STOCK INFO AND PREDICTOR

UCS503 Software Engineering Project

Report End-Semester Evaluation

Submitted by:

Sarthak Dhonchak **(**101803434**)**

Arpit Garg **(**101803450**)**

Manan Garg **(**101803451**)**

Yash Mittal **(**101803457**)**

Group 5

BE Third Year, COE

Submitted to:

Ms. Kanu Goel

Lecturer(NT)

Computer Science and Engineering Department TIET,

Patiala

November 2020

TABLE OF CONTENTS

Contents **Page No**

1. Project overview …… 3

2. Software Requirement Specification (SRS…… 4

3. Structured Analysis…… 13
   - 3.1 Data Flow Diagrams
     - 3.1.1 DFD Level 0
     - 3.1.2 DFD Level 1
     - 3.1.3 DFD Level 2
   - 3.2 ER Diagram

4. Object Oriented Analysis……. 17
   - 4.1 Use Case Diagram
   - 4.2 Use Case Templates
   - 4.3 Activity Diagram
   - 4.4 Collaboration Diagram
   - 4.5 Sequence Diagram
   - 4.6 State Chart Diagram
   - 4.7 Component Diagram
   - 4.8 Deployment Diagram
   - 4.9 Class Diagram

5. Testing…… 32
   - 5.1 Test Plan
   - 5.2 Test Case Report
   - 5.3 Screenshots

1. Project Overview

This Website will be extremely useful for People with basic financial knowledge. This Website will provide the user with all the information required for studying any stock. This Website will fetch data from online stock market database using an API and the user will have an easy and interactive environment. The user can get all the information about a particular stock with the current price and a continuous graph depicting the highs and lows of the particular stock. All the information user needs will just be a click away. The Website also provide feature like saving any favorite stocks and getting there information only so user doesn't need to search for them repeatedly.

With such ease of data to the user this website will also help user in predicting the values of a particular stock by which user can have an ease of purchasing any particular stock. User can find the profit or loss and predicted price of the stock. For our Website LSTM Machine Learning model is used in an attempt to predict values of the stock. No prediction can be accurate but this will give a brief idea to the user about the company by giving the predicted price and the information about it.

Currently this is a small project but it can be changed to large scale in the future and it can be used for continuous display of data and fast responses with close to precise values which can help users significantly.

Software Requirements Specification

for

Stock Info and Predictor

Prepared by

Group-5

13-09-2020

Table of Contents

1. Introduction..... 6

1.1 Purpose..... 6

1.2 Project Scope..... 6

1.3 References..... 6

2. Overall Description.....7

2.1 Product Perspective..... 7

2.2 Product Features..... 7

2.3 User Classes and Characteristics..... 7

2.4 Operating Environment..... 7

2.5 User Documentation..... 8

2.6 Assumptions and Dependencies..... 8

3. System Features..... 9

3.1 Displaying the Prices of stocks 9

3.2 Predicting the Prices of stocks..... 9

4. External Interface Requirements..... 10

4.1 User Interfaces..... 10

4.2 Hardware Interfaces..... 11

4.3 Software Interfaces..... 11

4.4 Communications Interfaces..... 11

5. Other Nonfunctional Requirements..... 12

5.1 Performance Requirements..... 12

5.2 Safety Requirements..... 12

5.3 Security Requirements..... 12

5.4 Software Quality Attributes..... 12

1. Introduction

1.1 Purpose

The purpose of our project is to provide customers with easy access to statistics of a particular stock in the stock market, we enable users to select their favorite stock of which we'll tell him/her the most predicted value of that in future.

1.2 Project Scope

People with basic financial knowledge are the primary targeted audience this enhances them with the right choice of the stock in which they can invest on.

- Users are provided the platform on which they are provided with the most appropriate values of the stock, so this can be used as a financial model.
- We provide the predicted value of the stock. So that common people with the basic financial knowledge can choose wisely while trading.
- Users can manage their stock under their portfolio.

1.3 References

The data for the stock has been taken from the API.

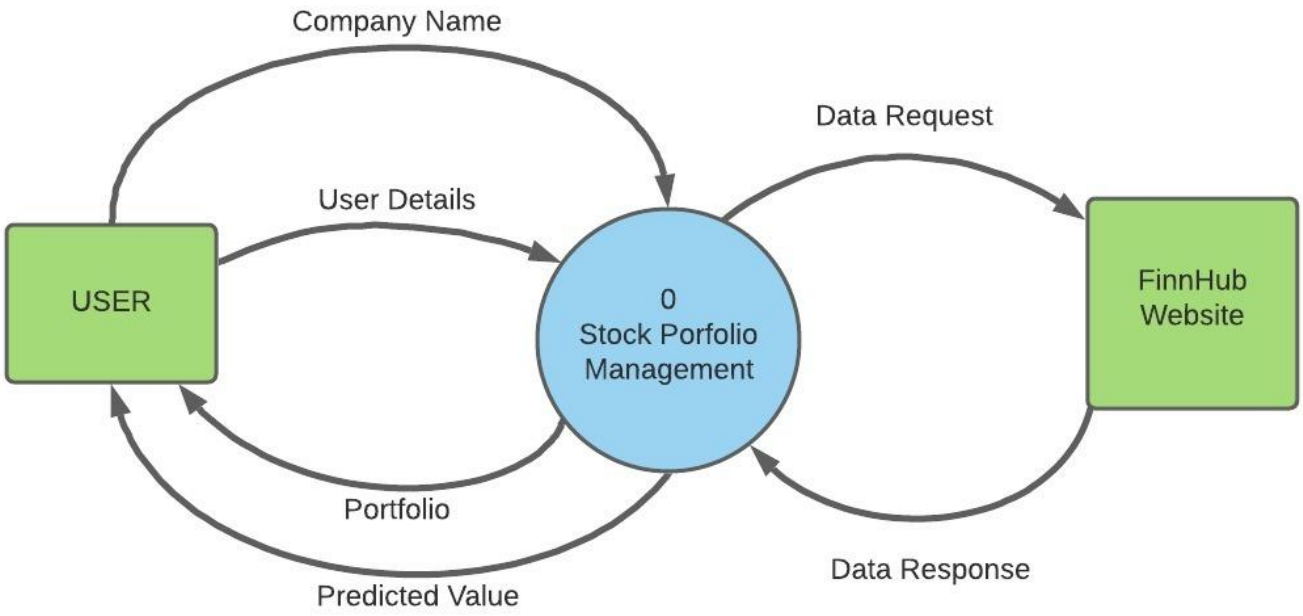
- Complete documentation of the API. https://finnhub.io/docs/api
- key for the API. https://finnhub.io/register

The prediction is based on the LSTM model in ML using python.

2. Overall Description

2.1 Product Perspective

This is an self-contained product which uses the preexisting API for the data of the stock



Level 0 Data Flow Diagram of
Stock Prediction System

The very basic data flow diagram of the project can be stated as follows,

2.2 Product Features

- We'll be providing a portfolio to the user; users are first required to log into the site.
- In the portfolio users will be able to check out their favorite stock and their predicted values.
- Also, we'll be displaying user the stats of the particular stock on their demand.

2.3 User Classes and Characteristics

1. **User:** next one will be the normal user who can access the the other main features of the product.

2.4 Operating Environment

The software components used in our projects are:

- The frontend will be based on HTML5,CSS, Materialize, Jquery, and javaScript.
- For the back-end we are using Node and express as a framework
- The database we are using is MYSQL.

- For hosting we are going to host it locally.

2.5 User Documentation

a) You can enter any of the provided stocks from the list to get the actual price and the predicted price.
b) Predicted price may not be accurate, they are just for the reference.

2.6 Assumptions and Dependencies

- For the time being the ML model is not so dependable so predicted values might not be much trustable
- Since the data fetching is dependent on API's so speed of data displaying is dependent on API
- Also, currently this is a small prototype so the server can't handle too many loads.

3. System Features

3.1 Displaying the Prices of stocks

3.1.1 Description and Priority

Displaying the prices of the stocks which are either owned by the user or shortlisted to be put on watchlist by the user.
This is a high priority task

3.1.2 Stimulus/Response Sequences

Users need to sign in to their account to access the features.

3.1.3 Functional Requirements

3.1.3.1 The software needs to send requests to FINNHUB API, then parse the response sent by the API and present it to the user.

3.1.3.2 The API calls need to frequent so the data is up to date.

3.1.3.3 The data needs to be presented in a graphical manner using chart.js library.

3.2 Predicting the Prices of stocks

3.2.1 Description

Predicting the prices of the stocks which are either owned by the user or shortlisted to be put on watchlist by the user. This is a low priority task

3.2.2 Response Sequences

Users need to log in to their account and click on “more info icon” for the stock they want to get the predicted value of.

3.2.3 Functional Requirements

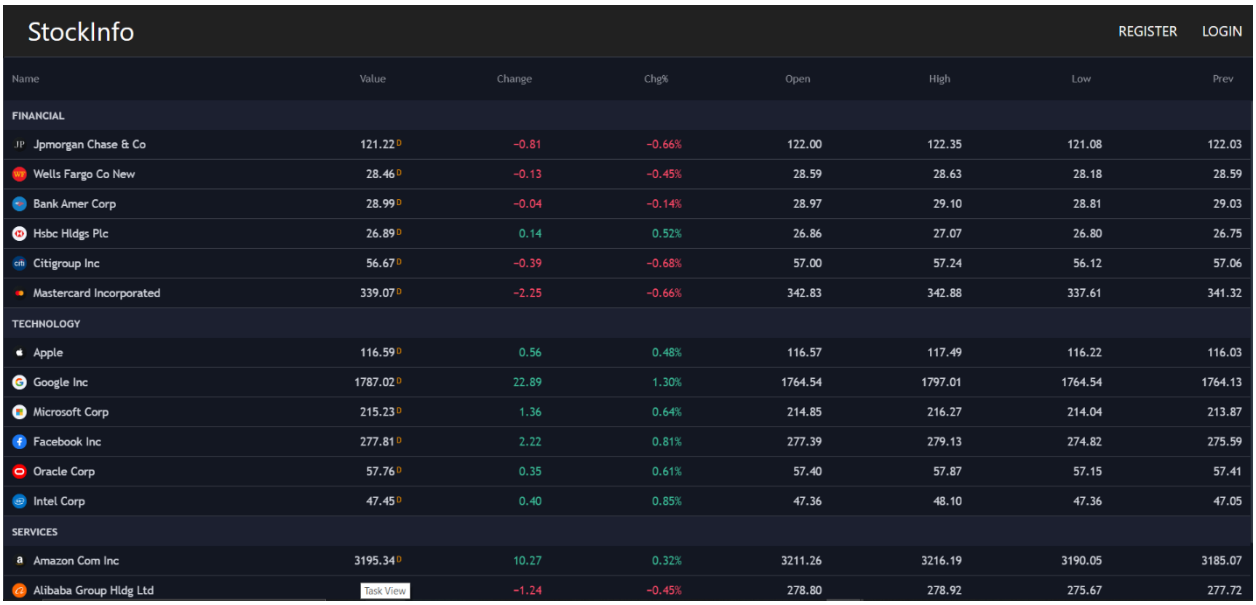
3.2.3.1 The software needs to access previous price history of the stock

