function we use in LSTM often works like the identity function which is the derivative of 1. So in gradient backpropagation the magnitude of the gradient

it won't disappear. This is a popular RNN architecture introduced by Sepp Hochreiter and Juergen Schmidhuber as a solution to the vanishing gradient problem. This means that if the previous condition that affects the current prediction is not in the recent past, the RNN model may not be able to accurately predict the current state. Let us give an example where we said we wanted to predict the italicized words in the following text: "Alice is allergic to nuts. She can't eat peanut butter." The context of a nut allergy can help us predict that a food that cannot be eaten contains nuts. However, if that context was a few sentences earlier, then yes would make RNN difficult or even impossible to link information. On

LSTMs have "cells" in the hidden layers of the neural network, which have three gates – entry gate, exit gate and forgotten gate. These gates control the flow information that is needed to predict network output. For example, if gender pronouns such as "she" have been repeated several times in previous sentences, you can exclude them from the cellular state.

### 3) Datasets

To get the dataset and the intraday data we will be using the python library yfinance or Yahoo finance. To webscrape our data we will use multiple open source libraries available in Python. To use our data more effectively we will also use pandas data reader library function to do some manipulations to train and test our data more accurately.

## **4. Development**

### **4.1 - Components of the website**

Various components were made, in order to bring our web application together, to name them:

1. Landing page
2. About us page
3. Features page
4. Login / Signup page
5. Dashboard
6. Portfolio
7. Contact us page
8. Navigation bar (common in all pages)

Components promise modular programming, which helps us develop complex web applications by writing less, but powerful code. Navigation bar and footer was common for all pages, but in order to change/customize links for certain pages, conditional coding was implemented and navigation bar was made dynamically, based on the state of the web application

### **4.2- Backend development stages**

Firebase (BaaS) proved to be extremely useful in implementing the backend of the web application. It made the web application faster in responding and easier to store integral data about the users and their respective transactions made per second. Setting up a firestore in the react app, using API keys was easily done using the detailed documentation provided by Firebase. Unique user IDs were generated every time a new user registered in the application, and these IDs were referred to when any user tried to log into their dashboard.

For each transaction they made, a unique document ID was generated, which could easily be referred to by querying the document ID, when update was needed.

### 4.3 - Prediction

Python code for training and testing of the data

Importing the libraries:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import pandas_datareader as data
import plotly.graph_objs as go
import yfinance as yf
start='1980-01-01'
end='2022-11-21'
```

Getting our training data:

```
df=data.DataReader('TSLA','yahoo',start,end)
df=df.reset_index()
df=df.drop(['Date','Adj Close'],axis=1)
```

Training the data on closing price

```
train=pd.DataFrame(df['Close'])
```

Getting our testing data

```
train=pd.DataFrame(df['Close'])
print(train.shape)
test = yf.download(tickers='TSLA', period='5d', interval='5m')
test=test.reset_index()
```

```
test=test.drop(['Datetime','Adj Close'],axis=1)
test=pd.DataFrame(test['Close'])
```

## Pre-processing the data

```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
train1=scaler.fit_transform(train)
train1.shape
```

## Splitting the data into x\_train,y\_train

```
x_train=[]
y_train=[]

for i in range(100,train1.shape[0]):
    x_train.append(train1[i-100:i])
    y_train.append(train1[i,0])

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

## Building the LSTM model

```
from keras.layers import Dense,Dropout,LSTM
from keras.models import Sequential

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=70,activation='relu',return_sequences=True))
model.add(Dropout(0.4))

model.add(LSTM(units=90,activation='relu'))
model.add(Dropout(0.5))
```

```
model.add(Dense(units=1))  
model.summary()
```

```
model.compile(optimizer='adam',loss='mean_squared_error')  
model.fit(x_train,y_train,epochs=50)
```

```
prev100=train  
final_df=prev100.append(test,ignore_index=True)  
input_data=scaler.fit_transform(final_df)  
input_data.shape
```

