



# MANIPAL UNIVERSITY JAIPUR

*A Project Report*

*on*

## **Stock Management and Analysis System**

*carried out as part of the course CS-1532 Submitted by*

***Kavleen Sabharwal***

***169105093***

***Vth Sem, B.Tech CSE***

*and*

***Maheeka Kaistha***

***169105101***

***Vth Sem, B.Tech CSE***

*and*

***Mehil B Shah***

***169105105***

***Vth Sem, B.Tech CSE***

*and*

***Mehul Jhaver***

***169105106***

***Vth Sem, B.Tech CSE***

*in partial fulfilment for the award of the degree*

*of*

**BACHELOR OF TECHNOLOGY**

*in*

**Computer Science & Engineering**  
**Department of Computer Science & Engineering,**  
**School of Computing and IT,**  
**Manipal University Jaipur,**  
*November, 2018*

This is to certify that the project entitled "**Stock Management and Analysis System**" is a bonafide work carried out as part of the course **CS-1553**, under my guidance by **TEAM 14**, team of B.Tech V<sup>th</sup> Semester, Section B at the Department of Computer & Communication Engineering , Manipal University Jaipur, during the academic semester 5, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering, at MUJ, Jaipur.

Place: Manipal University, Jaipur.

Date: Signature of the Instructor (s)

## **DECLARATION**

We hereby declare that the project entitled **Stock Management and Analysis System** submitted as part of the partial course requirements for the course **Software Engineering** for the award of the degree of Bachelor of Technology in Computer Science & Engineering at Manipal University Jaipur during 5th Semester, August 2018 - December 2018, has been carried out by us. I declare that the project has not formed the basis for the award of any degree, associate ship, fellowship or any other similar titles elsewhere.

Further, I declare that I will not share, re-submit or publish the code, idea, framework and/or any publication that may arise out of this work for academic or profit purposes without obtaining the prior written consent of the Course Faculty Mentor and Course Instructor.

Signature of the Student:

Place: Manipal University, Jaipur.

Date: 23<sup>rd</sup> November, 2018.

# ABSTRACT

This project aims to create a stock management and analysis system, from which various users can buy or sell stocks according to their convenience and analyze stocks for convenience.

Application is divided into two parts:

- ❖ Management
- ❖ Analysis

## **Management**

In this module, we aim to facilitate the user with the stock buying and selling facility. User will have to provide his bank details and using that bank details, we can fetch the balance in his/her account. Then, Demat Account number will be generated which will serve as User ID for the client. After that, user can buy/sell stocks according to his/her current balance and current status of a particular stocks.

## **Analysis**

In this module, we aim to analyze various stocks. There are two types of analysis included in this project: Single Stock Analysis and Comparative Stock Analysis. In Single Stock Analysis, we take a name of single stock, start date and end date as inputs. Based on that, we give user a choice from 4 parameters, and when user clicks on the parameter, stock is analyzed on that particular parameter. Similar module has been developed for Comparative Stock Analysis, where 2 stock names are taken as inputs and stocks are compared based on the particular parameters.

# TABLE OF CONTENTS

S.no	Contents	Page No.
1.	Certificate and Declaration	1
2.	Abstract, Introduction 1. Purpose 2. Scope of Work	3
3.	Requirements Analysis 1. Functional Requirements 2. Black Box Test Cases 3. White Box Test Cases 4. Non-Functional Requirements 5. Project Diagrams	6
4.	System Design 1. Design Goals 2. UI/UX Design 3. System Architecture	15
5.	Work Done 1. Development Environment 2. Individual Contribution of Team Members	18
6.	Conclusion 1. Future Scope 2. Task board	19
7.	Testing Tool Report 1. App Performance Review using LoadRunner	21



## **1. INTRODUCTION**

### **1.1. SCOPE OF WORK**

Following are the scope of work for this project:

#### **Deliverables**

1. Use Case Diagram
2. Data Flow Diagram
3. Sequence Diagram
4. Class Diagram
5. ER Diagram
6. Software Requirements Specification
7. UI Design
8. Complete UI
9. Complete working Desktop App

#### **Timeline**

1. Requirement Analysis: 2 weeks
2. UI/UX Design: 4 weeks
3. System Design: 2 weeks.
4. Coding: 4 weeks
5. Testing: 2 weeks

### **1.2. PRODUCT SCENARIOS**

- Stock analysis is the evaluation of a particular trading instrument, an investment sector, or the market as a whole. Stock analysts attempt to determine the future activity of an instrument, sector, or market.
- Technical analysts analyze the financial market as a whole and are primarily concerned with price and volume, as well as the demand and supply factors that move the market.
- The user will use our application by first logging/signing in and then choose the kind of analysis he/she wants to conduct, after which the user will be able to analyze the stock and make decisions.
- This application also allows user to manage stock, that is, buy new stocks or sell existing stocks.