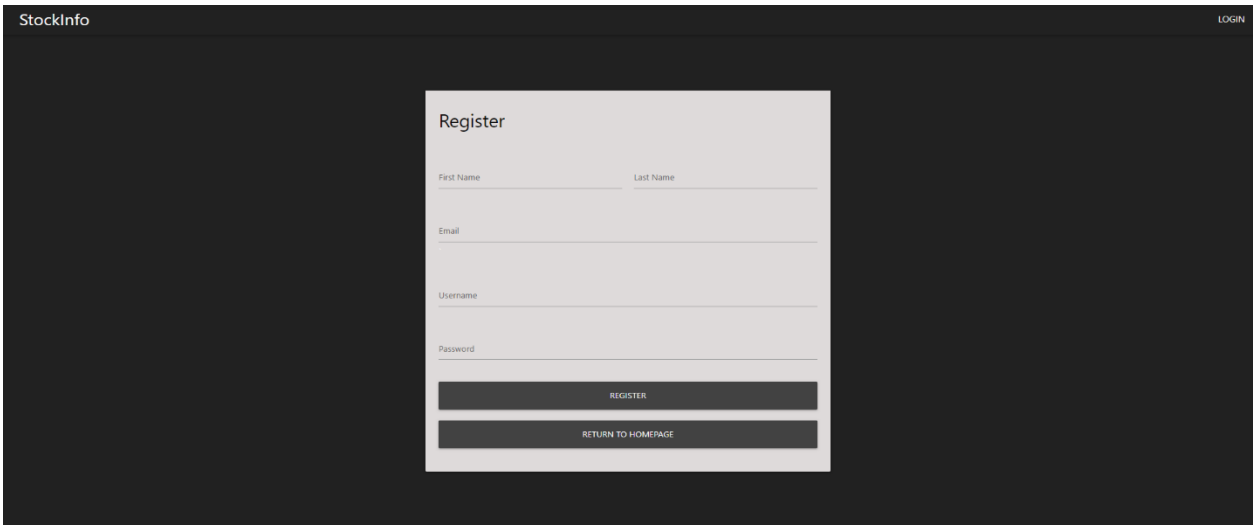
3.2.3.2 Process that information with the help of a ML model.

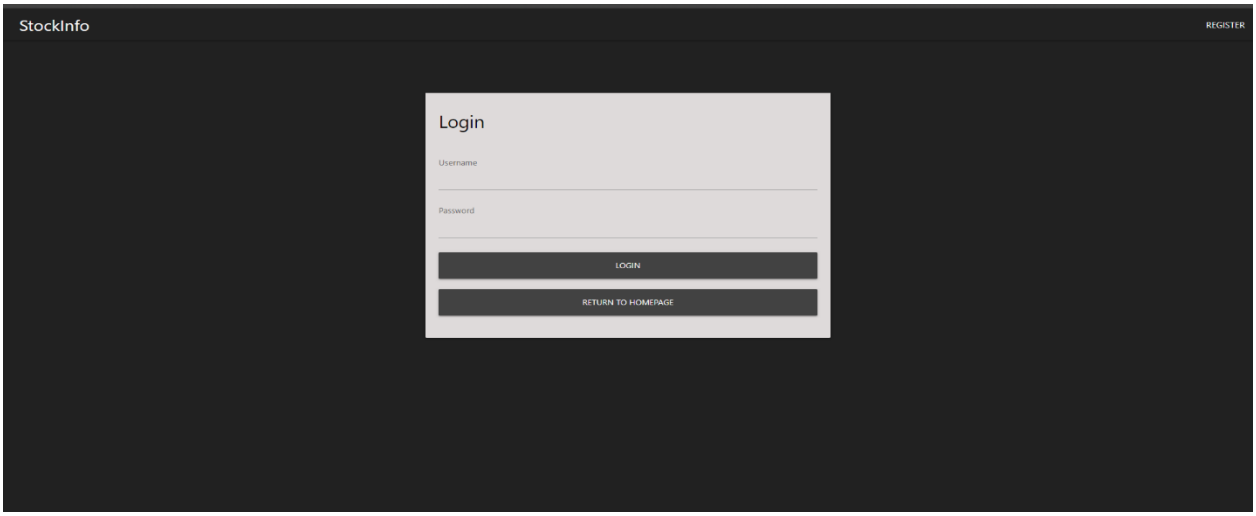
3.2.3.3 Display the predicted price.

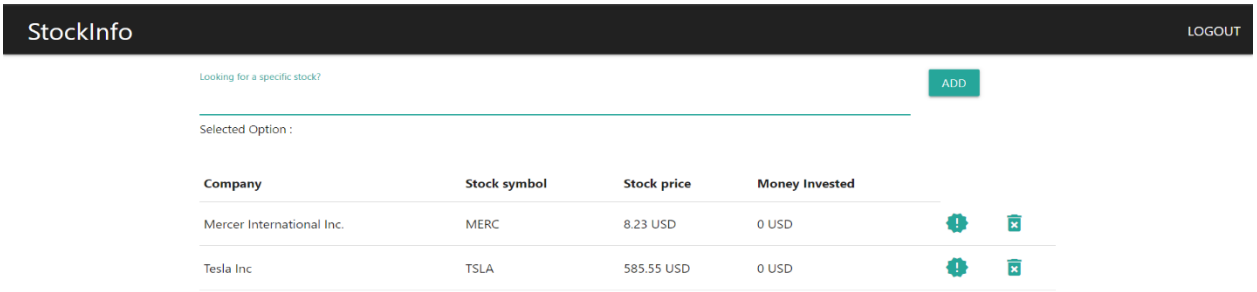
4. External Interface Requirements

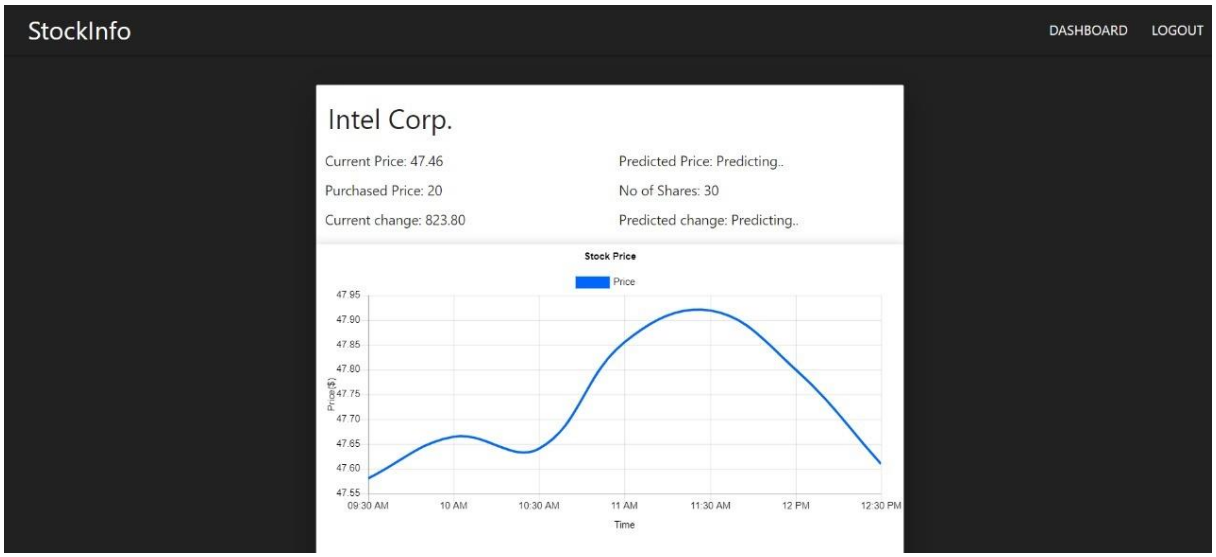
4.1 User Interfaces











4.2 Hardware Interfaces

4.2.1 The product requires graphics usage with just a simple keypad and mouse/touch for taking the user input.

4.2.2 The product does not require usage of sound or animation.

4.2.3 There are no specific OS requirements but a modern web browser is recommended and a desktop is preferred for the best user experience.

4.2.4 Sound is not an essential feature but it can be considered for future variants of the system for user notifications.

4.3 Software Interfaces

- https://finnhub.io/ Is required to get the information of the stocks
- Any modern web browser used to access the site.

4.4 Communications Interfaces

- HTTP will be used to communicate between the user and the product.
- HTTP will also be used to make request and get responses from the API

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- Only textual information will be handled by the software. Amount of information to be handled can vary from user to user.
- Specified companies will be displayed as per the requirement.
- Model will take a certain amount of time for predicting the data.

5.2 Safety Requirements

Our product is advised to be used by users above 18 years as minors are not legal to do the transactions of the money and can potentially lose the transactions

5.3 Security Requirements

The product has a login feature to ensure the user credentials and portfolio are safe and the user is authorized only when the username and password credentials match. password is secured and encrypted.