## Splitting our data for testing

```
x_test=[]  
y_test=[]  
  
for i in range(100,input_data.shape[0]):  
    x_test.append(input_data[i-100:i])  
    y_test.append(input_data[i,0])  
x_test,y_test=np.array(x_test),np.array(y_test)  
print(x_test.shape)  
print(y_test.shape)  
y_pred=model.predict(x_test)
```

## Predicting the output

```
y_pred=model.predict(x_test)
```

## Inverse Scaling

```
scale_f=1/float(scaler.scale_)  
y_pred=y_pred*scale_f  
y_test=y_pred*scale_f
```

## Plotting the figure and curves

```
plt.figure(figsize=(12,6))
plt.plot(y_test,'green',label='Original')
plt.plot(y_pred,'purple',label='Prediction')
plt.xlabel('Time')
plt.ylabel('Price')
plt.legend()
plt.show()
```

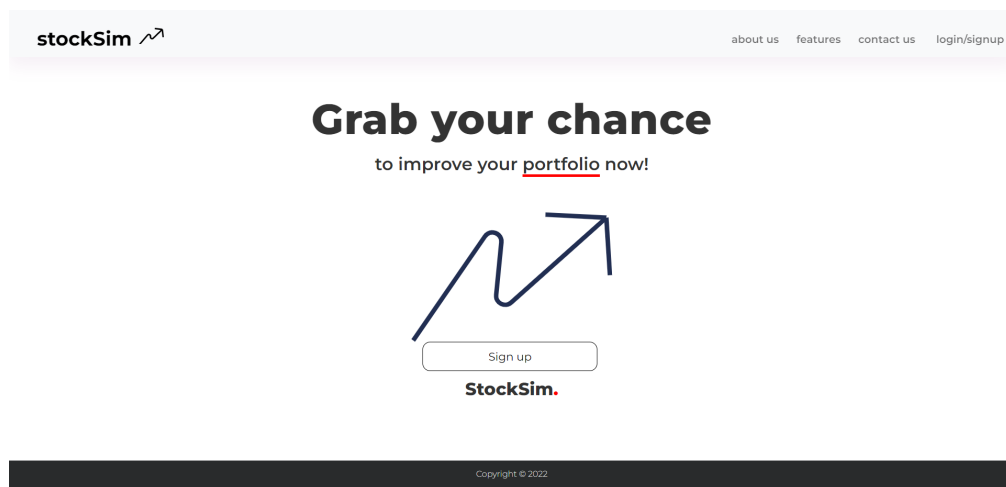
Saving our model

```
model.save('keras_model.h5')
```

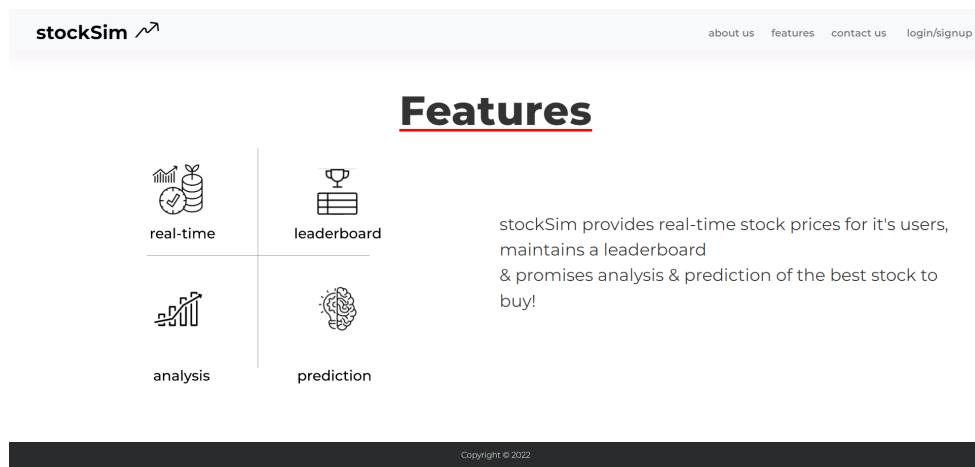
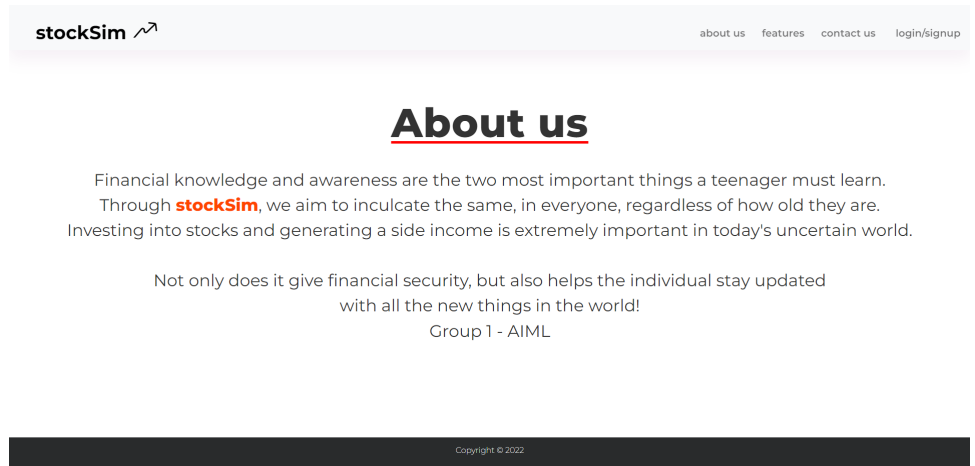
## 5. Result