## 2. Requirements Analysis

### 2.1 Functional Requirements

- The Client Name should only be Alphabets and Spaces. PAN Number should be of the form AAAAA9999A, where A specifies a character and 9 specifies a number.
- Any other structure shall be rejected. Bank Account Details should be in a prescribed format.
- Client ID and Demat Account Number should be generated accordingly.
- Client ID and Account Balance needs to be checked before processing. Value of Stock needs to be calculated and the amount needs to be deducted in case of buy and added, in case of sell.
- Administrator can view transactions, and generate the data for the firm's use. This data needs to be tallied with the user on phone-calls.
- List of stocks to be provided to the user.
- Ensuring the dates entered don't violate the conditions: End-Date should not occur after Current Date.
- Writing the back-end code for all Technical Indicators, by studying them thoroughly.
- Plotting various graphs according to user-defined requirements.

### Black-Box Testing

#### Test Cases:

Test Case ID	Email-ID	Password	Contact Number	Card Type	Card Number	Expected Output	Actual Output	Pass/Fail
1	m	123456789	9429357646	Debit	123456789101121	Invalid Email	Invalid Email	Pass
2	shahmehil6@gmail.com	123456	9429357646	Debit	123456789101121	Invalid Password	Invalid Password	Pass
3	shahmehil6@gmail.com	123456789	94293576	Debit	123456789101121	Invalid Contact	Invalid Contact	Pass
4	shahmehil6@gmail.com	123456789	9429357646	ABCD	123456789101121	Invalid Card Type	Invalid Card Type	Pass



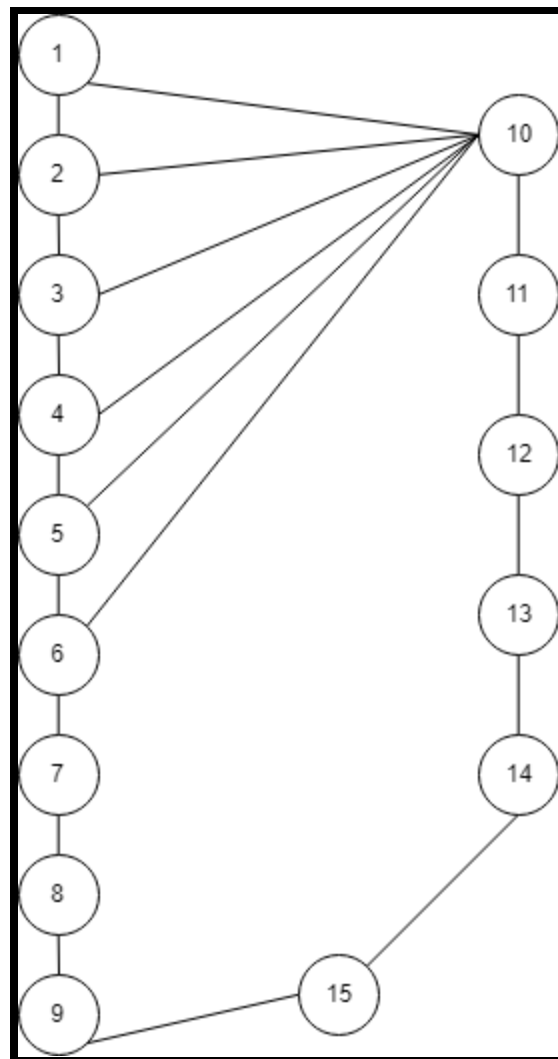
5	shahmehil6@gmail.com	123456789	9429357646	Debit	123456789	Invalid Card Number	Invalid Card Number	Pass
6	shahmehil6@gmail.com	123456789	ABCD	Debit	123456789101121	Invalid Contact	Invalid Contact	Pass
7	shahmehil6@gmail.com	123456	94293576	Debit	123456789101121	Invalid Password and Contact	Invalid Password and Contact	Pass
8	shahmehil6@gmail.com	123456789	9429357646	Debit	123456789101121	Valid	Valid	Pass