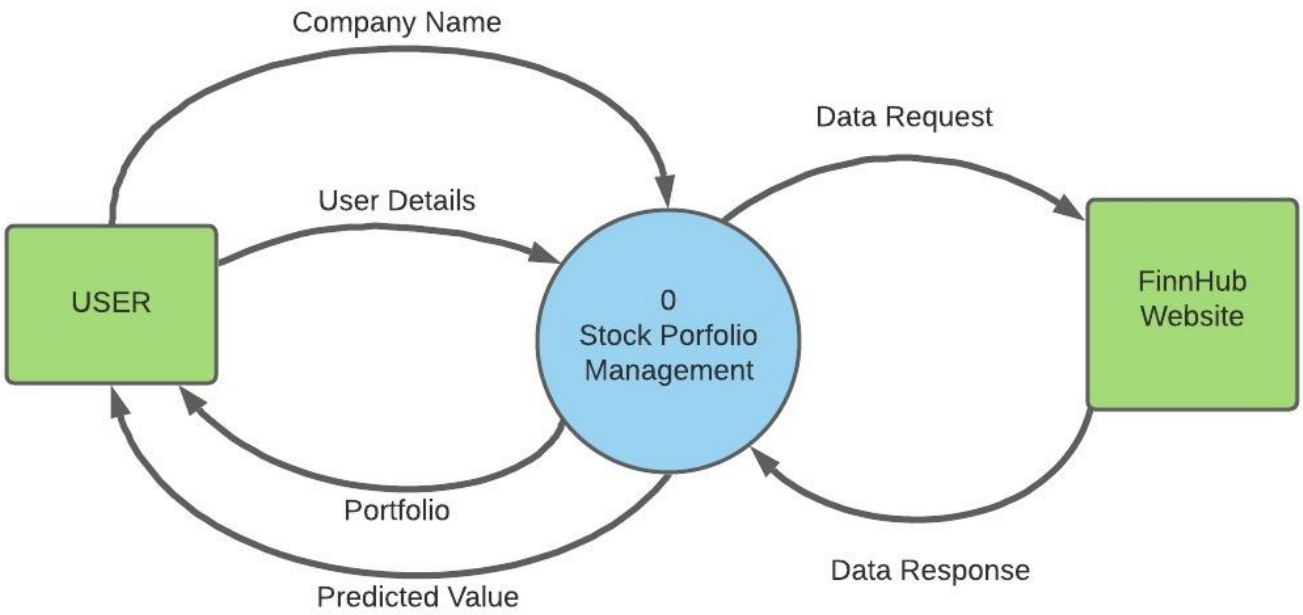
5.4 Software Quality Attributes

The product is targeted towards a wide variety of users with basic financial knowledge. The product must load quickly and work well on a variety of terminals. The product must be available to all the users without crashing by the traffic. Our product will work on all the browsers without any problems. It must also tolerate a wide variety of input possibilities from a user, such as incorrect responses or unforeseen keystrokes like incorrect logins and company details inputted by the user here. This product can be used by users with basic knowledge as predictions will be provided so it has an ease of use over ease of learning the stock market.

3. Structured Analysis

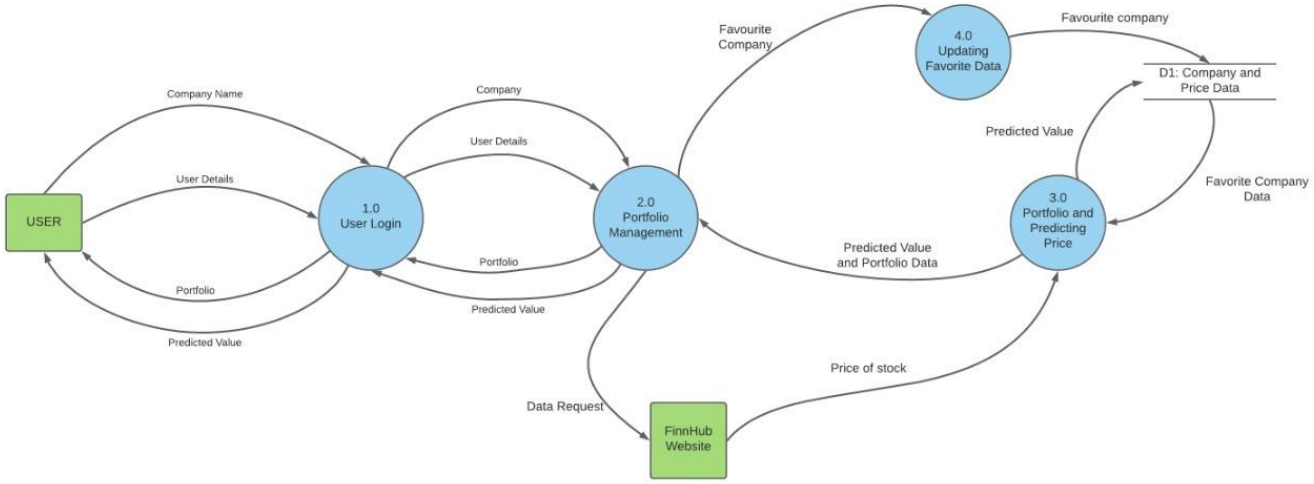
3.1 Data Flow Diagrams

3.1.1 DFD Level 0



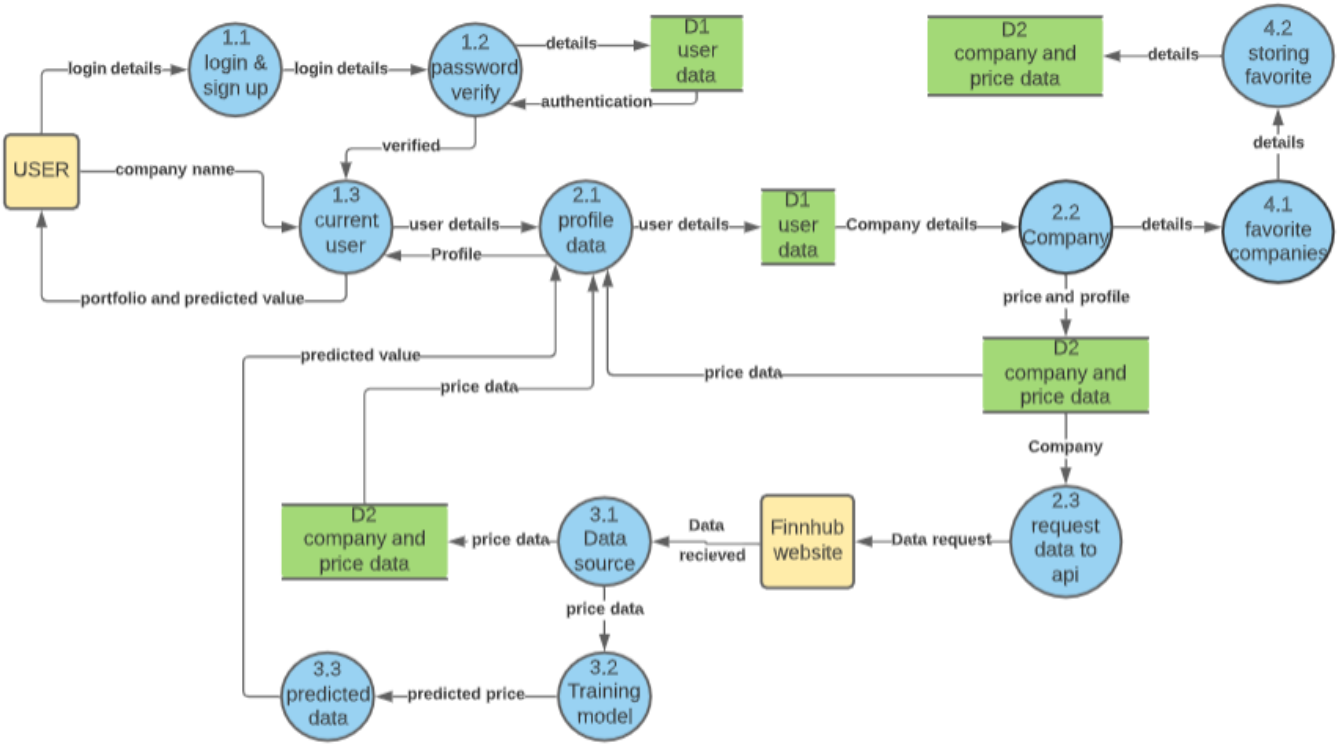
Level 0 Data Flow Diagram of
Stock Prediction System

