Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Information Technology

B.Tech. IT Semester – VI

Subject: SDP

Project Title:

Stock Market Look-up And Prediction System

NAME : KANJIYA YASH
ROLL NO : IT054

NAME : KARANGIYA KULDIP
ROLL NO : IT055

DHARMSINH DESAI UNIVERSITY, NADIAD

FACULTY OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "Stock Market Look-Up And Prediction System" is a bonafide report of the work carried out by Kuldip Karangiya, Student ID 19ITUBS068 and Yash Kanjiya ,Student ID 19ITUOS057 of Department of Information Technology, semester VI, under the guidance and supervision for the subject Software Development Project. They were involved in Project training during the academic year 2021- 2022.

Prof. Anand D Dave,

Information Technology,

Dharmsinh Desai University,

Nadiad.

Prof. Vipul Dabhi

Head of department.

Table of Content

Abstract

Stock Market systems contain real time live prices of the stocks and indexes.
Using our system users can make an account and also get the historical prices and charts. so that users can make decisions when to buy and how much.
Users can also make triggers on the stocks.
User can also make watch-lists as he wants.
User can see the information related to IPO.
Our System provides reliable price charts and also information about the company.
Our system provides information about basic fundamental data about the company.
Users can also see the market related prediction.
Users can manage portfolio and get ROI (Return of Investment) Value .

Introduction

Stock Market look-up system has 2 users which is End User and Admin.

In our system, we can provide a search option which is used for searching companies and add them to the watchlist. Users also can look into the chart and take decisions on them. Users also can get OHLC (Open,High,Low,Close) price of that day.User can Produce Trigger so that when price reaches to that level then system notified the user. Users can also view the company details (like current price,symbol,sectors,description,etc..).Users can see the stocks related prediction. Users can manage the portfolio and ROI(Return of Investment) value.

Purpose :

This document consists of all the functional and nonfunctional requirements of the Stock Market Look-up system and all the other requirement related details. This SRS document will be used as a reference and a guidance for the designer and development process ahead.

Scope :

This System can be useful to everyone who wants to see the real-time price of the US market stock and for he/she needs to register through our portal. This system is designed to help traders and investors to do research on the market at one platform. Scope of the application is global and open for all users. End Users have to provide his details (name, email, phone_no.) and other information which is used to notify users about triggering so that they can use this platform. Users can see the market related prediction and manage their portfolio.

Technologies :

- HTML 5, CSS3
- Material UI
- React JS
- Express JS
- Node.JS
- Highcharts
- MongoDB
- Alpaca API for getting market data
- Polygon.io for getting fundamental data
- Stremlite python App Framework
- Sqlite3 database
- Keras python library
- Numpy and pandas

Tools :

- Visual Studio Code with Extension for typescripting(ESLint)
- Postman for testing Web Request
- MongoDB compass (GUI of MongoDB)

Software Requirement Specification

Types of User

- End User
- Admin

Functional Requirements

R.1 : Manage User

R.1.1 : Sign Up

Primary Actor : User
Precondition : Internet connection available

INPUT : Name, Email, Password, Phone number
OUTPUT : Registration Successful

Description : User can make account through sign up.

R.1.2 : Login

Primary Actor : User
Precondition : User must be registered

INPUT : Email and Password
OUTPUT : Login successful and redirected to Homepage

Description : The user can login into the system.

R.1.3 : Forgot Password

INPUT : Click the forgot password link
OUTPUT : The user can add new password and confirm password

Description : If user forget password then they can reset password through OTP.

R.1.4 : Logout

INPUT : Click on Logout
OUTPUT : Successfully Logged out

R.2 : Manage Stocks

R.2.1 : Add the Stocks on Watch-List

INPUT : Company Symbol
OUTPUT : Stocks price and name shown to the dashboard

Description : User can add the company to watchlist so that they can see all data.

R.2.2 : Delete the Stocks on Watch-List

INPUT : Click on Stock in Watchlist
OUTPUT : Stock successfully deleted

Description : User can add stock to watchlist and Delete the stock on watchlist.

R.2.3 : Watch the stock prize

Precondition : NIL

INPUT : Search any Stock or Index
OUTPUT : Display the Stock price

Description : The user can search any company stock.

R.2.4 : Watch all information about stocks

Precondition : NIL

INPUT : Company Symbol
OUTPUT : Name,Current Price,(OHLC),Sector Description

Description : The user can see all the information about stock.

R.2.5 : View Chart (Historical and Live)

Precondition : NIL

INPUT : Click on selected stock in watchlist
OUTPUT : View chart of selected stock

Description : The user can see the historical chart of any stock.

R.2.6 : Computation

Precondition : NIL

INPUT : User add the invested amount and year
OUTPUT : Networth and ROI

Description : User can see the profit.

R.3 : Manage Triggering

Precondition : NIL

R.3.1 : Put the Trigger

INPUT : Trigger price of the particular stock
OUTPUT : Success/Failure message

R.3.2 : Update the Trigger

INPUT : Updated Trigger Price
OUTPUT : Success/Failure message

R.3.3 : Delete the Trigger

INPUT : Triggered Stock
OUTPUT : Success/Failure message

Description : The user can set alert for any stock price and update and delete the trigger. If the stock price will cross the limit then user can get instant e-mail Related to stock.

R.4 : Manage Portfolio and Market Prediction

Precondition : NIL

R.4.1 : Information about IPO

INPUT : Click on IPO tab
OUTPUT : Display all information related IPO

Description : User can see the all detail of IPO.

R.4.2 : Manage Portfolio

Input : Add the stock with quantity and date
Output : Display the net profit of invested amount.

Description : Display the total invested amount and give the net profit value after given time range.

.

R.4.3 : Market Prediction

Precondition : NIL

INPUT : Select the Index
OUTPUT : Display the Bullish or Bearish value of price.

Description : Display the market Trade predictive chart.

R.5 : Manage Admin

R.5.1 : Login

INPUT : Email and Password
OUTPUT : Login Successful

R.5.2 : View User

INPUT : Username
OUTPUT : Details of End User

Description : The Admin can see the details of user through Username.

Non Functional Requirements

Performance : It is very important to know that system performs certain functions under specific conditions. Examples like speed of response, throughput, execution time and storage capacity. This system should be designed in such a way that its performance is smooth for users.

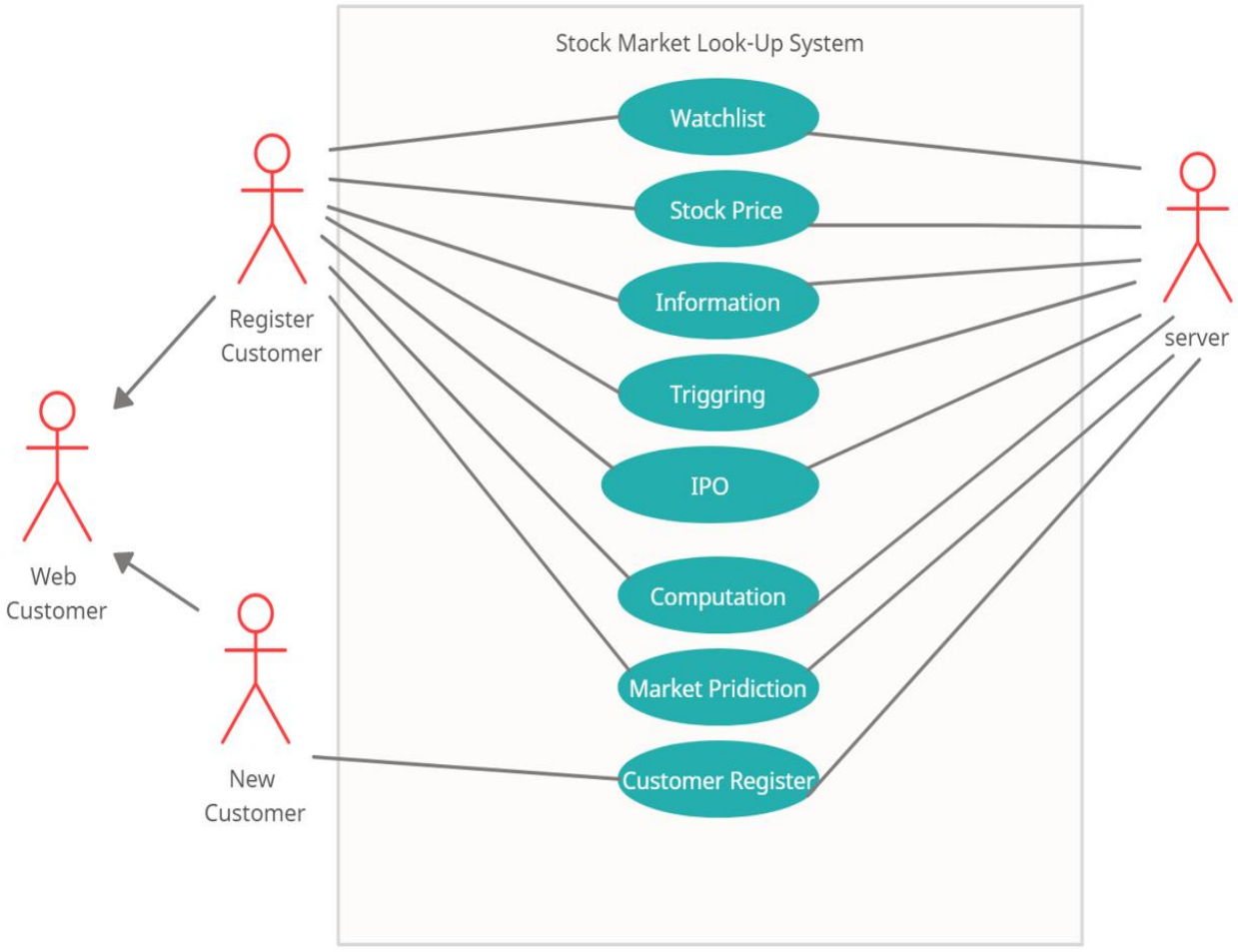
Security : It is a primary requirement of any system. It must maintain data in a secure way so that unauthorized entities do not have illegal access of the system.

Availability : System must be available to all authorized users when it is required at any time. In any abnormal situation like loss of internet connection user's data should not be lost.

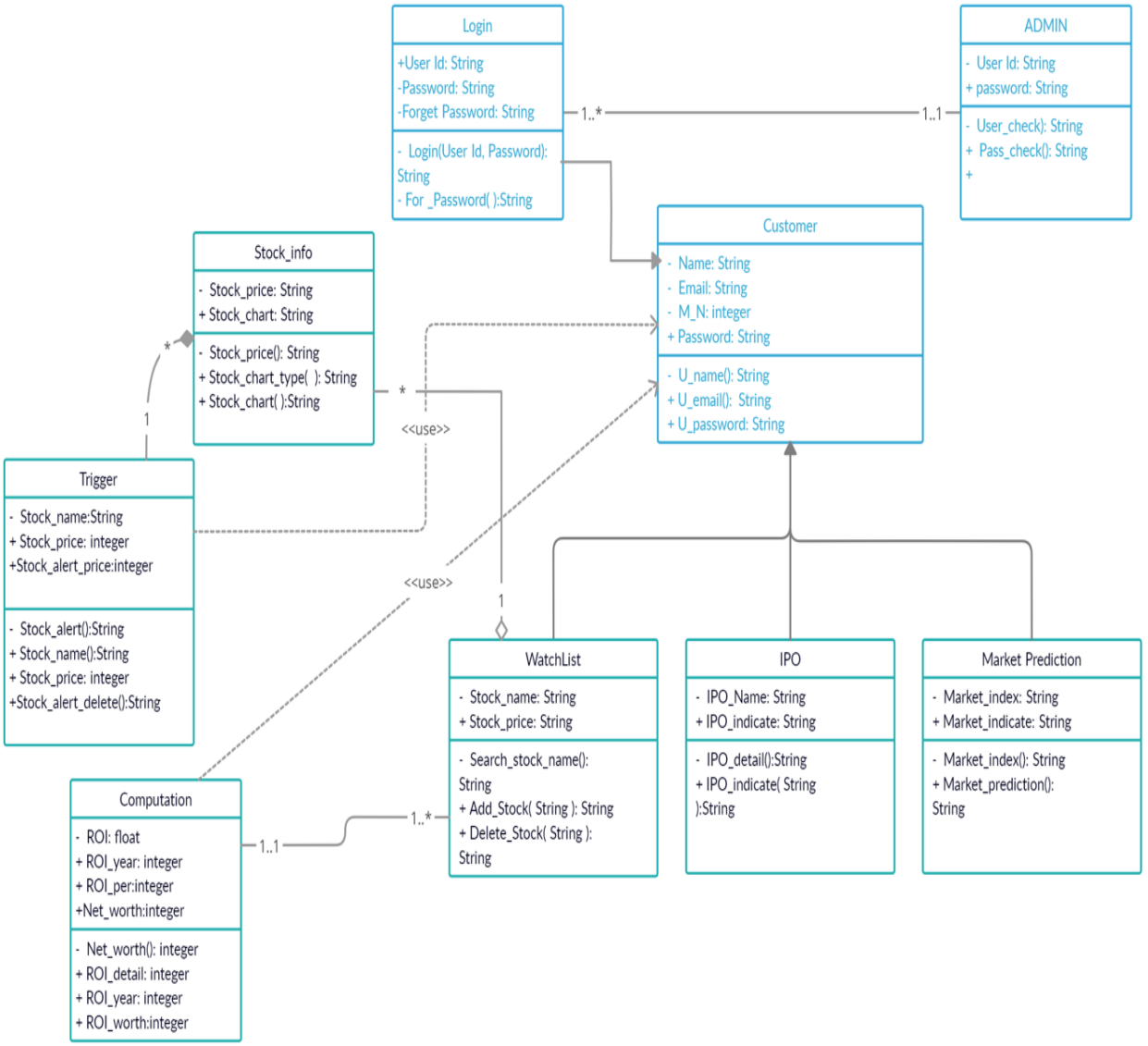
Scalability : It is a property of a system that describes the ability to handle increasing and decreasing workload. It means the ability to grow or shrink a piece of software to meet changing demands on a business. Software scalability is critical to support growth, but also to pivot during times of uncertainty and scale back operations as needed.