## White-Box Testing

### Test Cases:

- 1) if("@" in email and "." in email): emailf = 1
- 2) if(len(password) >= 8 and len(password) <= 12): passwordf = 1
- 3) if(len(number) == 10 and number.isnumeric()): numberf = 1
- 4) if(ctype == 'Debit' or ctype == 'Credit' or ctype == 'credit' or ctype == 'debit' or ctype == 'DEBIT' or ctype == 'CREDIT'): ctypef = 1
- 5) if(len(cnumber) == 16): cnumberf = 1
- 6) if(emailf == 1 and passwordf == 1 and numberf == 1 and ctypef == 1 and cnumberf == 1):
- 7) conn = sqlite3.connect('Form.db')
- 8) with conn:
- 9) cursor = conn.cursor()
- 10) cursor.execute('CREATE TABLE IF NOT EXISTS UserDB(fullname TEXT, email TEXT, password TEXT, number TEXT, bandb TEXT, ctype TEXT, cnumber TEXT, nameoncard TEXT)')
- cursor.execute('INSERT INTO  
UserDB(fullname,email,password,number,bandb,ctype,cnumber,nameoncard)  
VALUES(?,?,?,?,?,?,?,?),(fullname,email,password,number,bandb,ctype,cnumber,nameoncard)')
- conn.commit()
- else:
- 11) if(emailf == 0): tkinter.messagebox.showinfo('Recheck', 'Invalid Email')
- 12) if(passwordf == 0): tkinter.messagebox.showinfo('Recheck', 'Invalid Password')
- 13) if(numberf == 0): tkinter.messagebox.showinfo('Recheck', 'Invalid Contact Number')

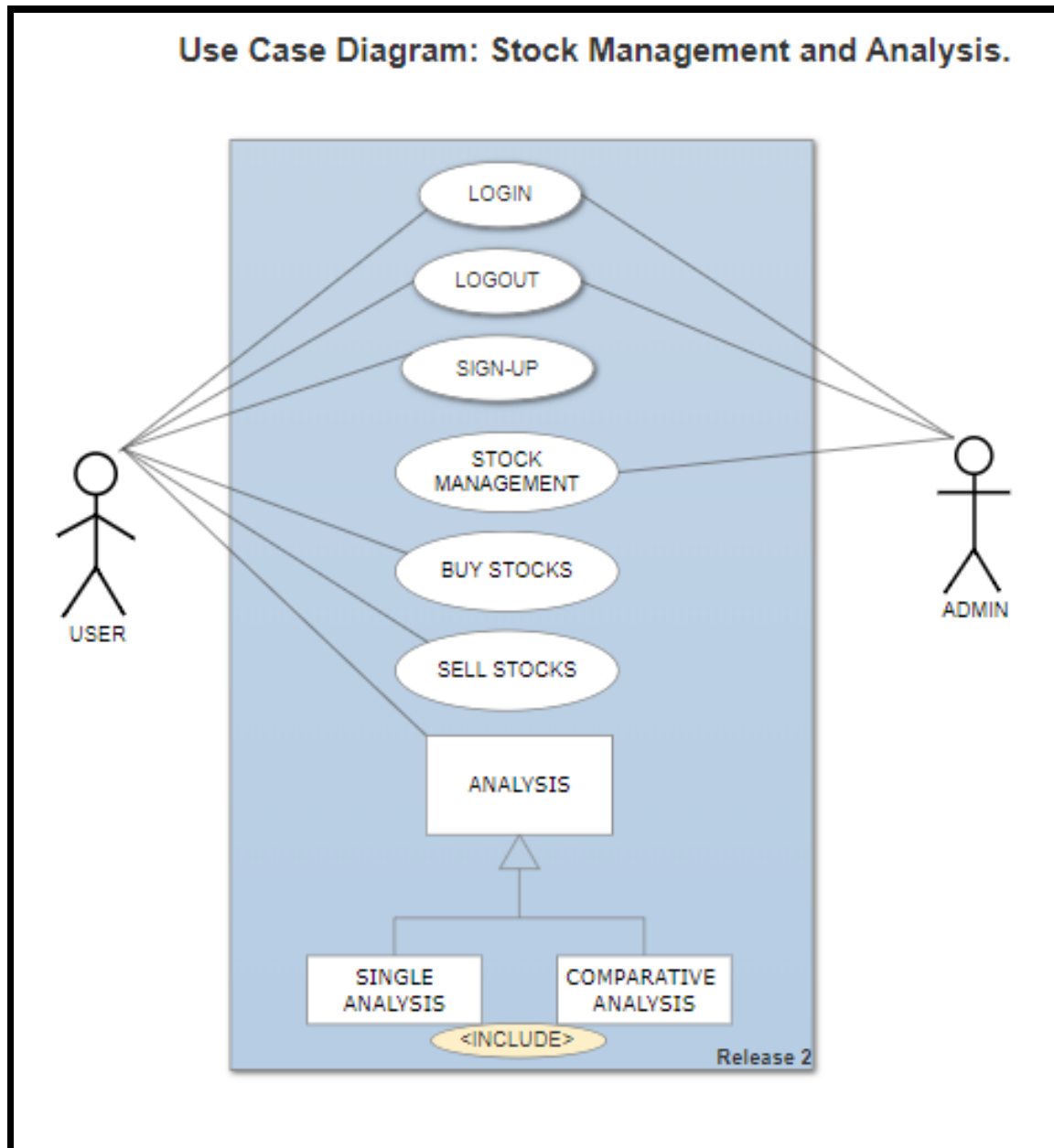
14) if(ctypef==0): tkinter.messagebox.showinfo('Recheck','Invalid Card Type')  
 15) if(cnumberf== 0): tkinter.messagebox.showinfo('Recheck','Invalid Card Number')