### 5.1 Working of application

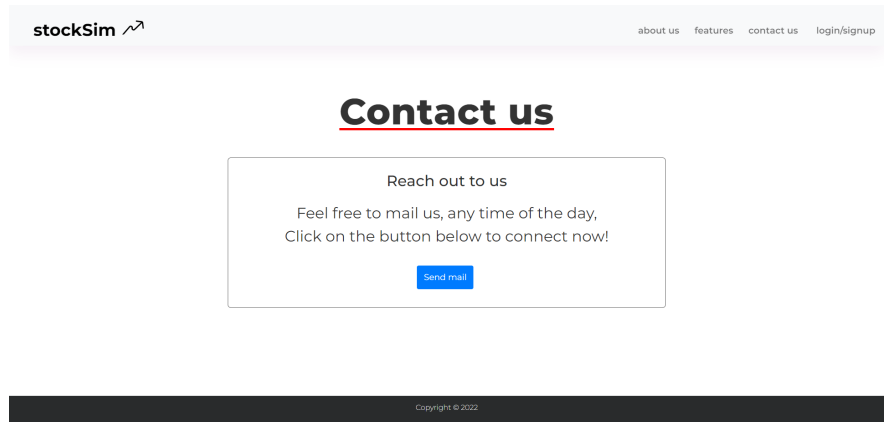
#### 1. Landing Page



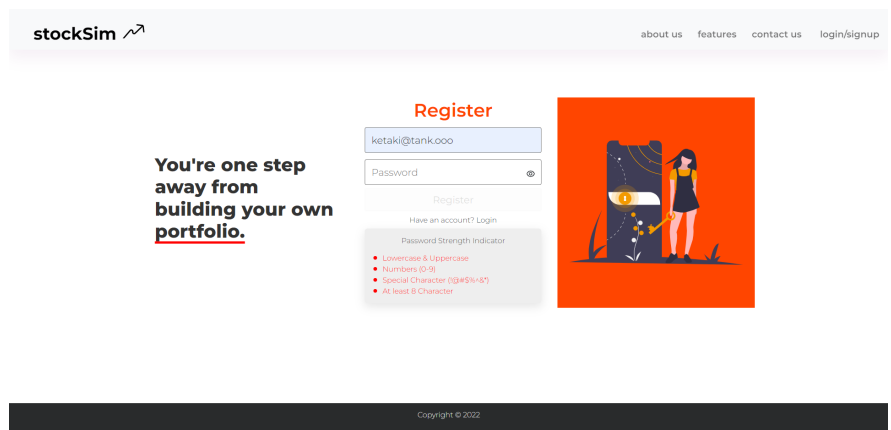
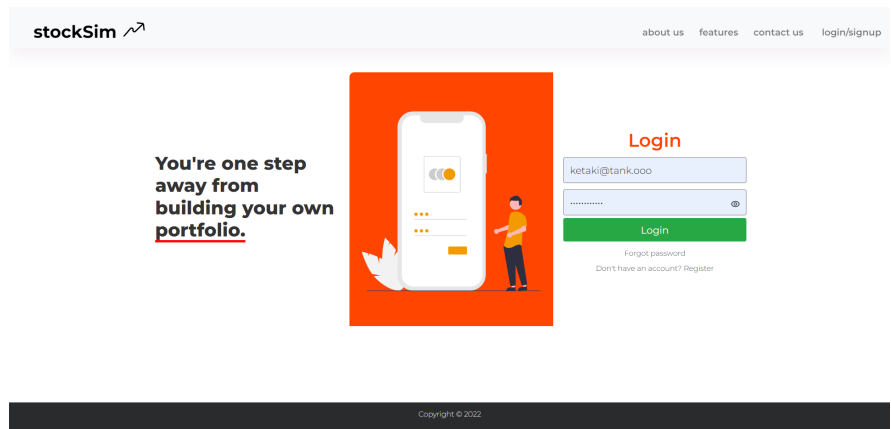
## 2. About us & Features page