DESIGN :

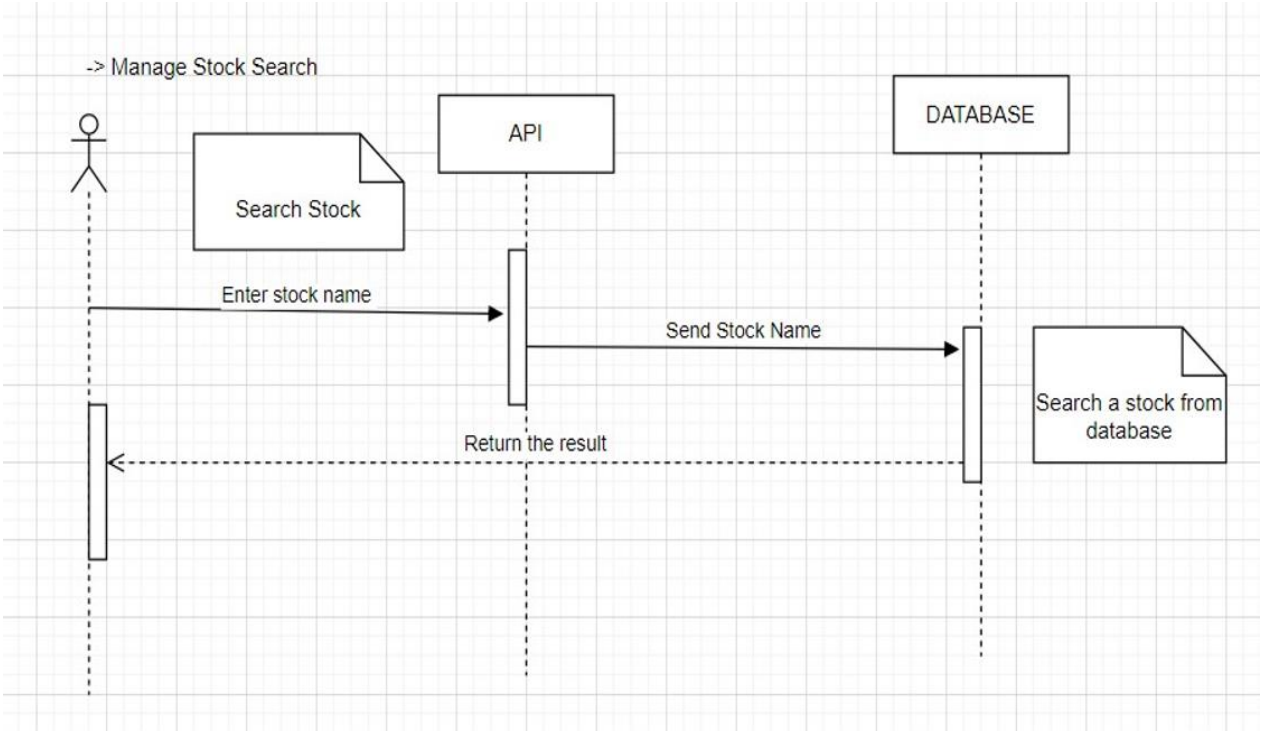
Use Case Diagram :

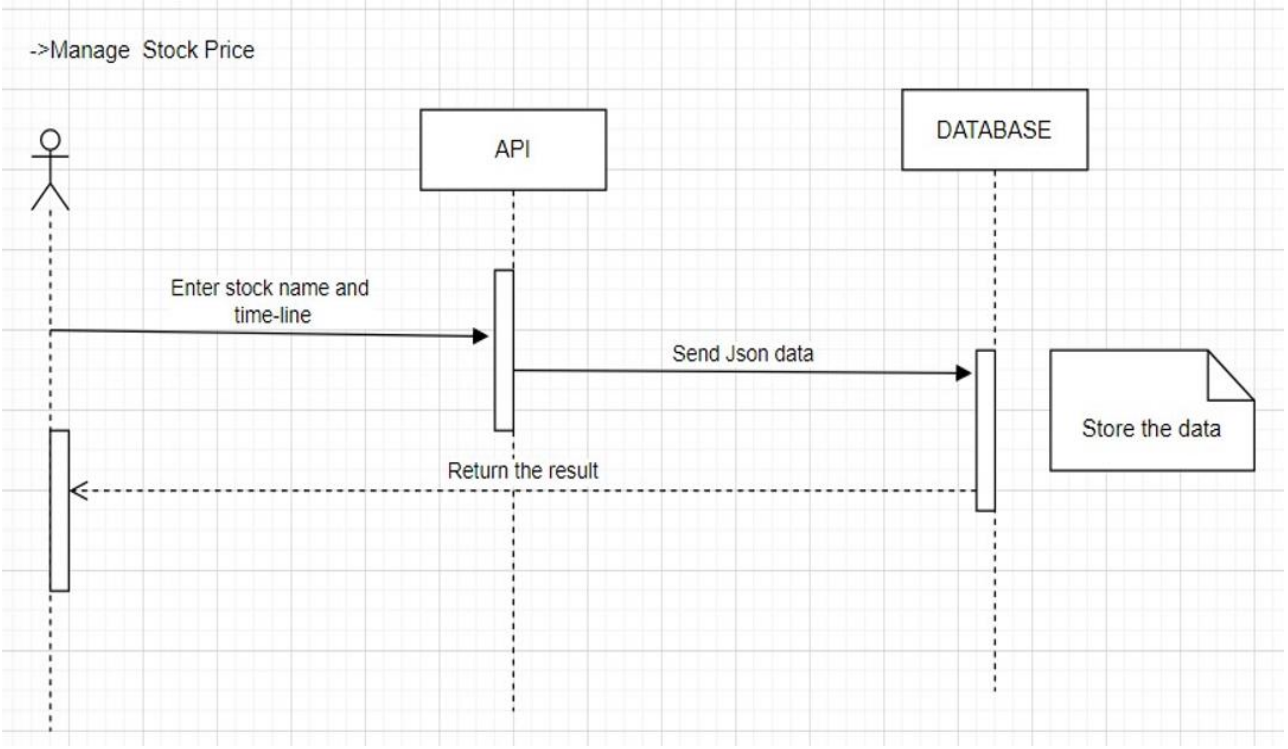


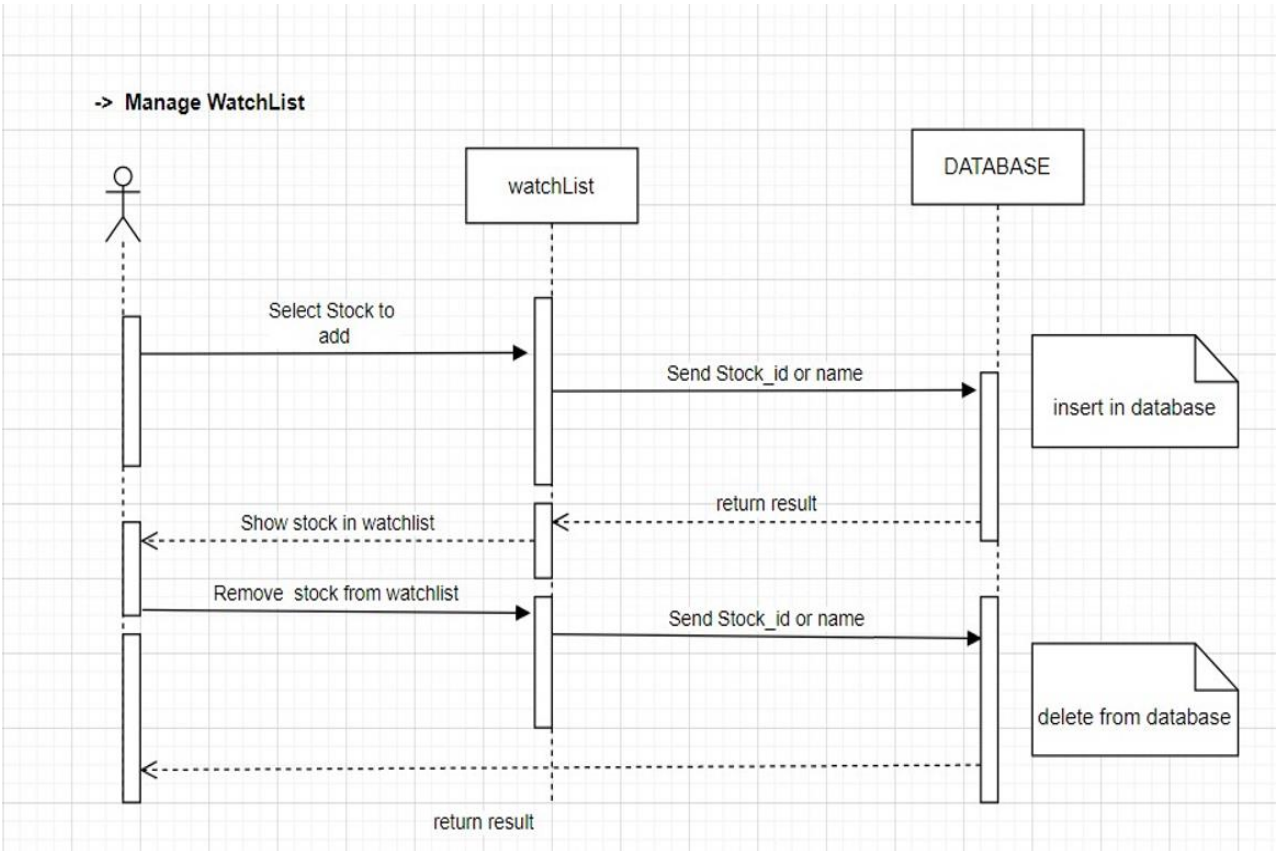
Class Diagram :

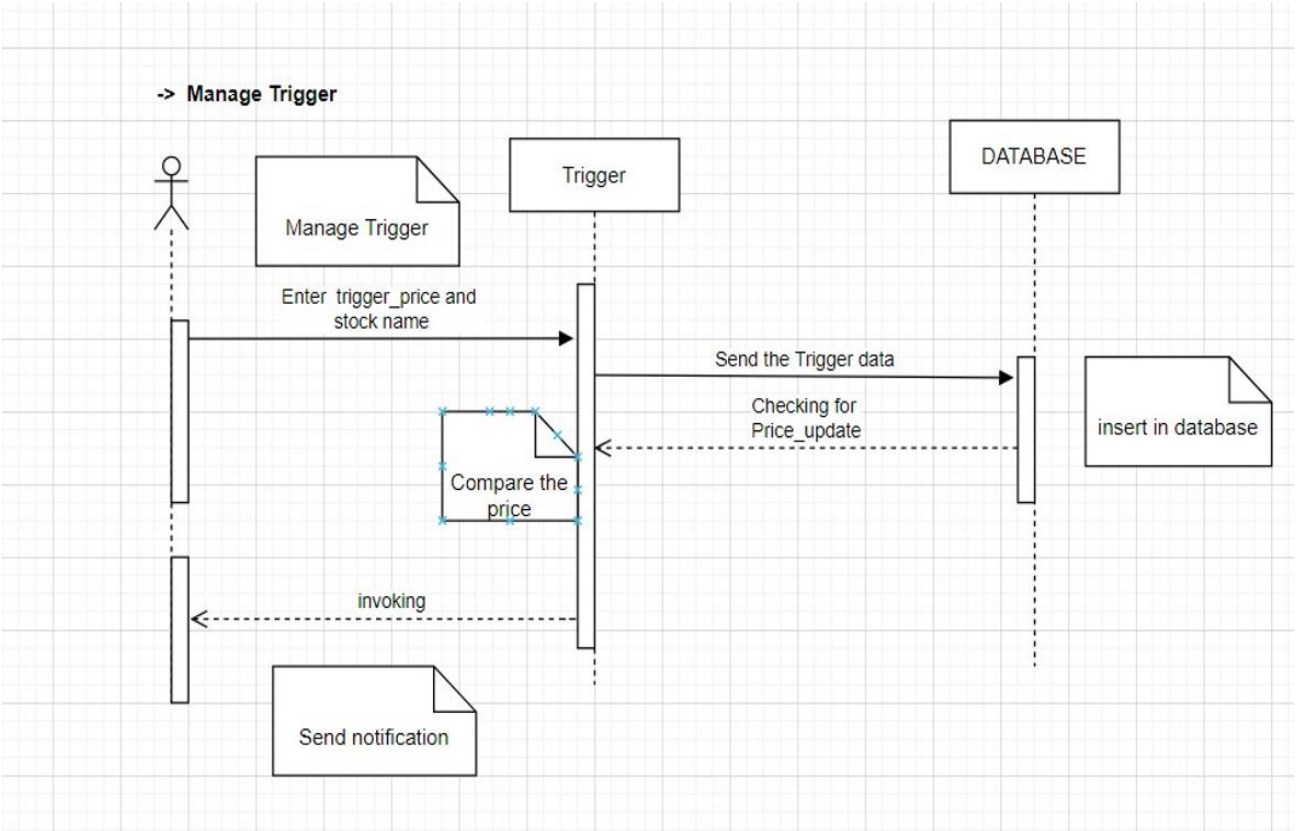


Sequence Diagram :