3.1.2 DFD Level 1

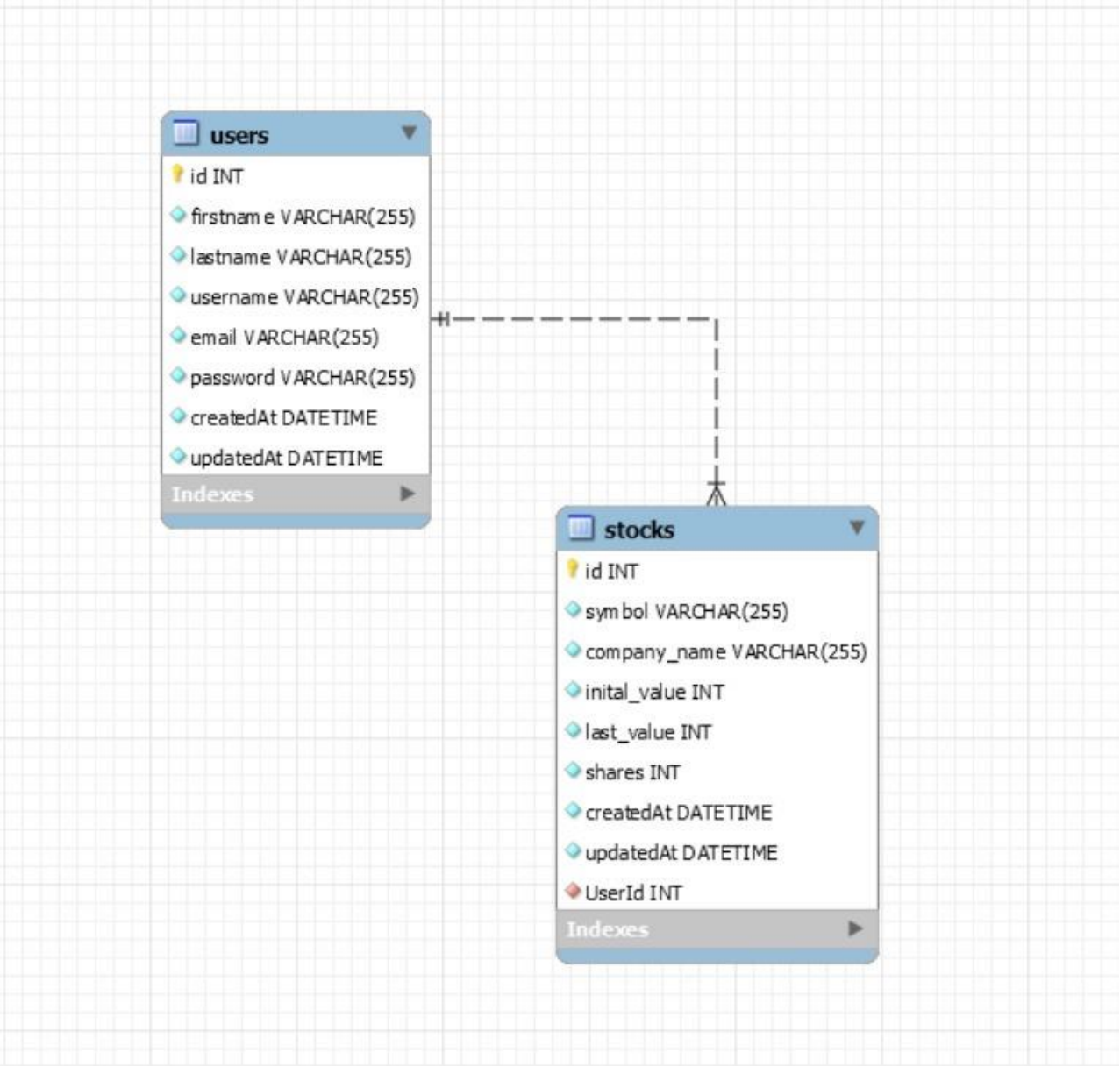


Level 1 Data Flow Diagram of
Stock Prediction System

3.1.3 DFD Level 2

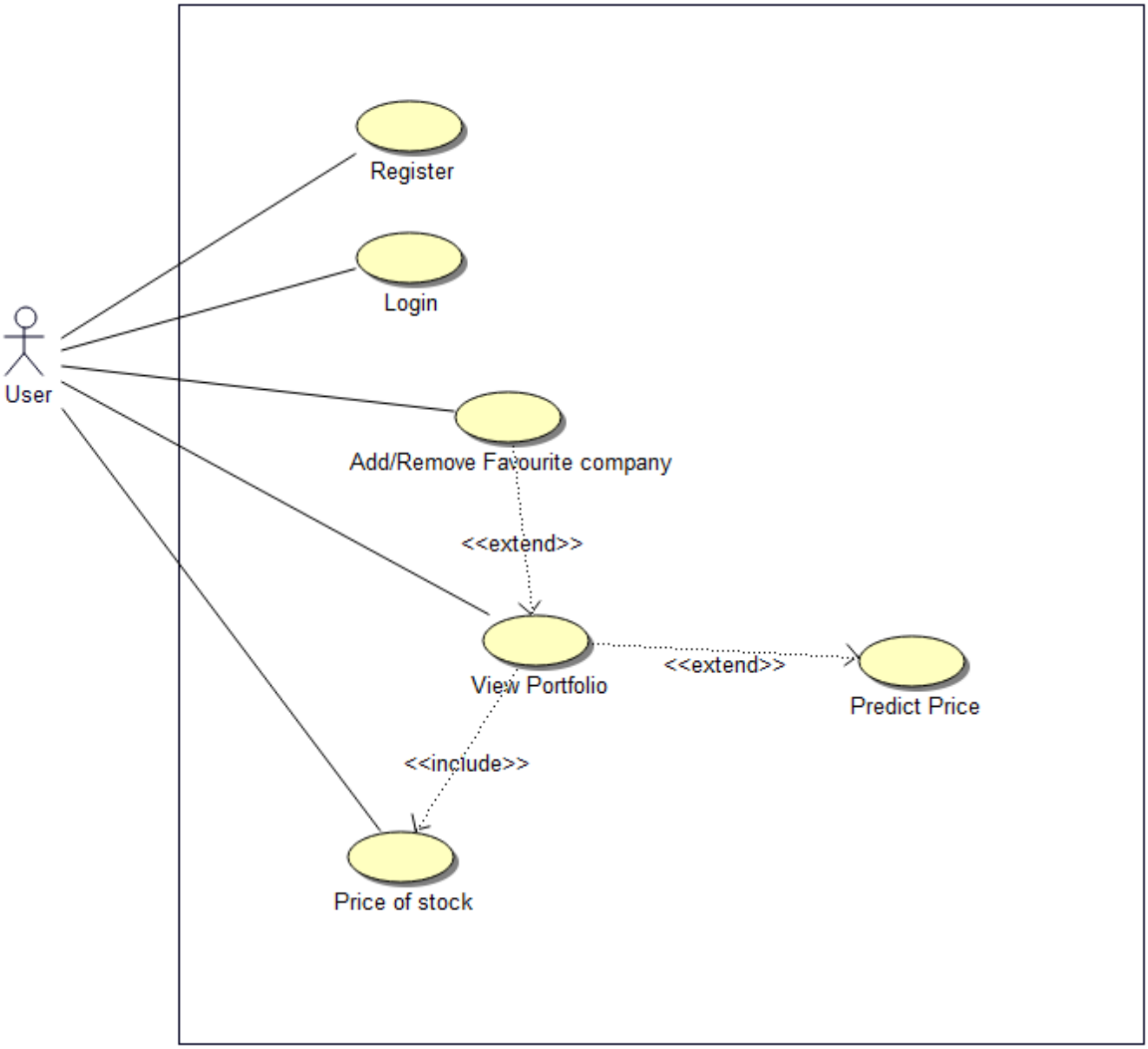


3.2 ER Diagram



4. Object Oriented Analysis

4.1 Use Case Diagram



4.2 Use Case Templates

Use Case Scenario

Use Case ID:	1
Use Case Name:	Register
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case allows user to Register in the system.
Primary Actor:	User
Secondary Actor:	None
Include use cases:	None
Extends use cases:	None
Preconditions:	User must have a valid Email address, name, username and password
Postconditions:	User will be able to access his portfolio
Task Sequence:	Details Asked Details Entered Email verified Duplicates checked User registered Portfolio displayed

Normal Flow of Events:

1. (SR) System asks the user to enter the details
2. (AA) User submits the details
3. (SR) System verifies the Email address
4. (SR) System checks for duplication.
5. (SA) System register the user.
6. (SA) System displays the Portfolio.

Alternative Flow of Events:

3a. (SR) System did not accept the invalid Email address

1. (SA) System prompts to enter valid email.
2. Use case resumes at main flow step 2.

4a. (SR) System did not accept the Duplicate entry

1. (SA) System prompts to enter valid details.
2. Use case resumes at main flow step 2.

Use Case ID:	2
Use Case Name:	Login
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case allows user to Login in the system
Primary Actor:	User
Secondary Actor:	None
Include use cases:	None
Extends use cases:	None
Preconditions:	User must have a valid account
Postconditions:	User will be able to access his portfolio
Task Sequence:	User Credentials Requested Credentials entered Credentials verified Portfolio Displayed

Use Case Scenario

Normal Flow of Events:

1. (SR) System asks the user to enter the username and password.
2. (AA) User submits the details
3. (SR) System verifies the details.
4. (SA) System displays the Portfolio.

Use Case ID:	3
Use Case Name:	View Portfolio
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case allows user to access portfolio
Primary Actor:	User
Secondary Actor:	None
Include use cases:	Price of stock
Extends use cases:	Add/remove fav company, Predict Price
Preconditions:	User must be logged in
Postconditions:	User will be able to view the portfolio
Task Sequence:	Favourite companies Retrieved Stock price triggered Favourite companies triggered Add/Remove favourite companies Predict Price

Use Case Scenario

Normal Flow of Events:

1. (SR) System retrieve the favorite companies of the user.
2. (SA) System Triggers Price of stock use case.
3. (SA) System displays the favorite companies along with prices.
4. (AA) Extension point: Add/Remove favorite companies.
5. (AA) Extension point: Predict Price.

Use Case ID:	4
Use Case Name:	Add/Remove Favorite Companies
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case allows user to add and remove companies to portfolio
Primary Actor:	User
Secondary Actor:	None
Include use cases:	None
Extends use cases:	None
Preconditions:	User must be logged in and Portfolio is already loaded
Postconditions:	User gets modified Portfolio
Task Sequence:	New Company entered Possible options displayed Option submitted Company added

Use Case Scenario

Normal Flow of Events:

1. (AA) User enters the new Company.
2. (SR) System displays the possible options.
3. (AA) User submits one of the options.
4. (SA) System adds the Company to the Portfolio.

Alternative Flow of Events:

1a. (AA) User enters a non-valid Company.

1. (SR) System doesn’t display any option.
2. (AA) User submits the non-valid input.
3. (SR) System prompts to enter the valid company.
4. Use case resumes from at main flow step 1.

Use Case ID:	5
Use Case Name:	Price of stock
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case gets the price of the stock
Primary Actor:	User
Secondary Actor:	None
Include use cases:	None
Extends use cases:	None
Preconditions:	User must be logged in and Portfolio is already loaded
Postconditions:	User gets the price of the stock
Task Sequence:	Favorite Companies Retrieved Prices fetched Companies displayed

Use Case Scenario

Normal Flow of Events:

1. (SA) System retrieve the user's Favorite companies
2. (SA) System fetch the prices of all those companies
3. (SA) System displays the favorite companies along with prices.

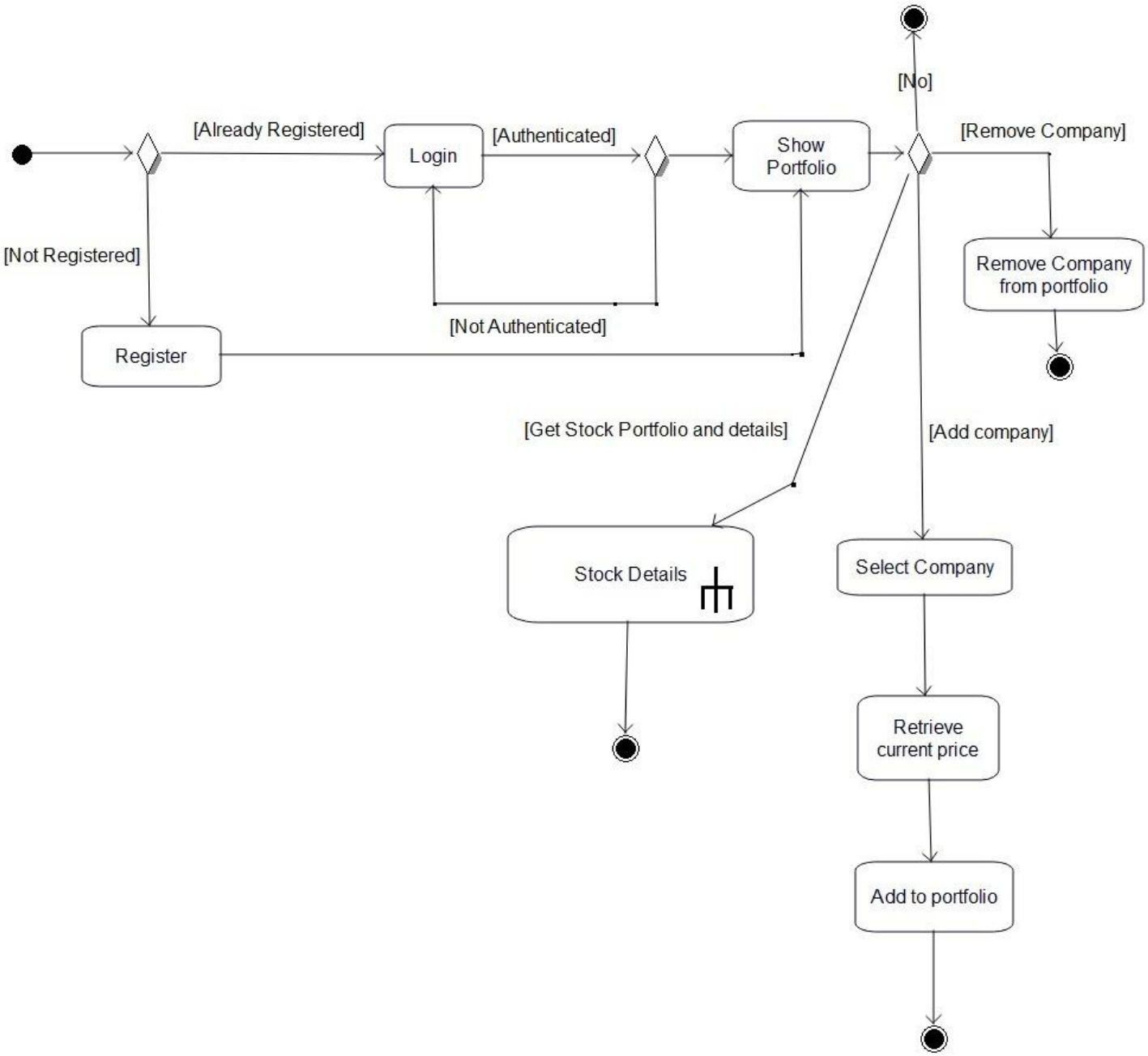
Use Case ID:	6
Use Case Name:	Predict Price
Created By:	Group_5
Date Created:	16-11-2020
Description:	This use case allows user to Predict Price
Primary Actor:	User
Secondary Actor:	None
Include use cases:	None
Extends use cases:	None
Preconditions:	User must be logged in and Portfolio is already loaded
Postconditions:	User will be able to get the predicted price
Task Sequence:	Stock selected Data retrieved Price predicted Prediction displayed