### 3. Contact us page

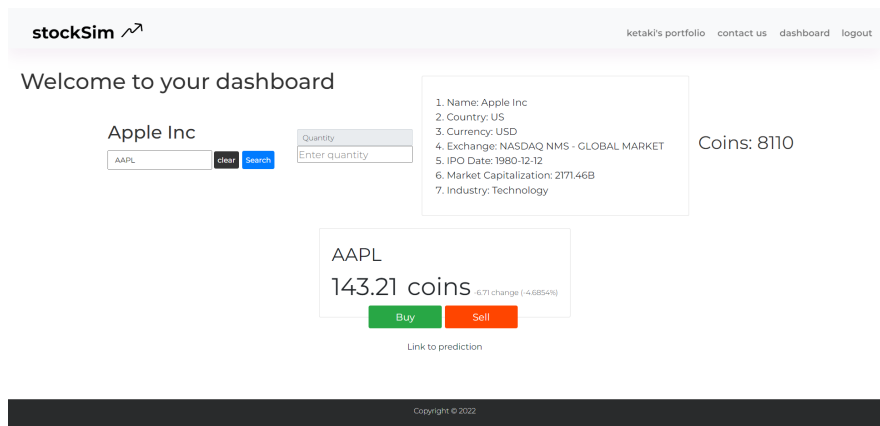


### 4. Login/Sign up page

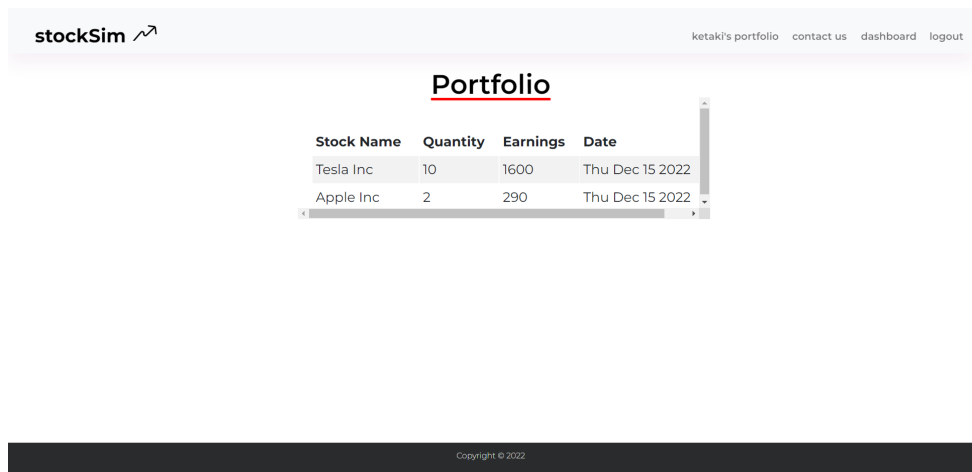




## 5. Dashboard



## 6. Portfolio page



## 7. Buying outcome

stockSim

localhost:3000 says  
Bought: 3

ketaki's portfolio   contact us   dashboard   logout

Welcome to your dashboard

Starbucks Corp

SBUX

clear

Search

Quantity  
5

1. Name: Starbucks Corp  
2. Country: US  
3. Currency: USD  
4. Exchange: NASDAQ NMS - GLOBAL MARKET  
5. IPO Date: 1992-06-26  
6. Market Capitalization: 114.77B  
7. Industry: Hotels, Restaurants & Leisure

Coins: 8110

SBUX

102.19 coins

2.2 (change: 2.1529%)

Buy

Sell

Link to prediction

Copyright © 2022

stockSim

ketaki's portfolio   contact us   dashboard   logout

Portfolio

Stock Name	Quantity	Earnings	Date
Tesla Inc	10	1600	Thu Dec 15 2022
Apple Inc	2	290	Thu Dec 15 2022