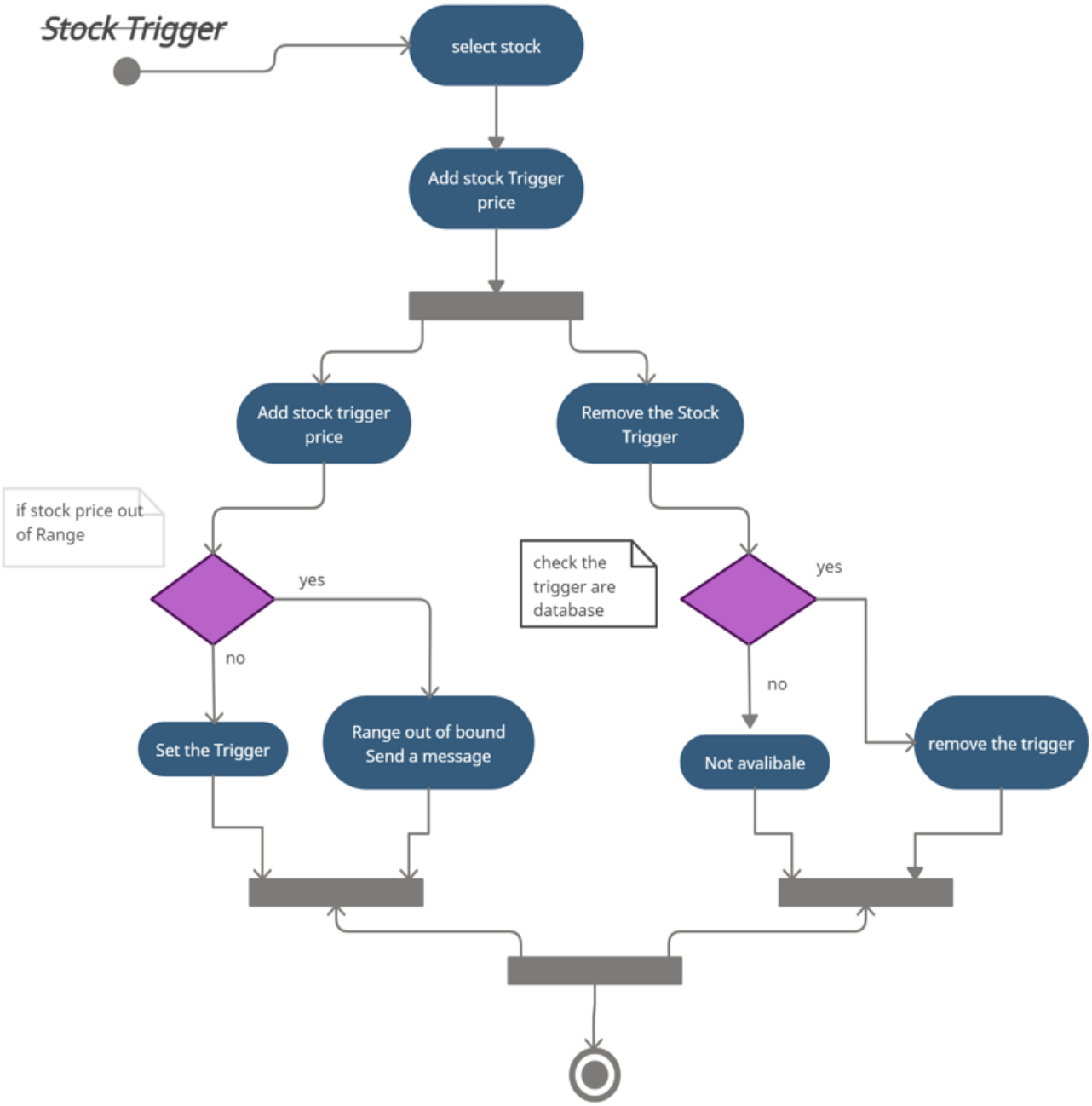


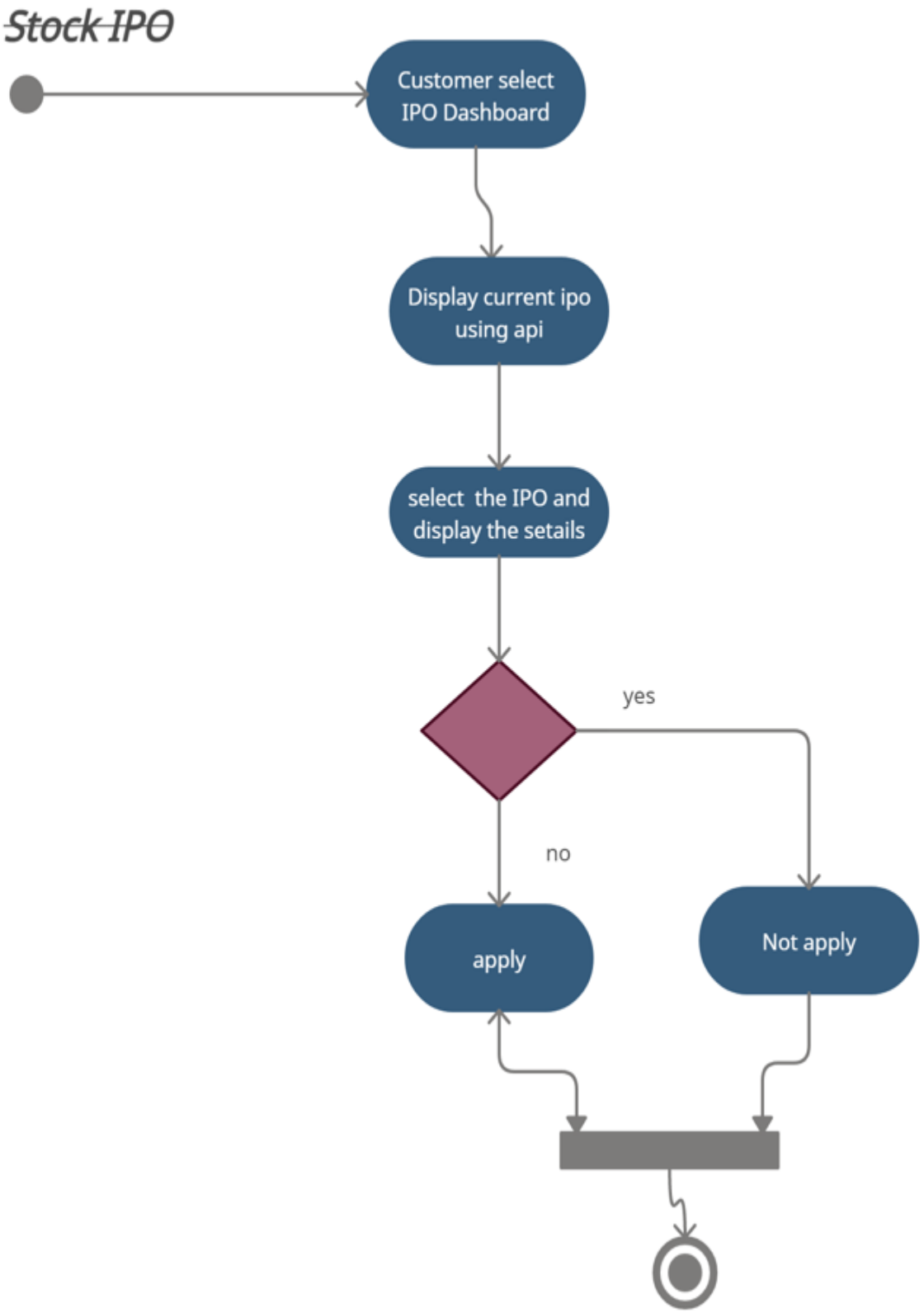



-> Manage WatchList
watchList
DATABASE
Select Stock to add
Send Stock_id or name
insert in database
return result
Show stock in watchlist
Remove stock from watchlist
Send Stock_id or name
delete from database
return result

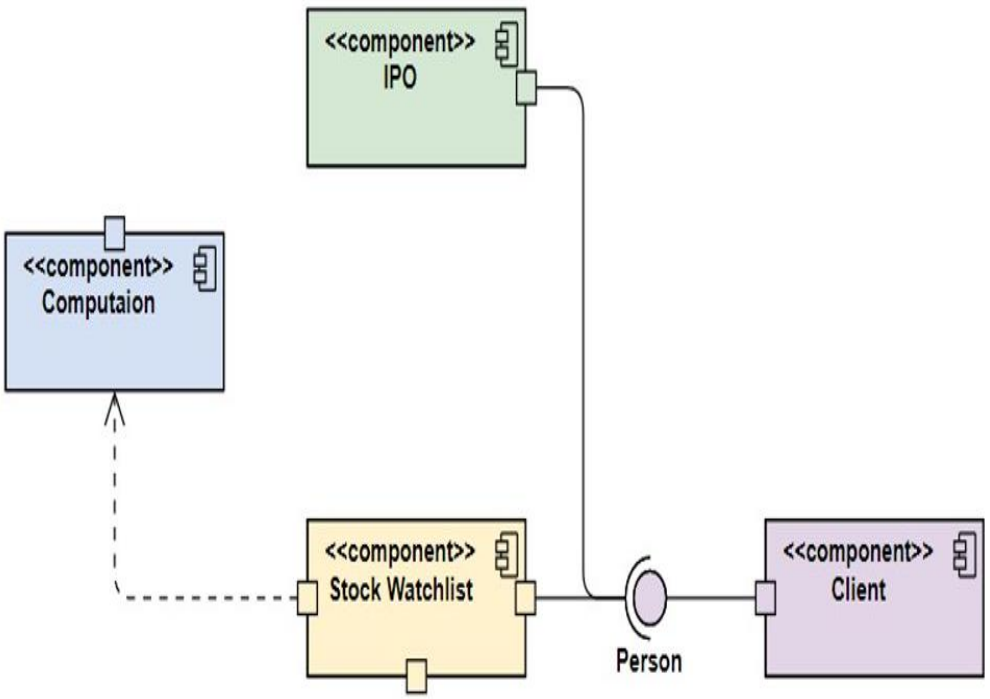

-> Manage Trigger
Trigger
DATABASE
Manage Trigger
Enter trigger_price and stock name
Send the Trigger data
Checking for Price_update
insert in database
Compare the price
invoking
Send notification

ACTIVITY DIAGRM :