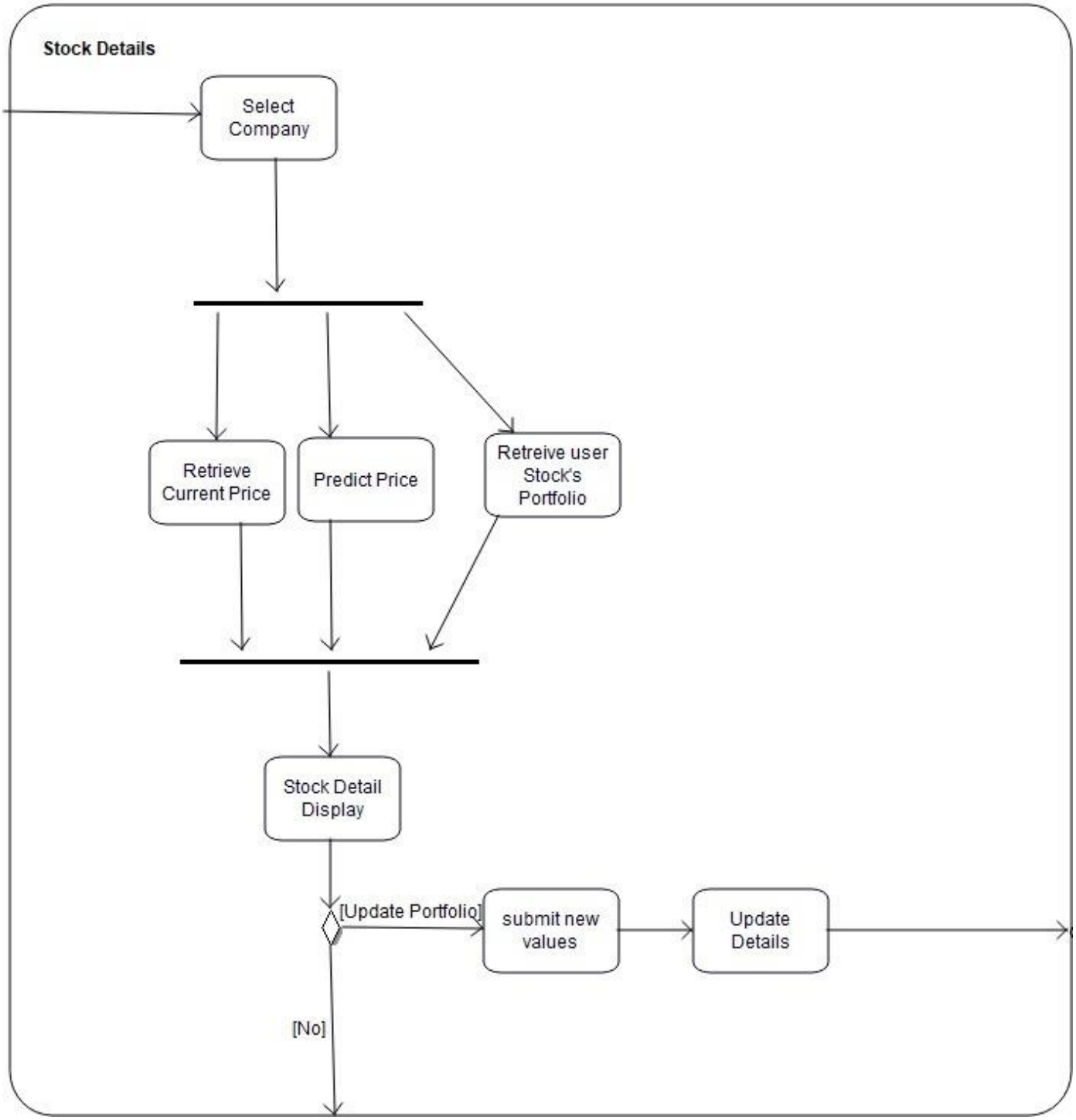
Use Case Scenario

Normal Flow of Events:

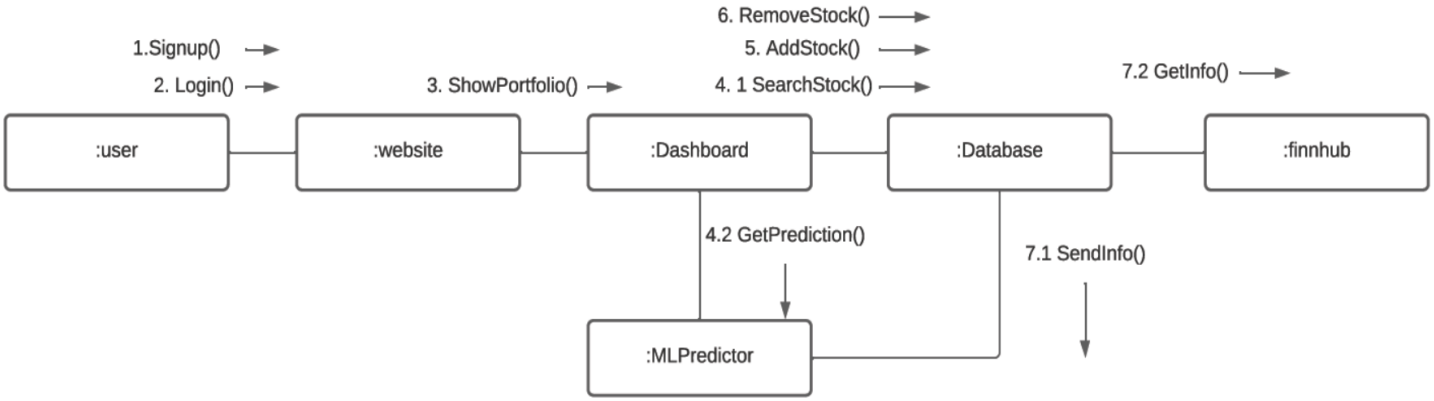
1. (AA) User selects the desired stock.
2. (SR) System retrieve the data required to predict the price.
3. (SA) System predicts the price.
4. (SR) System displays the predicted price to the user.

4.3 Activity Diagram

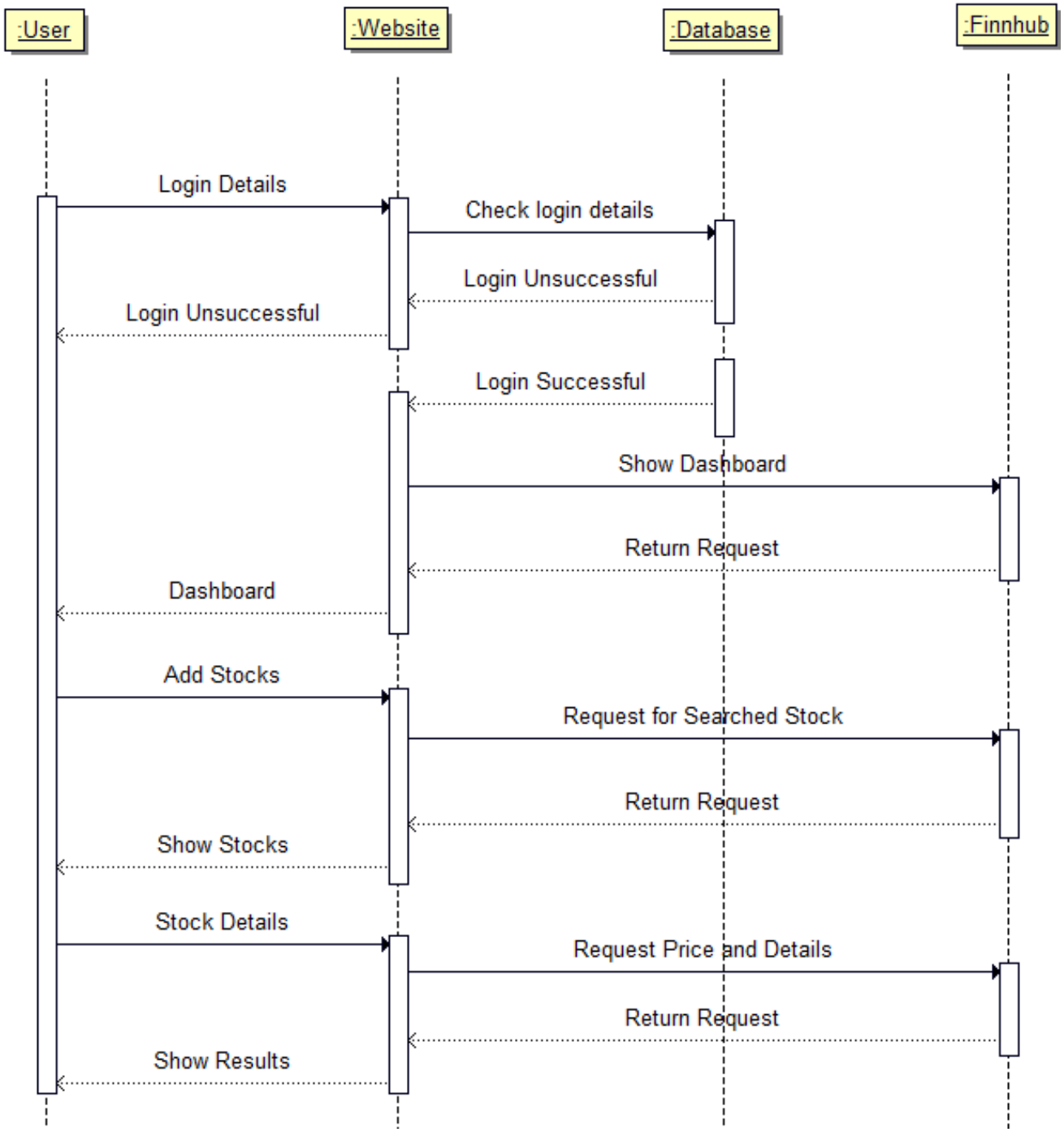



Stock Details
Select Company
Retrieve Current Price
Predict Price
Retreive user Stock's Portfolio
Stock Detail Display
[Update Portfolio]
submit new values
Update Details
[No]