Starbucks Corp	5	510	Fri Dec 16 2022

Copyright © 2022

8. Selling outcome

stockSim ↗

localhost:3000 says  
Sold: 3

ketaki's portfolio   contact us   dashboard   logout

Welcome to your dashboard

Starbucks Corp

Enter stock symbol   [Search](#)

Quantity  
3

1. Name: Starbucks Corp  
2. Country: US  
3. Currency: USD  
4. Exchange: NASDAQ NMS - GLOBAL MARKET  
5. IPO Date: 1992-06-26  
6. Market Capitalization: 114.77B  
7. Industry: Hotels, Restaurants & Leisure

Coins: 7600

SBUX

102.19 coins   -2.2 change (-2.1529%)

[Buy](#)   [Sell](#)

[Link to prediction](#)

Copyright © 2022

stockSim ↗

ketaki's portfolio   contact us   dashboard   logout

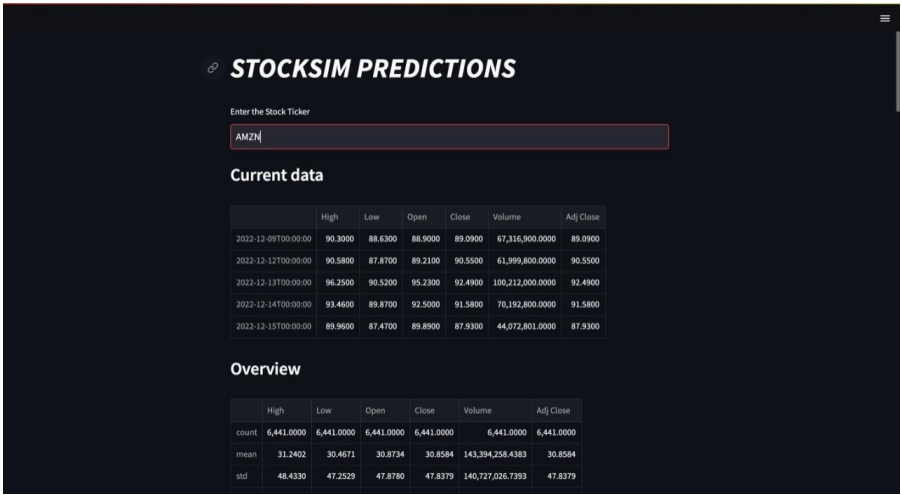
Portfolio

Stock Name	Quantity	Earnings	Date
Tesla Inc	10	1600	Thu Dec 15 2022
Apple Inc	2	290	Thu Dec 15 2022
Starbucks Corp	2	204	Fri Dec 16 2022