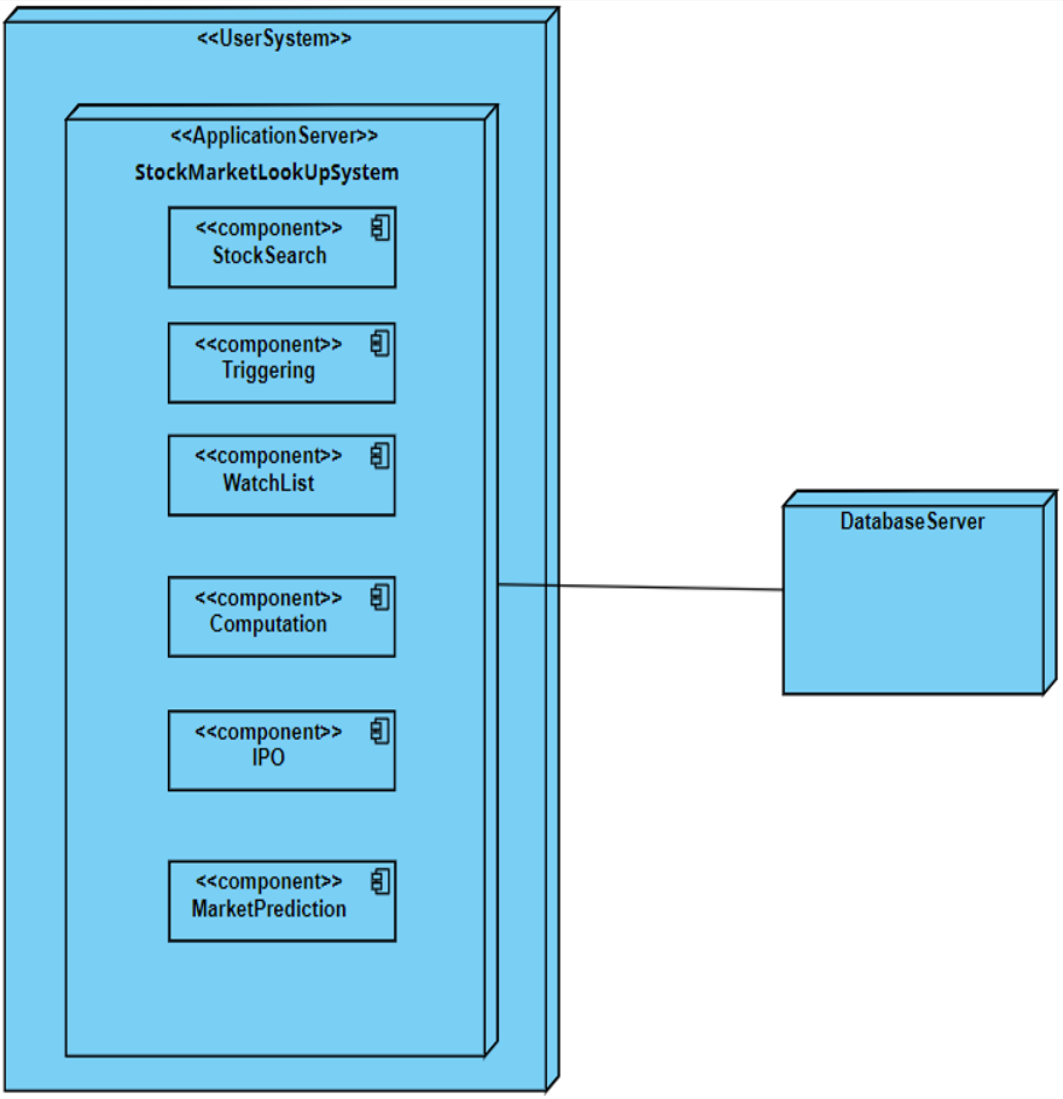

Stock Trigger
select stock
Add stock Trigger price
Add stock trigger price
Remove the Stock Trigger
if stock price out of Range
check the trigger are database
yes
no
yes
no
Set the Trigger
Range out of bound Send a message
Not avalibale
remove the trigger


Stock IPO
Customer select IPO Dashboard
Display current ipo using api
select the IPO and display the setails
yes
no
apply
Not apply

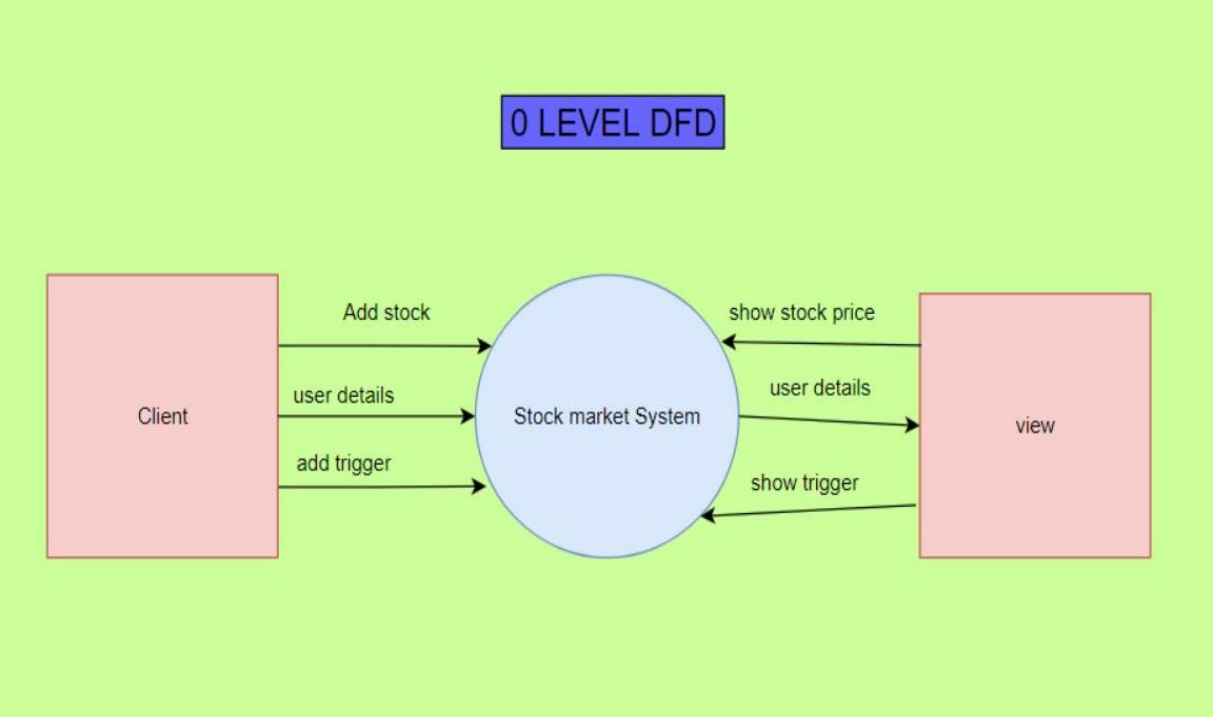
Component Diagram :

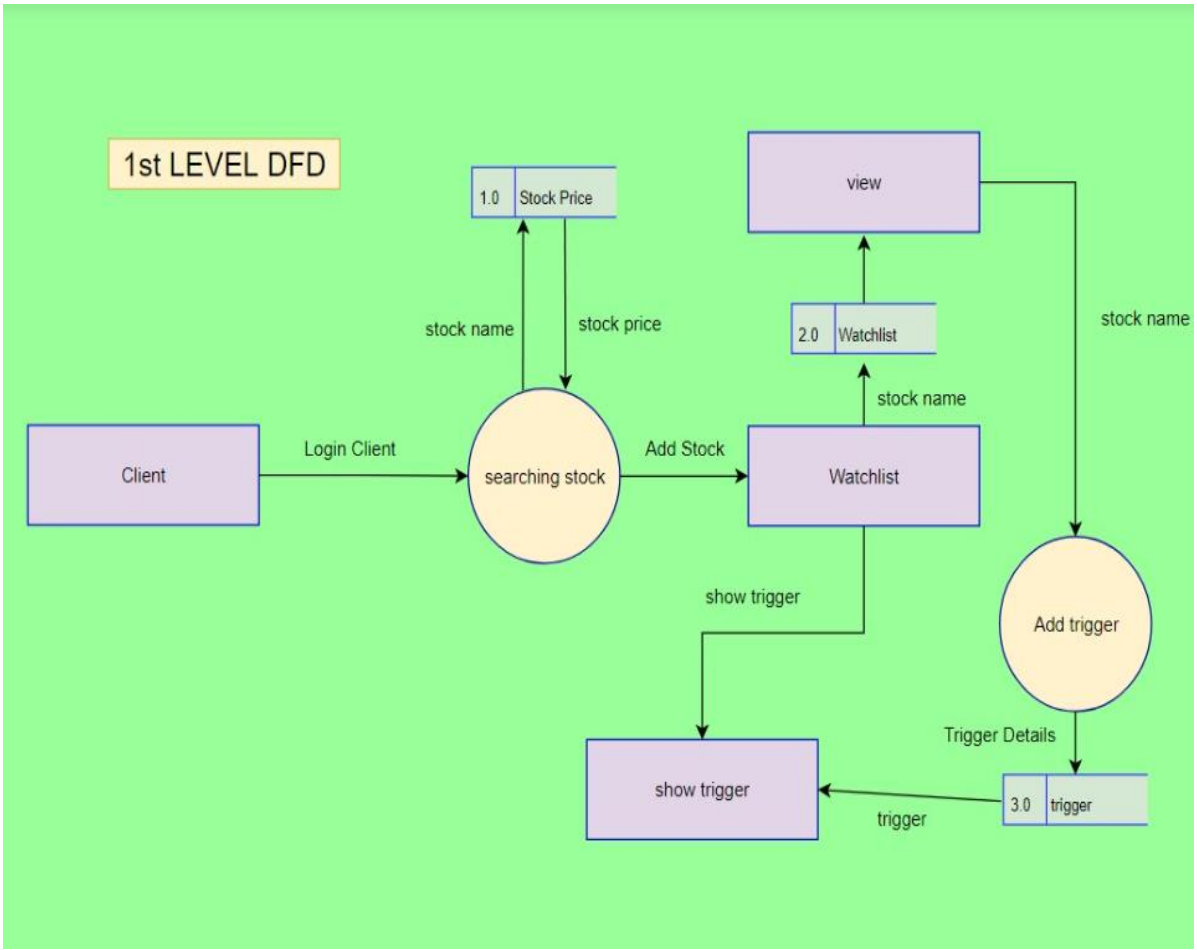


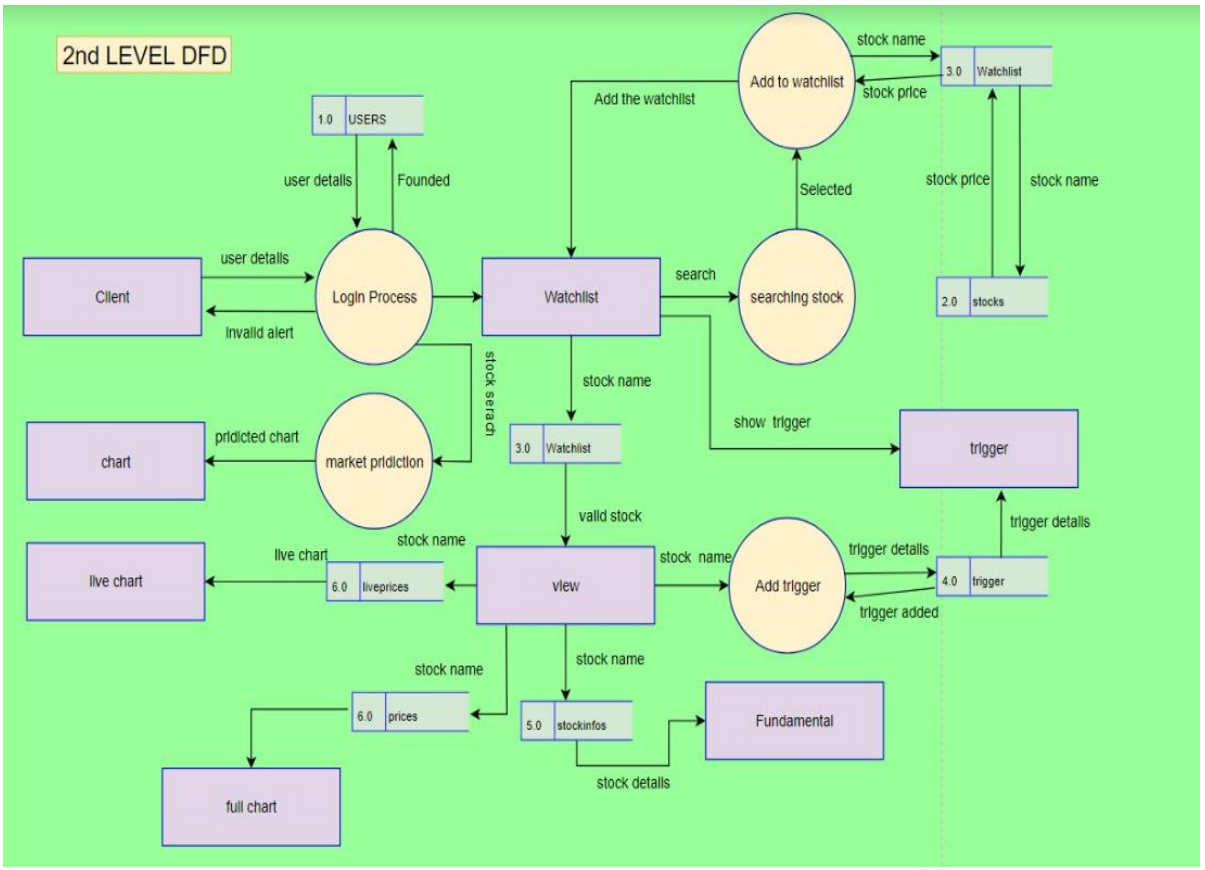
Deployment Diagram :