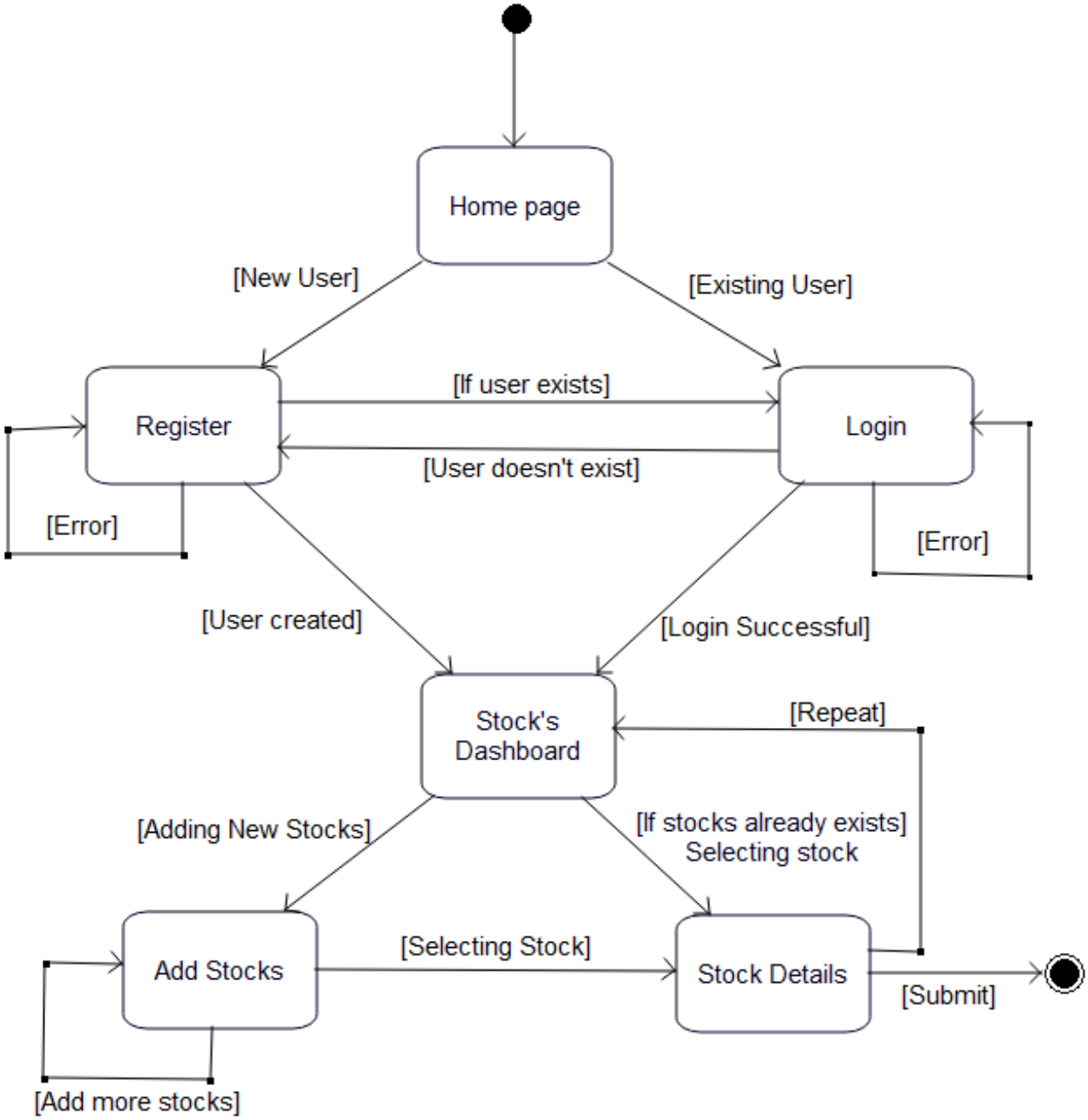
4.4 Collaboration Diagram



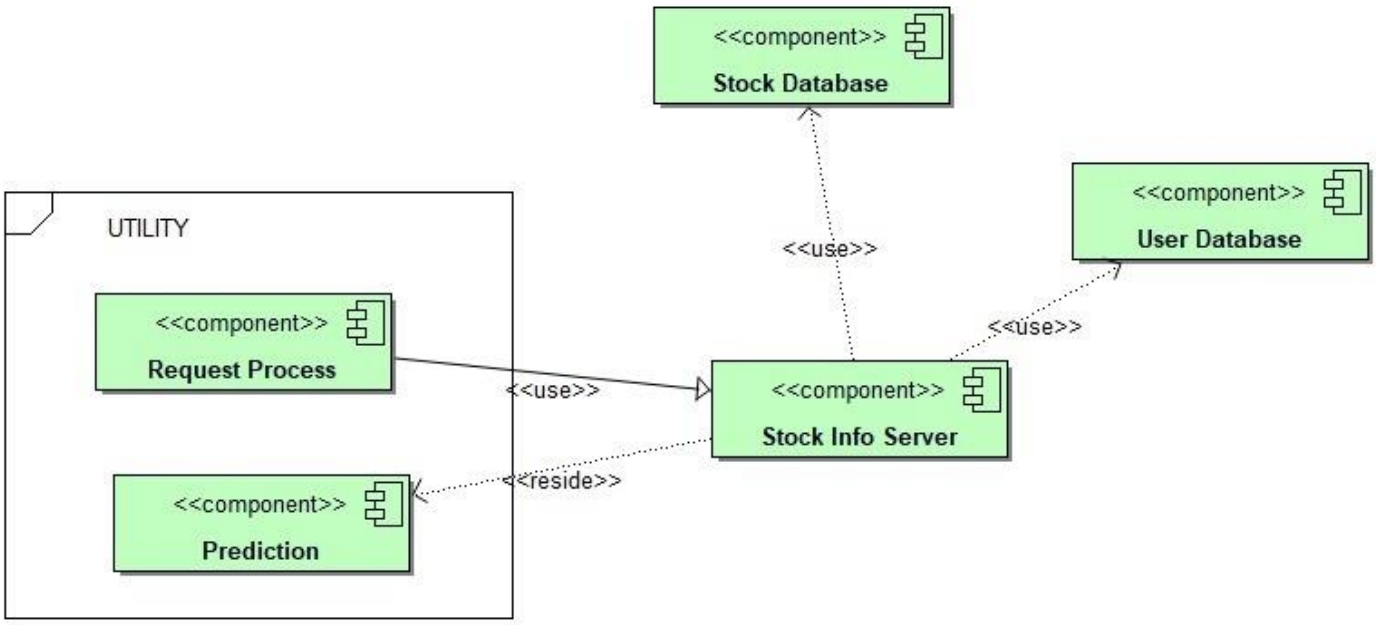
4.5 Sequence Diagram



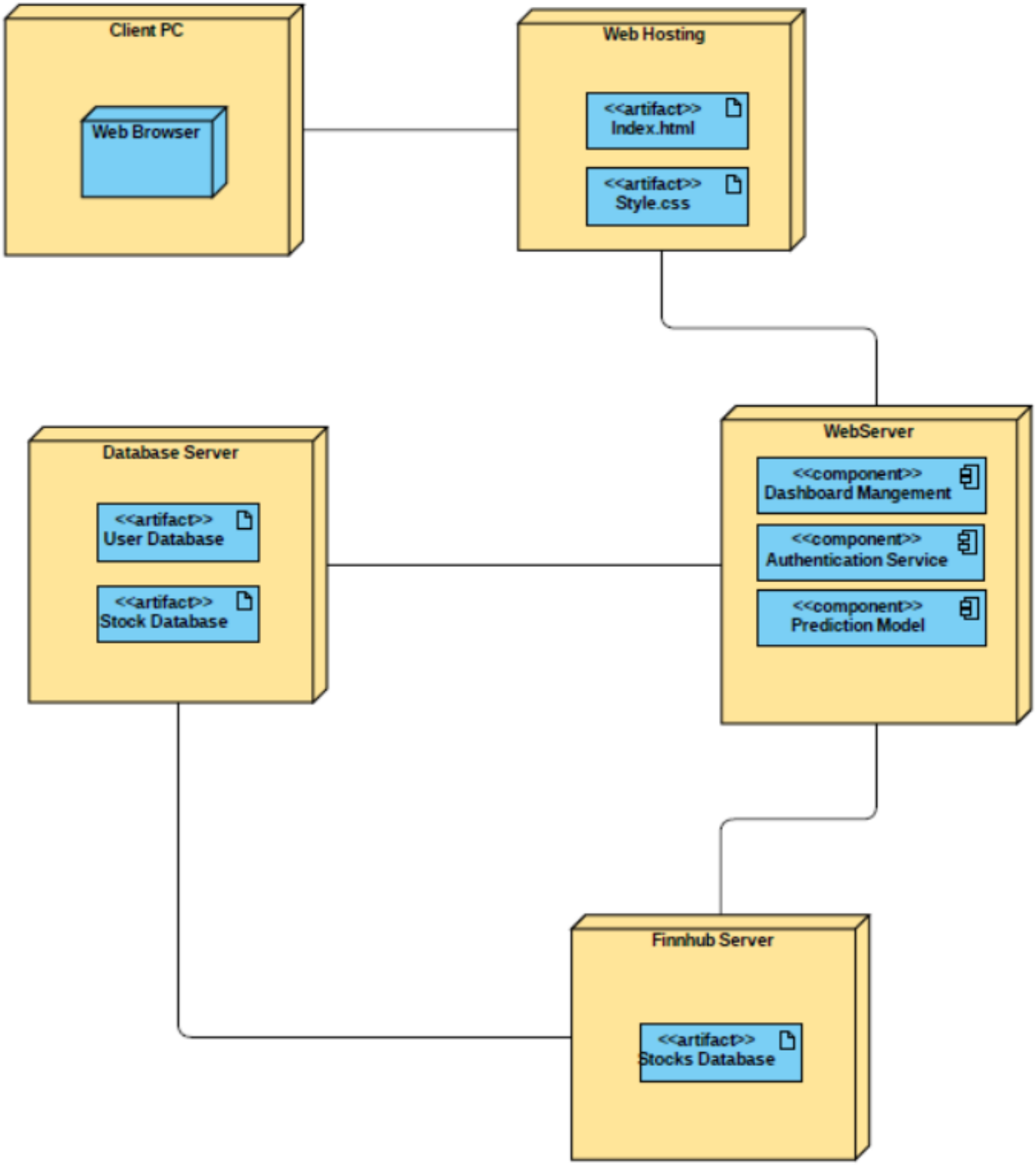
4.6 State Chart Diagram



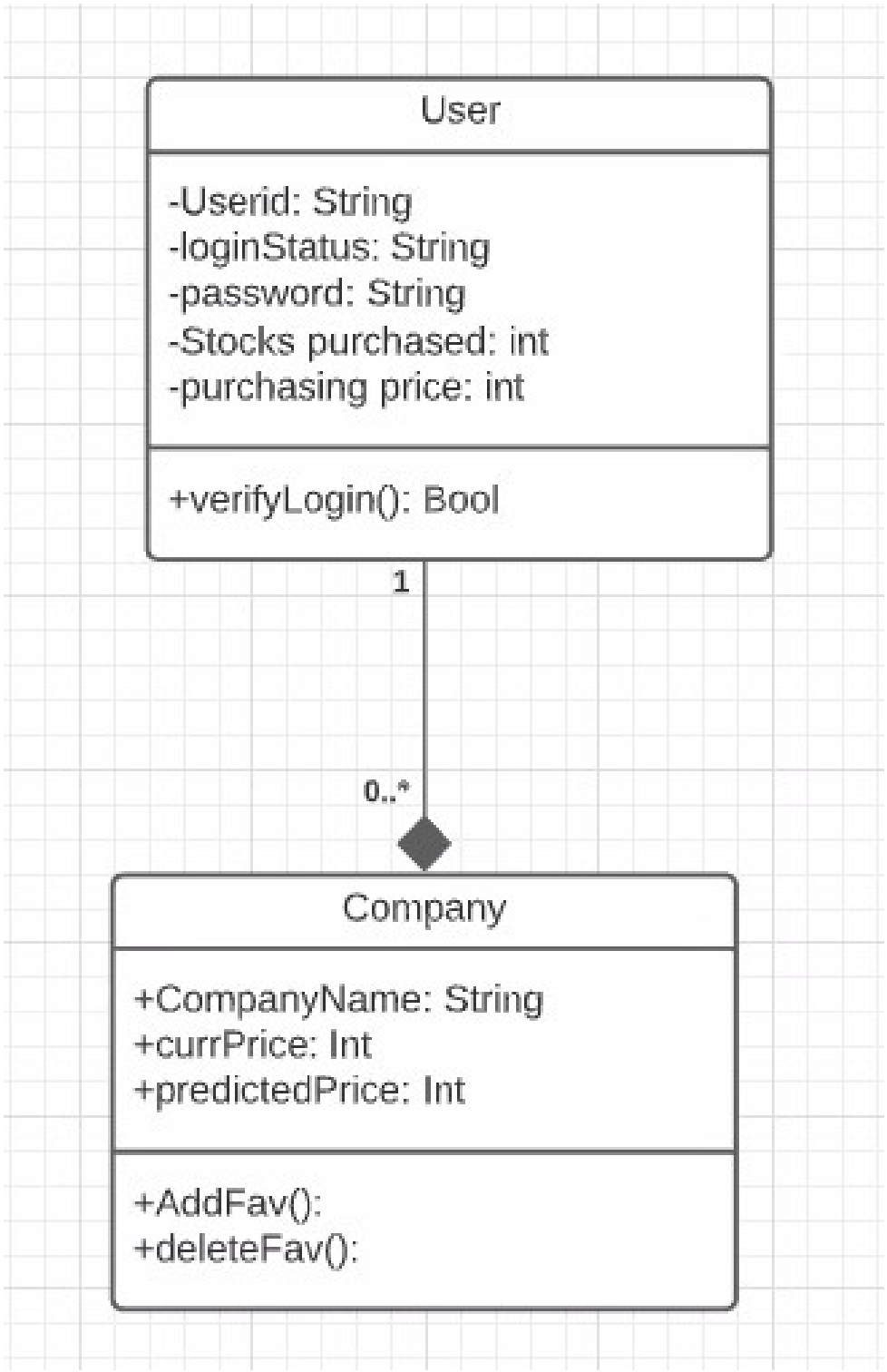
4.7 Component Diagram



4.8 Deployment Diagram



4.9 Class Diagram



5. Testing

5.1 TEST PLAN

Test cases were run on the Stock info website on a local host. If the website did not crash and it produced the desired result, the test case is considered to be passed. The testing involved the documentation of average time taken by the application to respond to an activity. Testing is also done on Non valid inputs to test the error handling capacity of website

Features to be tested (various functionalities)

- Login
- Register
- Adding the stock
- adding the valid stock quantity and prices
- prediction and graph display

Item Pass/Fail criteria

If the testing produces the desired result for every activity without any error, it is considered pass, otherwise fail.

5.2 TEST CASES:

Test Case #: 1 **Test Case Name:** Login
System: Stock Info Website **Subsystem:** Login
Designed by: Group 5 **Design Date:** 23/11/2020
Executed by: Group 5
Execution Date: 24/11/2020
Short Description: Tests the login functionality for the application

Pre-conditions
User has active internet connection
User is already registered and has a valid username and password

Step	Action	Expected System Response	Pass/Fail	Comment
1.	User enters valid username and correct password	User is authenticated and is logged in, showing his dashboard with his Portfolio.	Pass	
3.	**Check Post Condition 1**			
2.	User enters invalid username but incorrect	System displays “Wrong Credentials” message and asks the user to enter the again	Pass	
4.	User enters valid username and correct password	User is authenticated and is logged in, showing his dashboard with his Portfolio.	Pass	
5.	**Check Post Condition 2**			

Post-conditions

1. User is logged into his Portfolio Page.

2.User is logged into his Portfolio Page.

Test Case #: 2 **Test Case Name:** Register **System**: Stock Info Website
Subsystem: Register
Designed by: Group 5 **Design Date:** 23/11/2020
Executed by: Group 5
Execution Date: 24/11/2020
Short Description: Tests the register functionality for the application

Pre-conditions
User has an active internet connection.