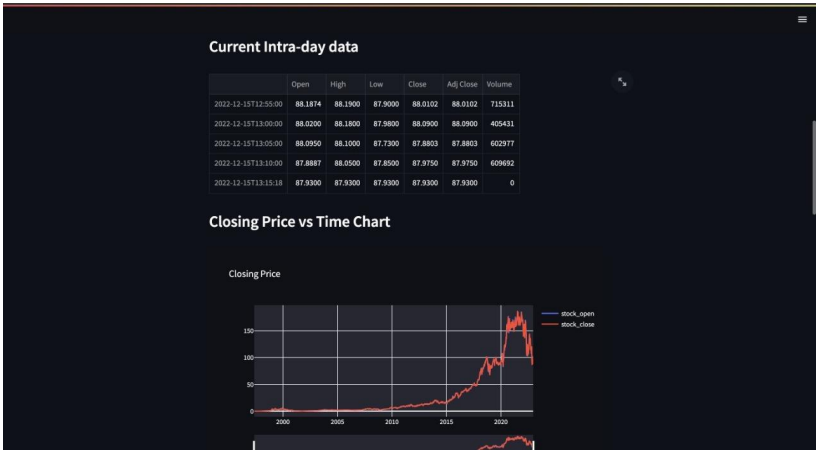
Copyright © 2022

## 5.2 Process of Prediction

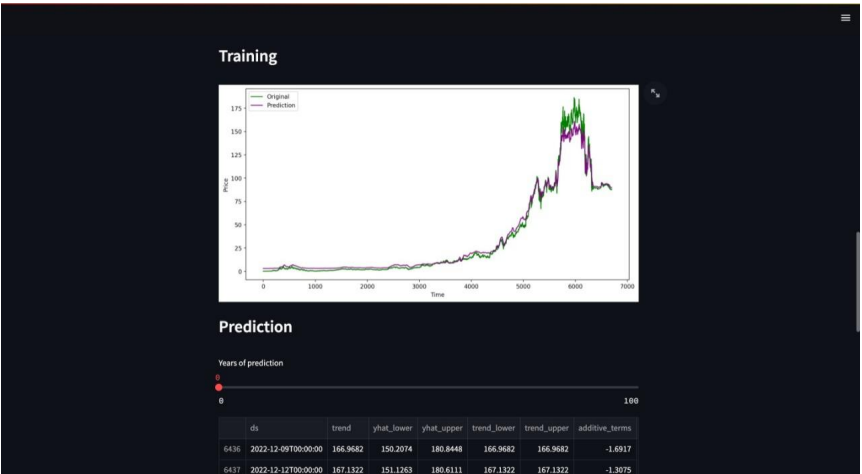
Prediction Page:



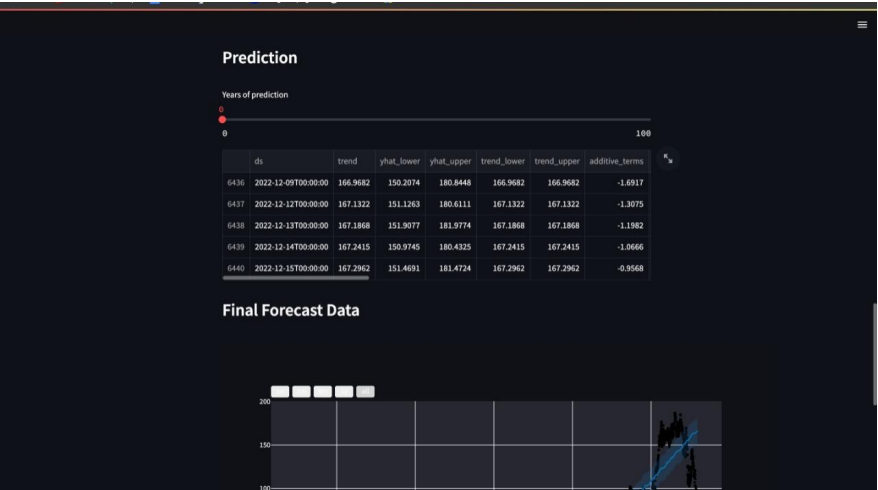
Current intraday data:



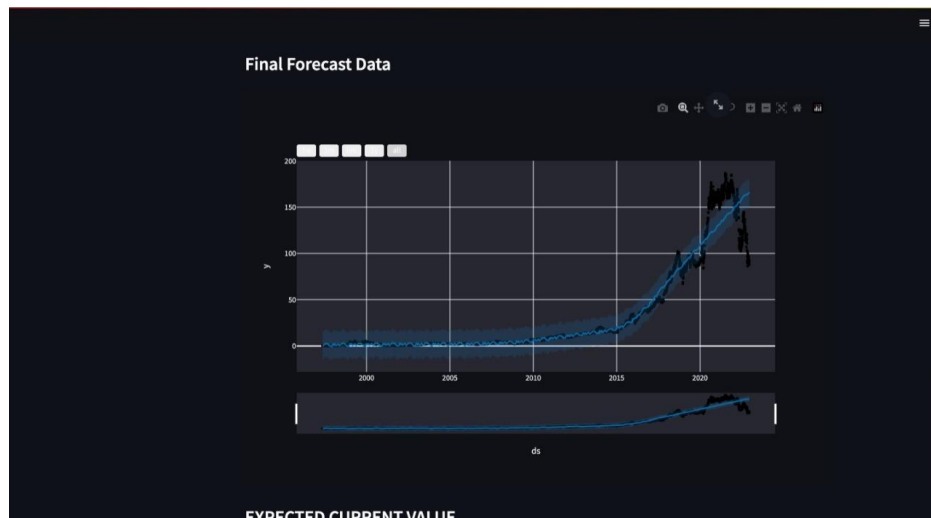
Training vs Prediction curves:



Final prediction:



Final predicted curve:



Expected current intraday value:



## 6. Summary and Conclusions

This research aimed to design a stock market simulation and prediction application (StockSim) based on real-time data analysis using Machine Learning techniques. It was aimed to help the beginners gain an insight into the working of the market and stock trading, by simulating the working of a real world trading platform and

helping the user gain an edge in trading through predictions.

To achieve our objectives, we created a React based web application with multiple pages such as the landing page, a dashboard, a personal portfolio, login/signup among others. StockSim application also makes it easy to build and design an individual portfolio whose security is ensured to the user using the Firebase authentication. The other contribution of this application is that it works on a real-time database using Firestore, a scalable NoSQL-JSON database which helps the users to utilize and analyze the real-time stock prices and help them to buy or sell the stocks accordingly. The central question for this research was utilizing real time stock prices and foreseeing the most ideal stock to purchase based on the concept of Machine Learning and Deep Learning. For instance, if we buy a stock of Apple INC, not only can we see the value of the amount of stock and its price but also predict the value of the stock for the next one hundred years. This research has shown that the accuracy of the prediction algorithm is 85.97% , so the users can determine and consider the selling or buying of the stocks accordingly by analyzing the synclines and anticlines seen in the prediction of the stock prices based on the intra-day trading.

After the conduction of the seminar, we were made aware about the points to improve our Mini-Project. A list of tickers (Names of the Stocks enlisted in the Stock Markets) was not shown. The addition of this list can improve the User Interface of our Mini-Project by making it more user friendly. The accuracy of our Prediction model was 85.97%. It can be improved by using more epochs and by using an LMS filter. There is also a possibility where we can make use of LSTM and Random Forest which will improve the accuracy of our mini-project manifold.

Hence, during the course of the project, we got hands-on experience in the development of a full stack web application along with incorporation of machine learning and deep learning models. We realized how the application serves its purpose in creating higher awareness about investing, leading to ease of trading

practices with substantial growth and learning.

## **7. Reference Material**

[1] Somaraju Dinesh, Adduri Maruthi Siva Rama Raju, Sasumana Rahul, Oruganti Naga Sandeep- DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES  
- STOCK PRICE PREDICTION

[2] Kranthi Sai Reddy Vanukuru - Sreenidhi Institute of Science & Technology - Stock Market Prediction Using Machine Learning - 2018

[3] “API Documentation | Finnhub - Free APIs for realtime stock, forex ..” Finnhub-API.  
<https://finnhub.io/docs/api>.

[4] Gomathy, C K. (2021). THE STOCK MARKET PREDICTION SYSTEM.

[5] Pramod, & Pm, Mallikarjuna. (2021). Stock Price Prediction Using LSTM. Test Engineering and Management. 83. 5246-5251.