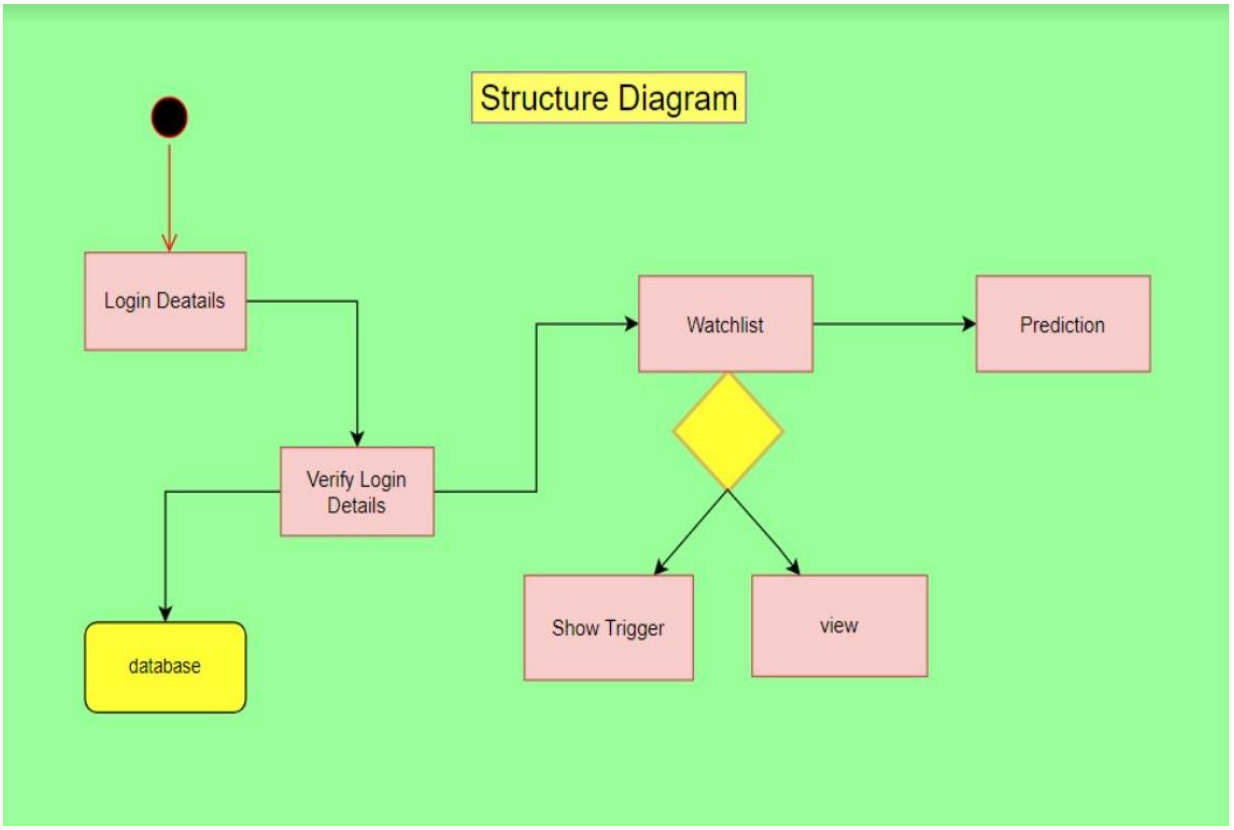
DFD :


0 LEVEL DFD
Client
Add stock
user details
add trigger
Stock market System
show stock price
user details
show trigger
view


1st LEVEL DFD
1.0
Stock Price
stock name
stock price
Client
Login Client
searching stock
Add Stock
Watchlist
view
2.0
Watchlist
stock name
stock name
show trigger
Add trigger
Trigger Details
show trigger
3.0
trigger
trigger



Structure chart :



Implementation Details

→ we have implemented a system in the MERN stack. We have used a highchart library for showing data. used Material-UI library for designing.

→ we used Alpaca api for fetching all price data and Polygon.io api for fetching Fundamental company data.

→ we have used JWT authentication for registering and login users. in that we send email & password details to the backend. so it generates the token and that token sends it to the frontend and saves it to localstorage of the browser. when we do another request then we put that token into the header of every request which is sent to the backend. so, the backend verifies it and receives the userid from that token.

→ we have a select-option bar in which users can select the stocks and it to the watchlist (limit is maximum 5 stocks)

→ stock charts show OHLC data from 1 jan,2018 to up-to date.

→ here we have used US stock market data. so the market opening time of the US market in India is 7:00 pm to 1:30 am. so, at that time we can see a live chart (Candlestick bar chart) which is updated every 1 minute.

→ we use a highcharts library which provides a wide range of charting tools. like, we can view full-screen, export chart data as image file and also .csv file, also (line,candlestick,ohlc) chart, we can set the flag & labels, draw the resistance and support line, many indicators we can apply and do technical analysis. also select the (range) 1m,3m,1y,3y data for analysis.

→we have created a IPO link where user can see all the details about IPO.

→ we can add the trigger (providing upper and lower limit). in that when current live price is going outside of this range then we mail the user about the trigger executed. no limit in adding no. of triggers.

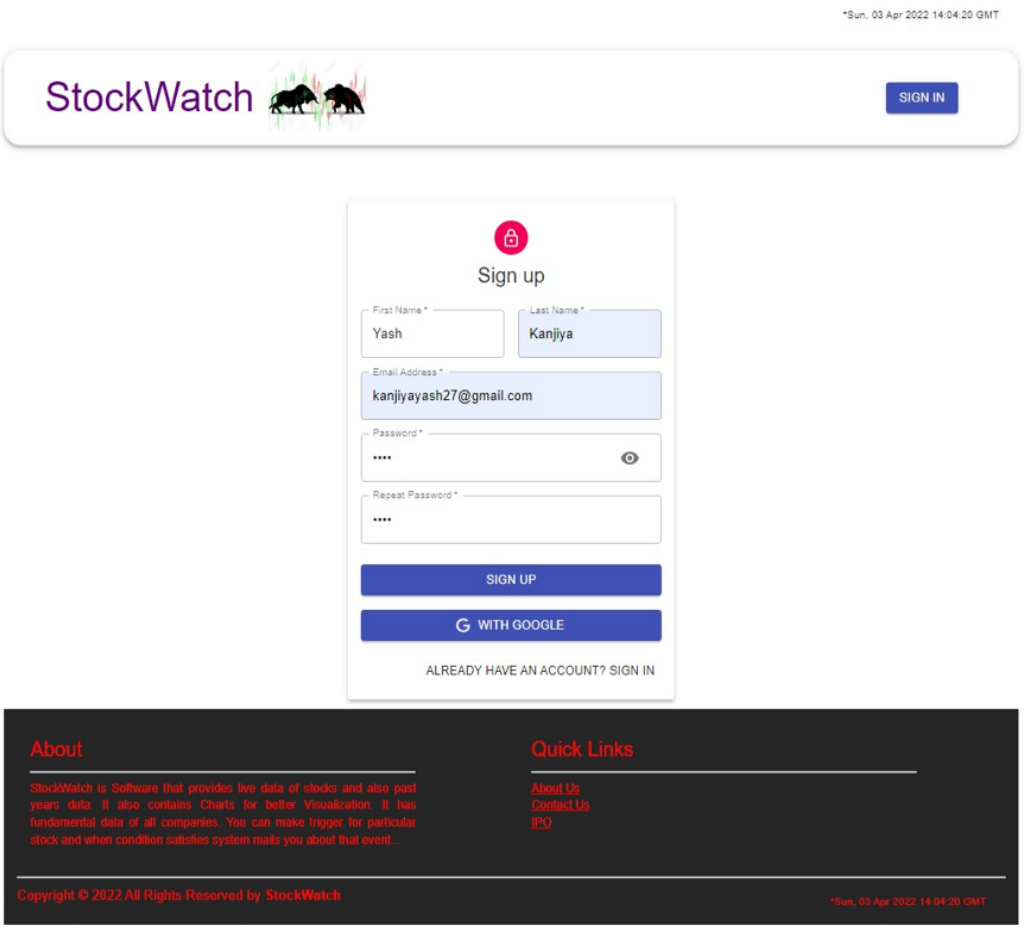
→ we have provided only 500 stock details which are in the list of S&P 500 index. fetched data is combined from many exchanges. Clock is present in our web-app showing UTC (Universal Time) time format.

→we have used streamlit App framework for stock prediction and manage portfolio.

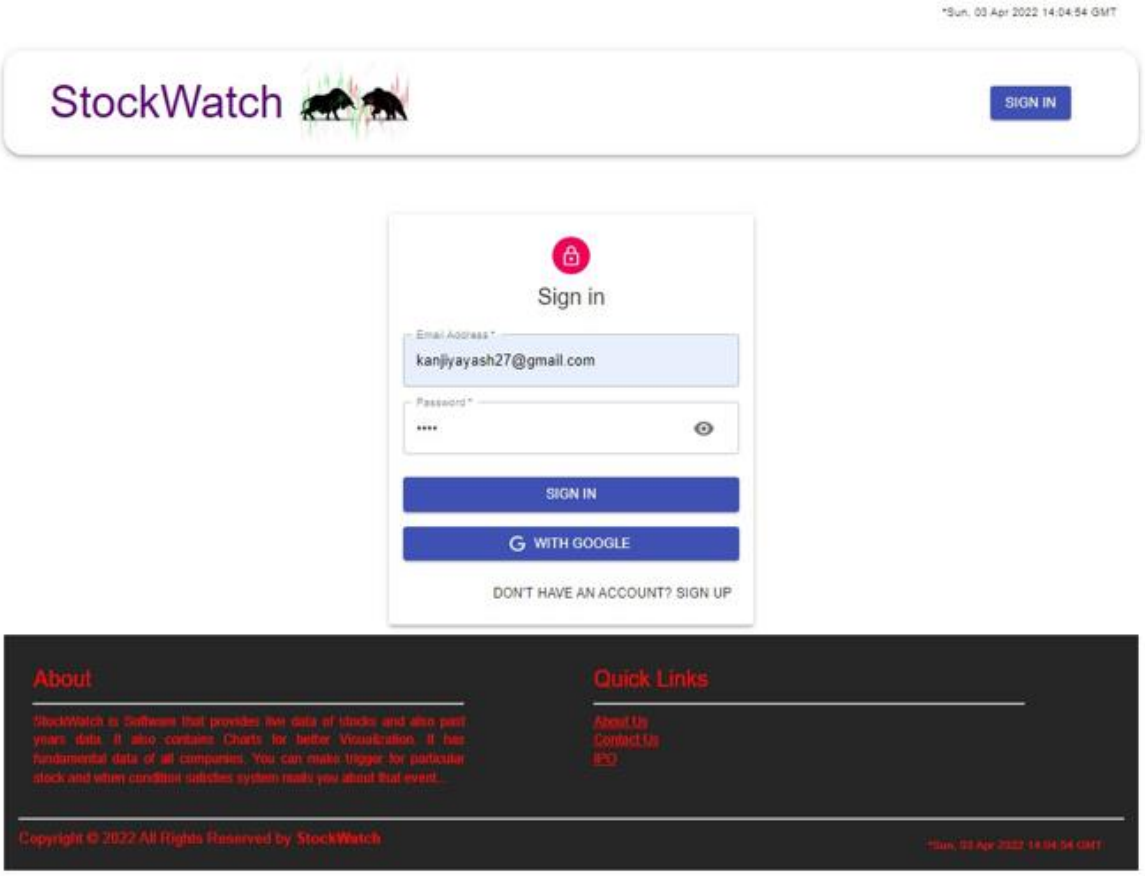
→In portfolio part user can see the assumed price after selected time range and get the net profit from invested amount.

Screen-shots of the System :

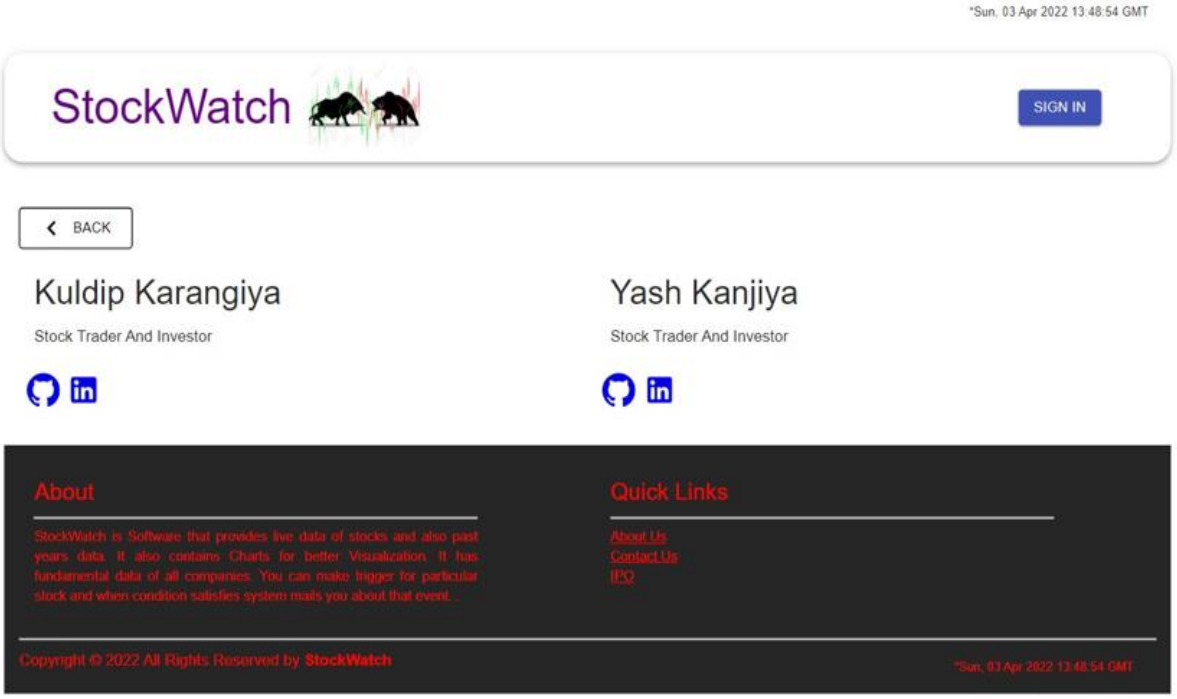
Sign Up :