User must have a valid Email address.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	User enters valid username(unique), E-mail(unique) and Other details	User is registered and is logged in, showing his dashboard with his Portfolio.	Pass	
2.	**Check Post Condition 1**			
3.	User enters an already existing username.	System displays "username not available" use another user name	Pass	
4.	**Check Post Condition 2**			
5.	User enters invalid Email address.	System prompts the user to enter valid Email address		
6.	**Check Post Condition 3**			
7.	User enters an already registered Email address.	System alert the user that it is already registered.		
9.	**Check Post Condition 4**			

Post-conditions

1. User is logged into his Portfolio Page
2. User is able to enter the details
3. User is able to enter the details
4. User is able to enter the details.

Test Case #: 3 **Test Case Name:** Adding the stock
System: Stock Info Website **Subsystem:** Stock Addition
Designed by: Group 5 **Design Date:** 23/11/2020
Executed by: Group 5
Execution Date: 24/11/2020
Short Description: Tests the stock validation functionality for the application

Pre-conditions
Users have an active internet connection.
User must be logged in.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	User enters valid Stock name	System displays the code of the company	Pass	
2.	User clicks the add button	System adds the stock and displays the success message.	Pass	
3.	**Check Post Condition 1**			
4.	User enters invalid Stock name	System doesn't displays the code of the company	Pass	
5.	User clicks the add button	System doesn't show any activity and waits for user to enter the valid name.	Pass	
6.	**Check Post Condition 2**			

Post-conditions

1. User successfully adds the stock in his portfolio
2. User is able to enter the valid company name

Test Case #: 4 **Test Case Name:** Value Prediction
System: Stock Info Website **Subsystem:** Predictor
Designed by: Group 5 **Design Date:** 23/11/2020
Executed by: Group 5
Execution Date: 24/11/2020
Short Description: Tests the stock prediction functionality of the application.

Pre-conditions
Users have an active internet connection.
Users must be logged in.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	User clicks on the More info button of the particular stock.	Graph of the stock is displayed along with the predicted price of the stock (with some time delay).	Pass	
2.	**Check Post Condition 1**			

Post-conditions

1.User is able to view the predicted price and the graph of the particular stock.