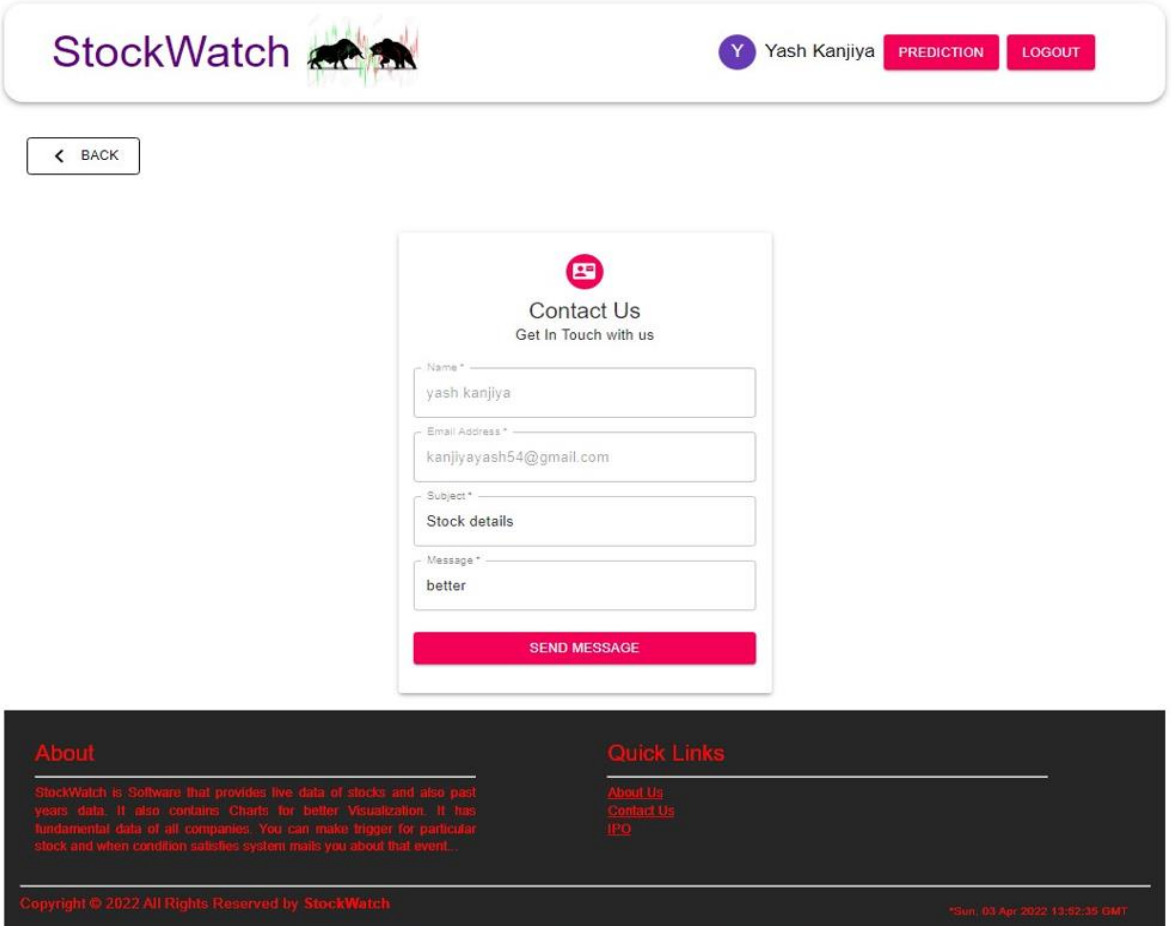
Log In :



About Us :



Contact Us :



Watchlist :

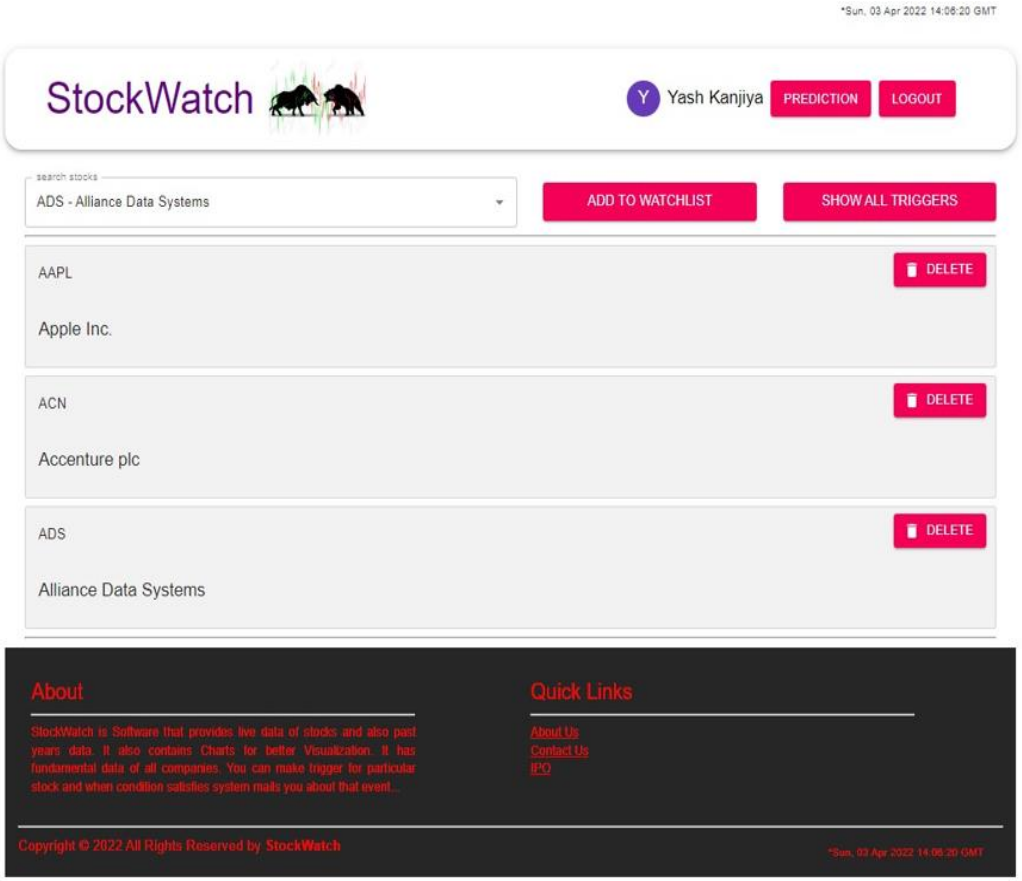
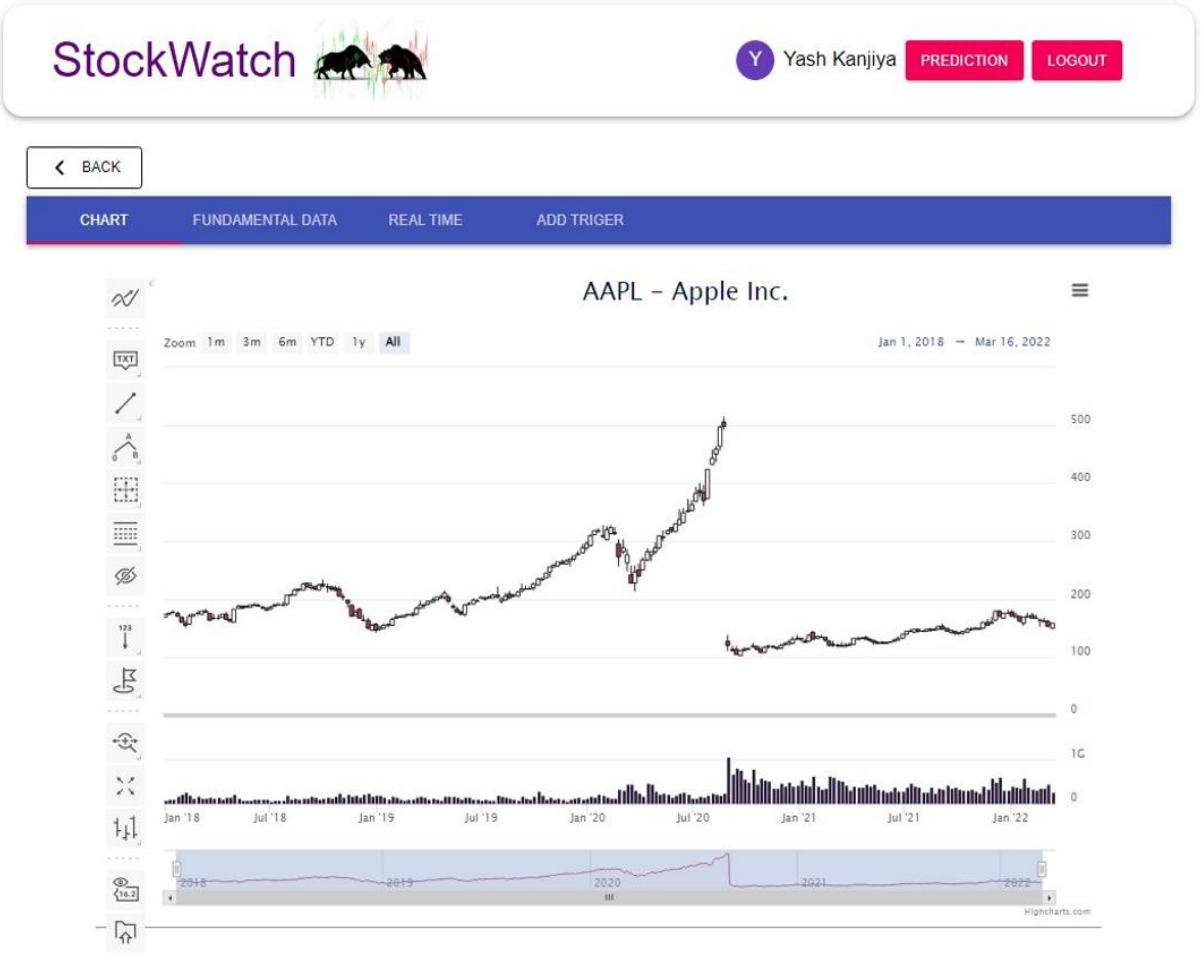
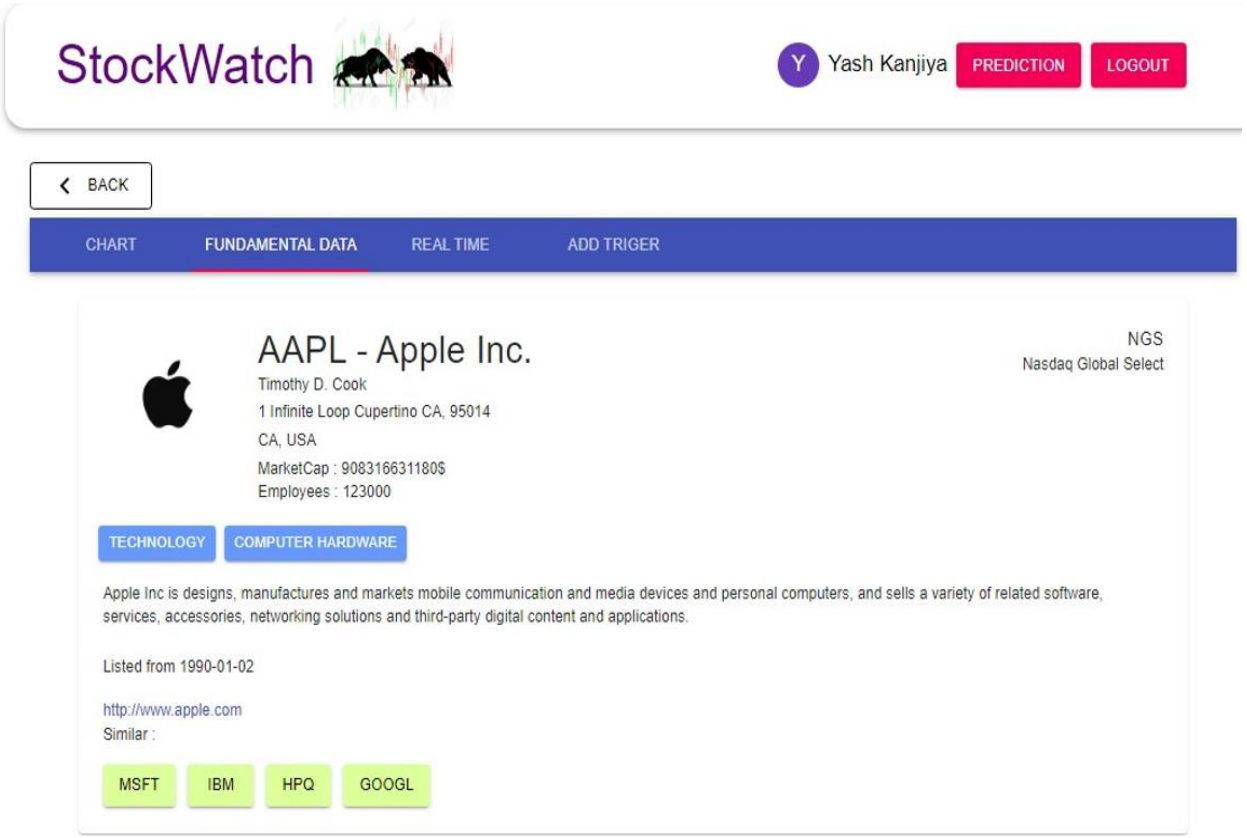


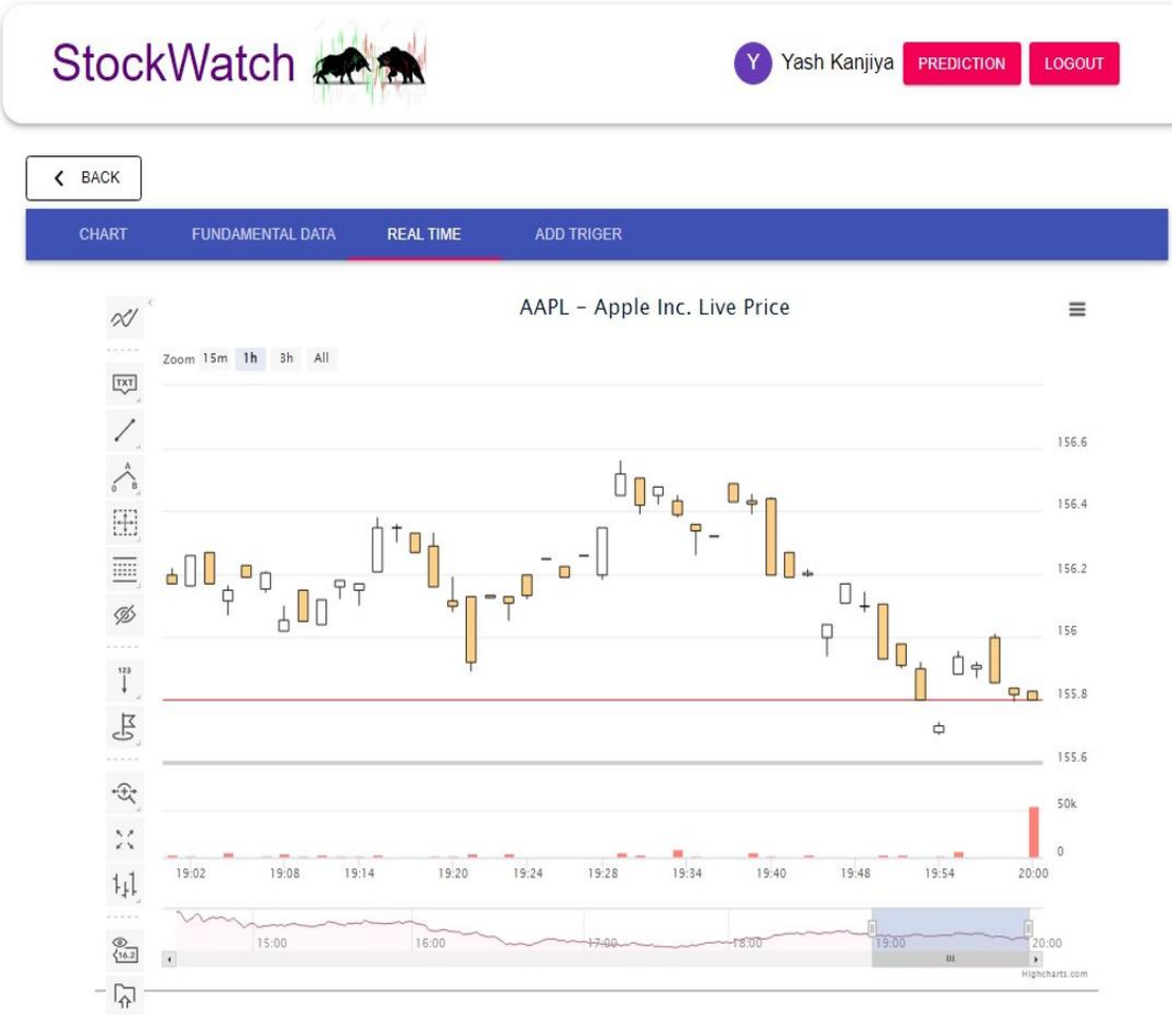
Chart :



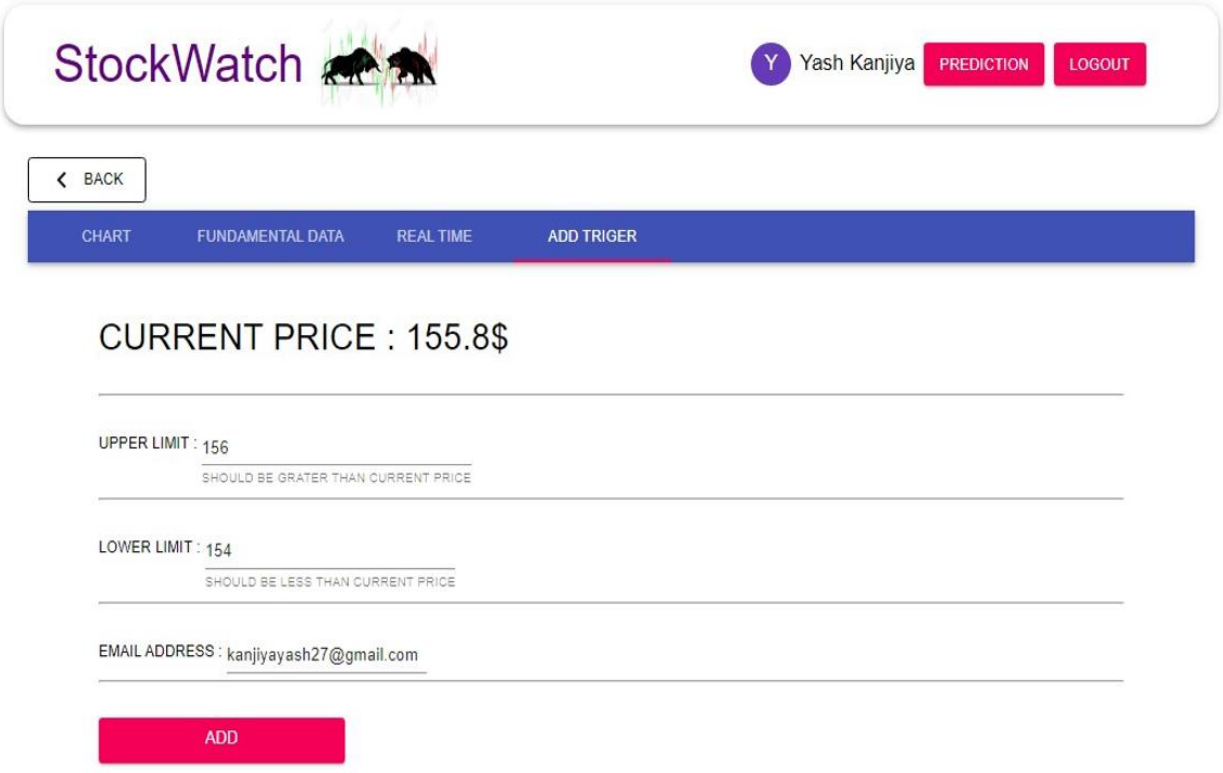
Fundamental Data :



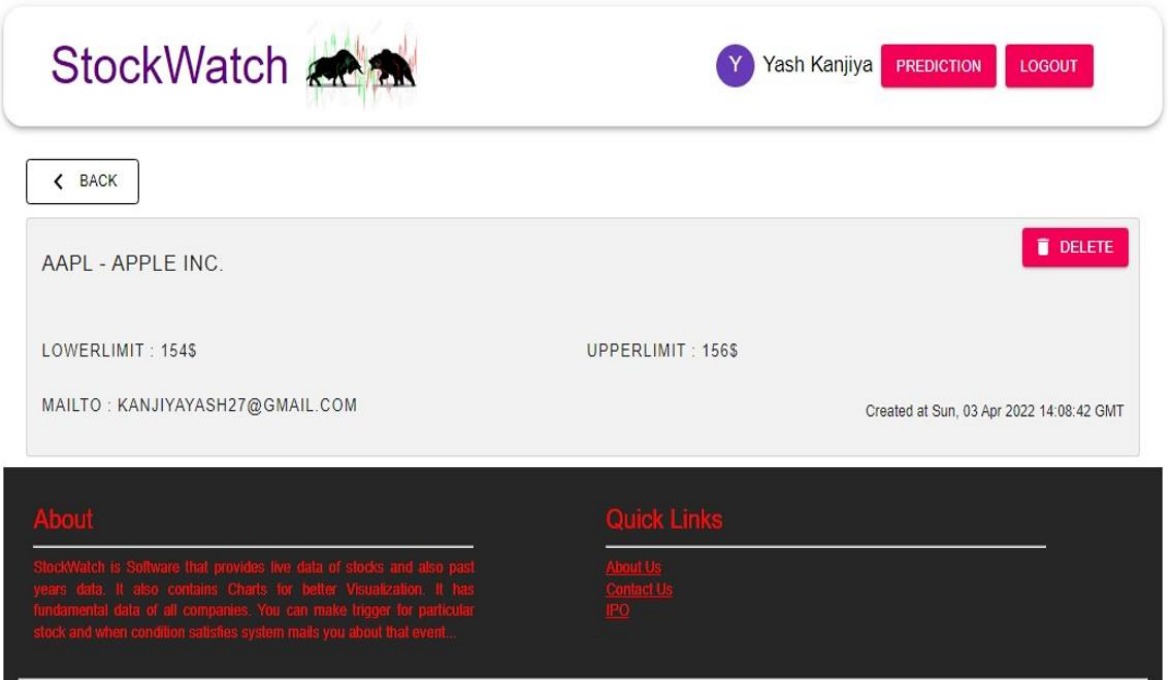
Real Time :



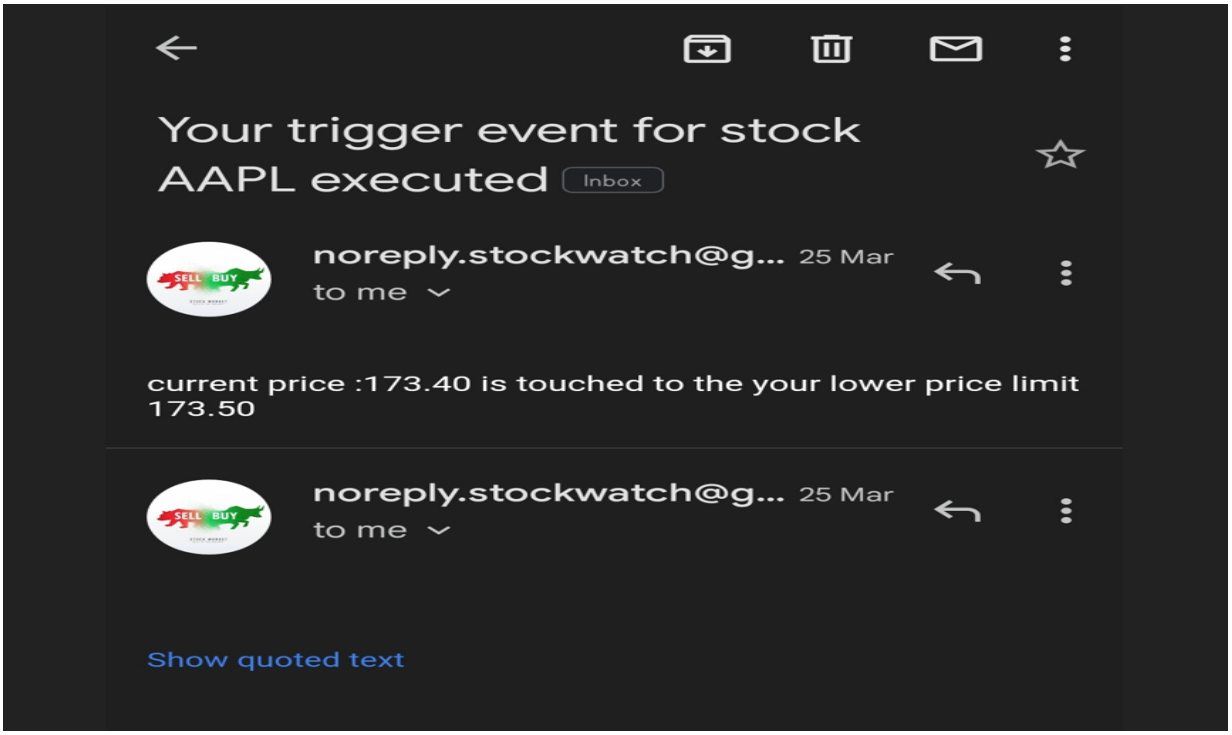
Add Trigger :



Show Trigger :



Mail to User :



IPO :

IPO (Initial Public Offer) in India - Explained in Brief:-

IPO is the short form for the Initial Public Offering. It is a sale of shares by a company to the public for the first time. Without an IPO a company remains a privately held company. The privately held company have a small number of shareholders. These shareholders may range from promoters, who are the owners of the company, their family members, friends, and relatives. Some strategic or professional investors may also hold the shares of a private company like venture capitalists, angel investors or private equity investors.