Test Case #: 5
Test Case Name: Enter valid details of purchased stocks
System: Stock Info Website **Subsystem:** Portfolio Update
Designed by: Group 5 **Design Date:** 23/11/2020
Executed by: Group 5
Execution Date: 24/11/2020
Short Description: Tests the valid input functionality of the application

Pre-conditions
User has an active internet connection.
User must be logged in.
User must be on more info page of the particular stock.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	User enters numeric data in the field.	System updates the details of the Portfolio of the particular stock and displays the info.	Pass	
2.	**Check Post Condition 1**			
3.	User enters invalid values	System doesn't update the details and wait for the user to enter valid details	Pass	
4.	**Check Post Condition 2**			

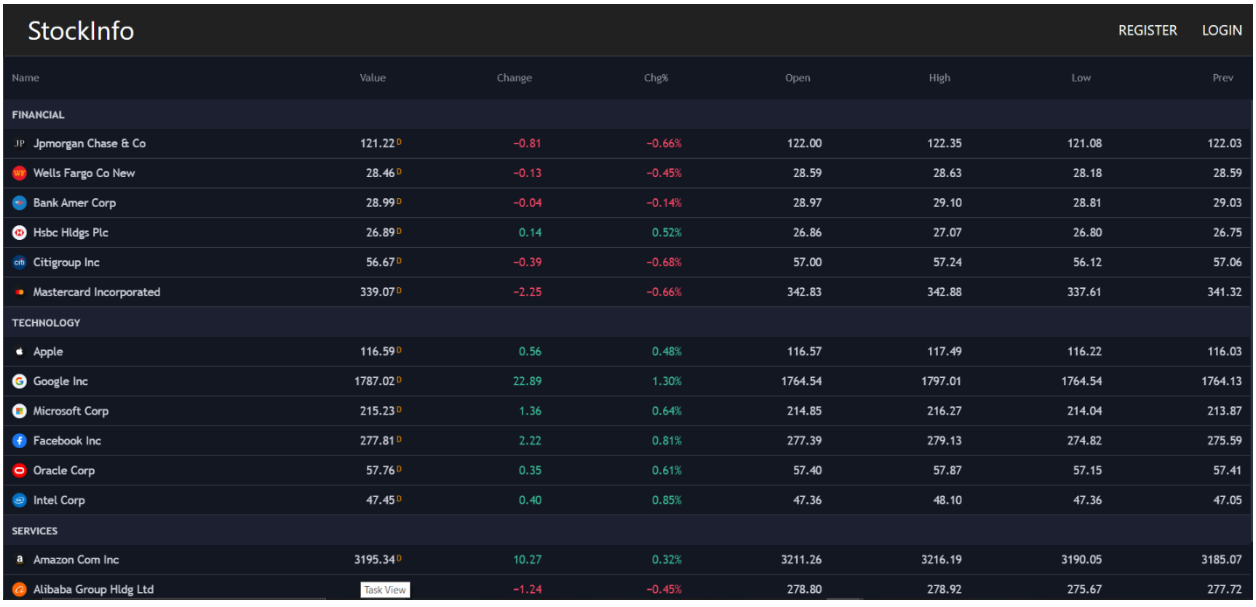
Post-conditions

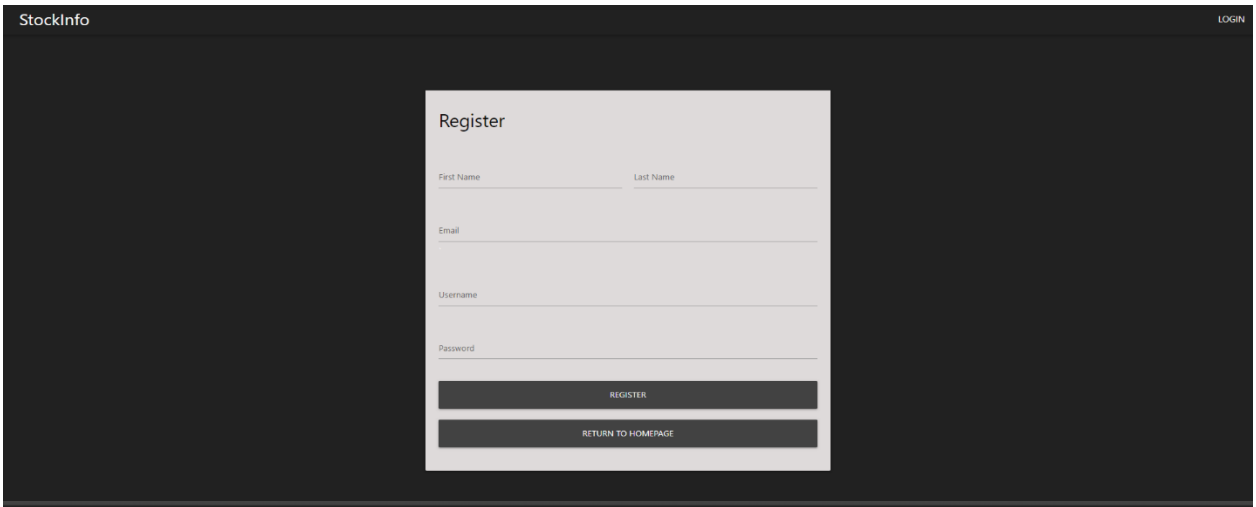
1. User successfully updates the portfolio
2. User is able to update his portfolio.

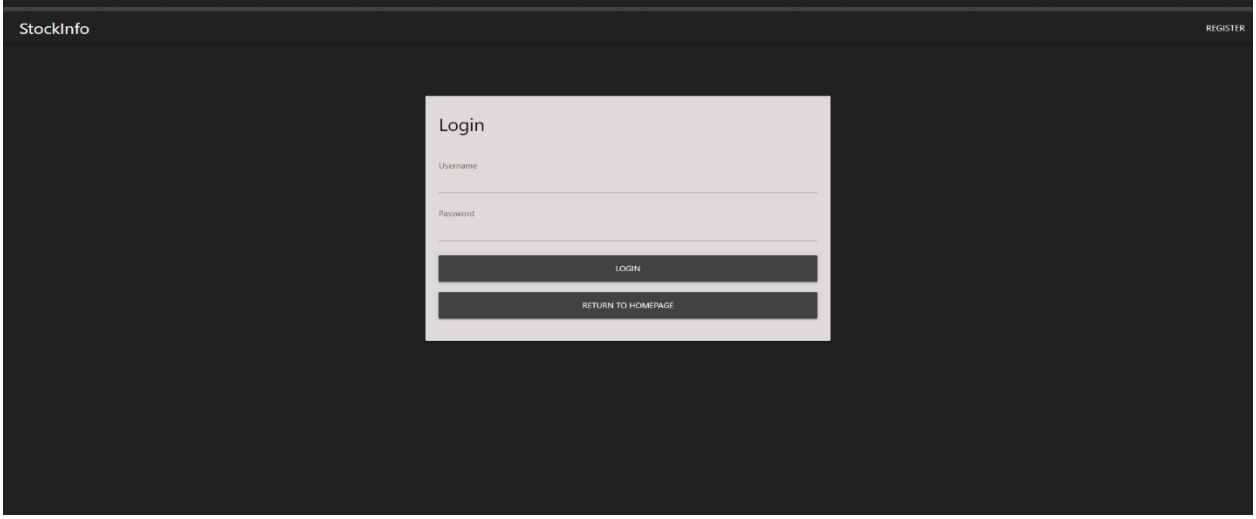
TEST CASE REPORT

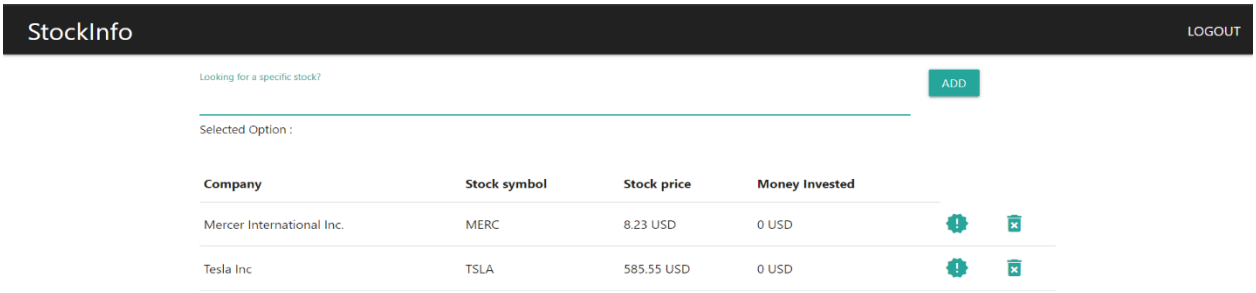
All The above test cases were passed on the website. There were no bugs. All the requirements were met and the website was running as required.

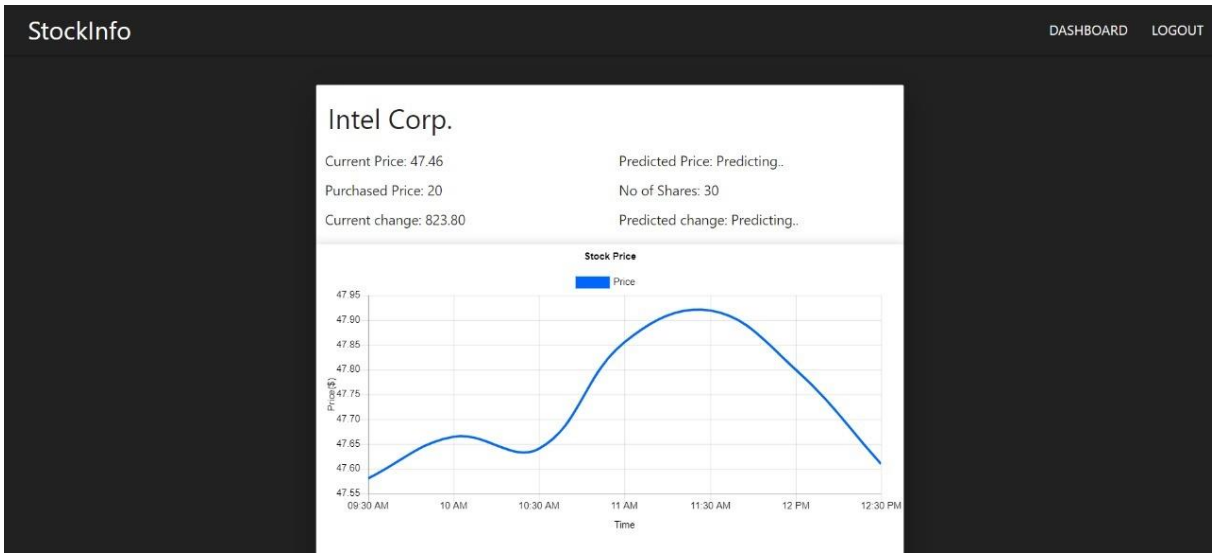
5.3 Screenshots:








StockInfo
LOGOUT
Looking for a specific stock?
ADD
Selected Option :
Company
Stock symbol
Stock price
Money Invested
Mercer International Inc.
MERC
8.23 USD
0 USD
Tesla Inc
TSLA
585.55 USD
0 USD


StockInfo
DASHBOARD
LOGOUT
Intel Corp.
Current Price: 47.46
Predicted Price: Predicting..
Purchased Price: 20
No of Shares: 30
Current change: 823.80
Predicted change: Predicting..
Stock Price
Price
47.95
47.90
47.85
47.80
47.75
47.70
47.65
47.60
47.55
Price($)
09:30 AM
10 AM
10:30 AM
11 AM
11:30 AM
12 PM
12:30 PM
Time