However, after the IPO process which informally is also known as going public, the number of shareholders increases multifold, when compared to the privately held company. IPO allows individuals like you and me and institutional investors to purchase and sell shares of the company. With this initial offering, a company's share becomes eligible for a stock exchange listing. Stock exchange listing facilitates in buying and selling of such shares of companies.

Advantages of IPO for a company

- Large and diverse group of investors
- Lower cost of capital to the company
- The largest amount of capital to be raised compared to other options
- Exposure, prestige, and public image of the company improve. This may help speed up growth
- May attract and retain better management and skilled employees
- Helps in further acquisitions (potentially in return for shares of stock)

Disadvantages for a company going public

- Disclosure and compliance requirement in financial, accounting, tax, and other business information field increases. This results in a significant rise in legal, accounting and marketing costs.
- Also, legal and regulatory risks such as lawsuits and shareholder actions rise. Moreover, time and efforts required of management for reporting rises
- Many information which may be confidential in nature goes to competitors, suppliers and customers offer listing.
- The company may lose control to new shareholders, who might obtain voting rights and can effectively control company decisions through the board of directors

What to look before investing through an IPO?

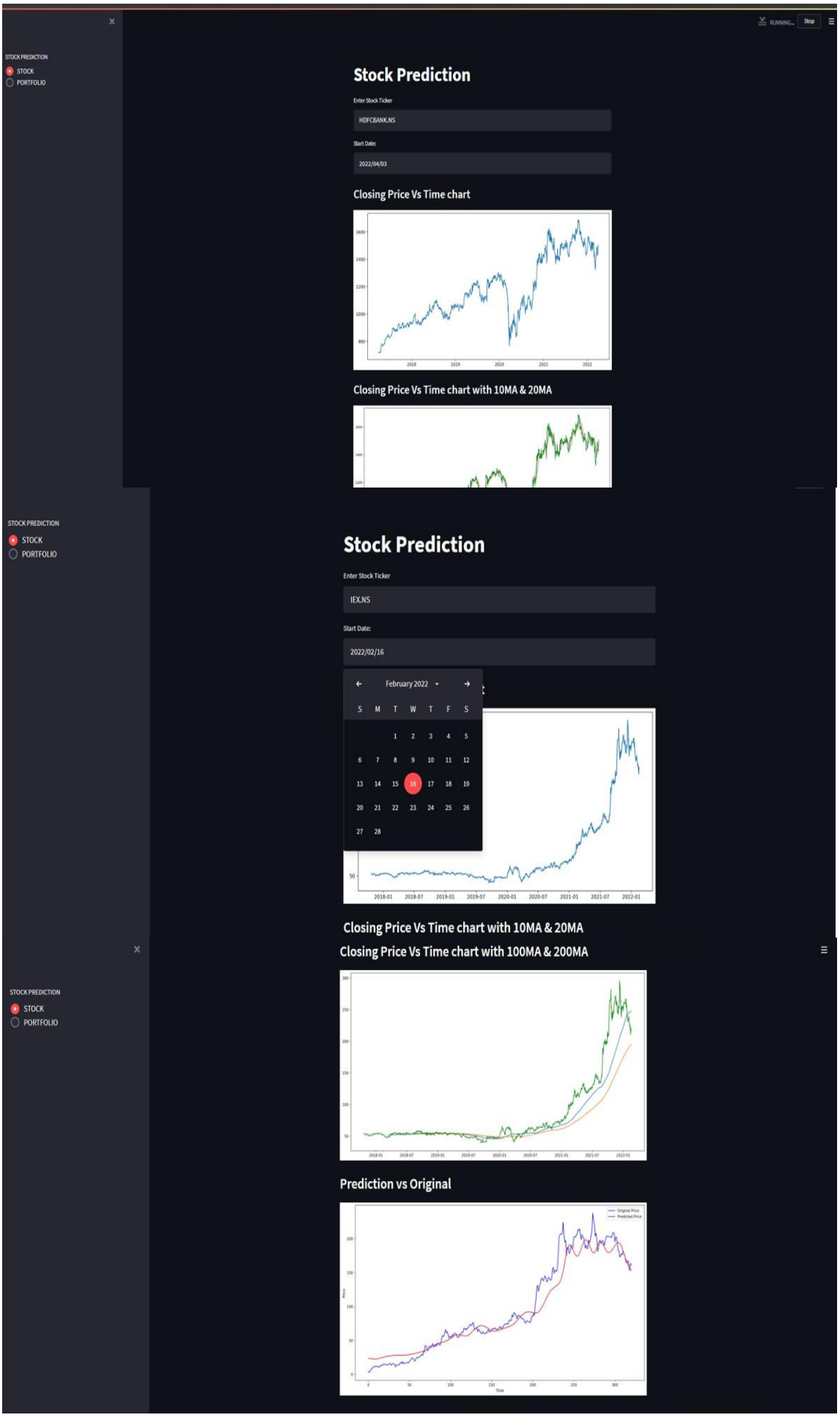
Analyzing a well-established listed company from an investment point of view is a tough job. It became more tough and trickier if you are going to analyze any company whom you want to invest through its IPO. There is almost nil historical information available on the public domain to start with. The only source of data abou such companies is their red herring prospectus (RHP) or the draft RHP. Thus, this is the document which needs to be scanned with utmost care before making any investment decision in the said IPO. Particularly one should look for the following in any company before making a decision to invest.

Points to consider before investing through IPO

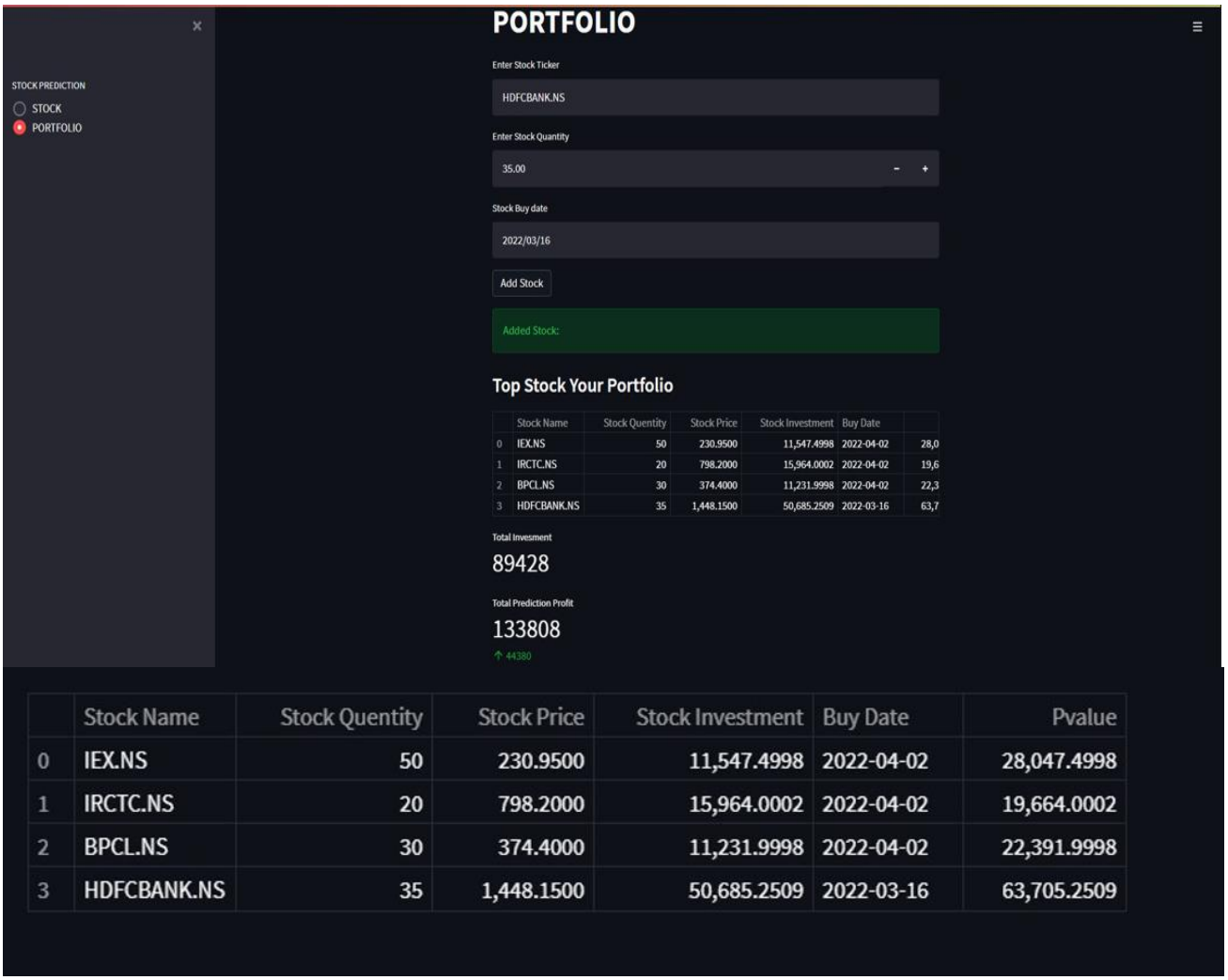
- Usual information on the management team, their qualification, experiences, legal aspects of the management personals and their expertise. A good company is one whose management have expertise and experience in line with the business of the company they are handling.
- Consider how the management plan to use the funds generated from the IPO if it is not offered for sale (OFS). In an OFS no proceeding from the sale of the share goes to the company, rather it goes directly to the selling shareholders. If the money collected through the IPO is to repay long-term debt or it is being used for expansion of the business in the phased manner it is generally considered good.
- Compare with the performance of similar companies who are already public. This will give you a sense of the overall performance of the sector in which the company is operating. Particularly look for margins of the companies and variations in Q-o-Q revenue of different companies, if any.
- Always invest in an IPOs that have good operating margins and at the same time lower valuation in comparison with established and reputed peers.
- Also, give proper attention to the quality of the underwriters and their deals with the company. IPOs that are successful are generally supported by brokerages who are capable to promote new issue well. Smaller investment banks generally underwrite any company and thus bear no valuable information for investors.

The underwriters and investment banks act like salesmen to an IPO they are dealing with. They stand in between companies and investing public. Actually, a company can't directly go to the market to sell their share. It needs to be done through proper mechanism.

Stock Prediction :



Portfolio :



	Stock Name	Stock Quentity	Stock Price	Stock Investment	Buy Date	Pvalue
0	IEX.NS	50	230.9500	11,547.4998	2022-04-02	28,047.4998
1	IRCTC.NS	20	798.2000	15,964.0002	2022-04-02	19,664.0002
2	BPCL.NS	30	374.4000	11,231.9998	2022-04-02	22,391.9998
3	HDFCBANK.NS	35	1,448.1500	50,685.2509	2022-03-16	63,705.2509

Conclusion

→ All the main functionality of the system we have completely implemented.

→ we learn about the user registration and login and more about how JWT works and use it in Authentication and Authorization.

→ learn about how to get data from api and store it in a database and convert json objects to a good readable or maintainable format for our mongoose models. even if we have a rate limit in api calls for fetching data and manages that using javascript setInterval method.

→ learn about highchart library and its options for showing data in charts.

→ learn about cron-job which is used for cleaning and many more…

→ learn about stock prediction using streamlit app framework and keras python library and numpy and pandas for manage portfolio.

→Our App is very useful for trading quants who want to make an instant move to change their position in the market when the trigger is executed. and also users can also apply many indicators and apply technical analysis using our advanced chart and stock tools.

→ Our App is very useful for the user who wants to see the predicted value of stock price after given range(like six month, one year, etc.)

→In Our app user can see the ROI(return of investment) using predicted price.

Limitations and Future Enhancement

→ apps are limited to only a few stocks (500). we can make it adding more stocks in this web-app.

→ we can also make comparison charts between 2 to 3 stocks.

→ we can add the latest news and info about the company. so our users can also catch it and react to the market with that environment.

→ we can improve our triggering system by increasing computational power.

→ we can improve the notification system which is currently only using mail.

→In the future, we can also update the clock based on the user's local region timing.

→we can improve the accuracy predicted price.

Reference / Bibliography

https://www.youtube.com/

https://www.google.com/

https://stackoverflow.com/

https://alpaca.markets/docs/

https://polygon.io/docs/getting-started

https://www.nasdaq.com/

https://docs.mongodb.com/

https://reactjs.org/tutorial/tutorial.html

https://material-ui.com/

https://console.cron-job.org