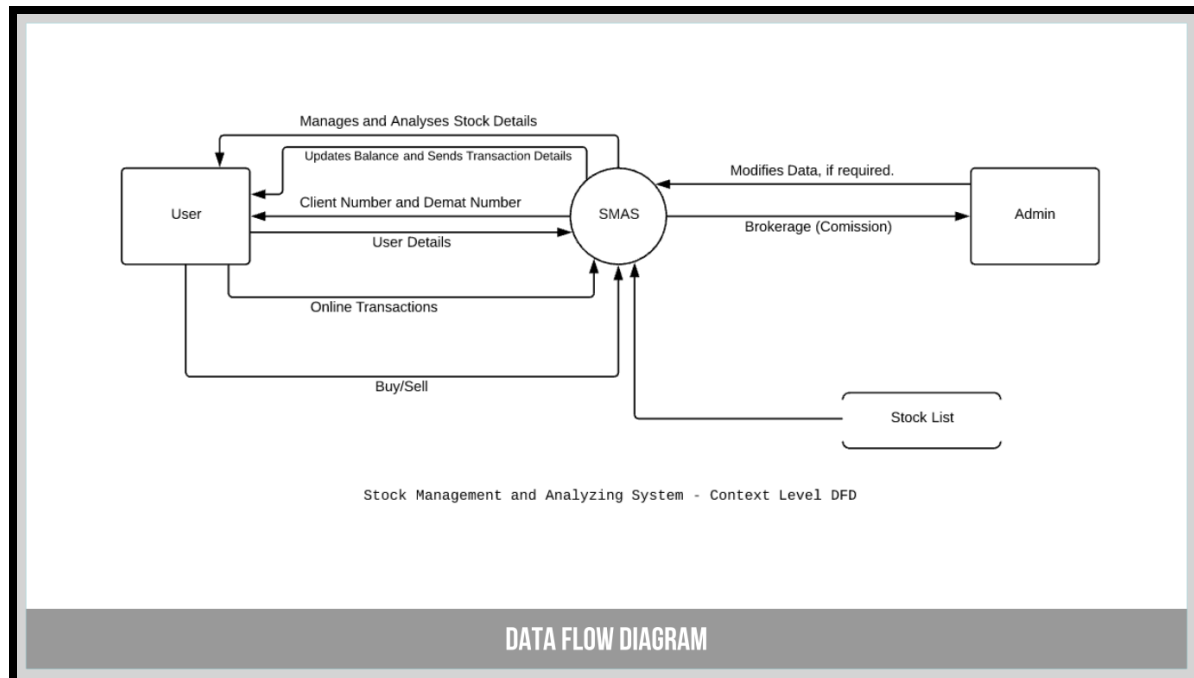


## 2.2 Non-Functional Requirements

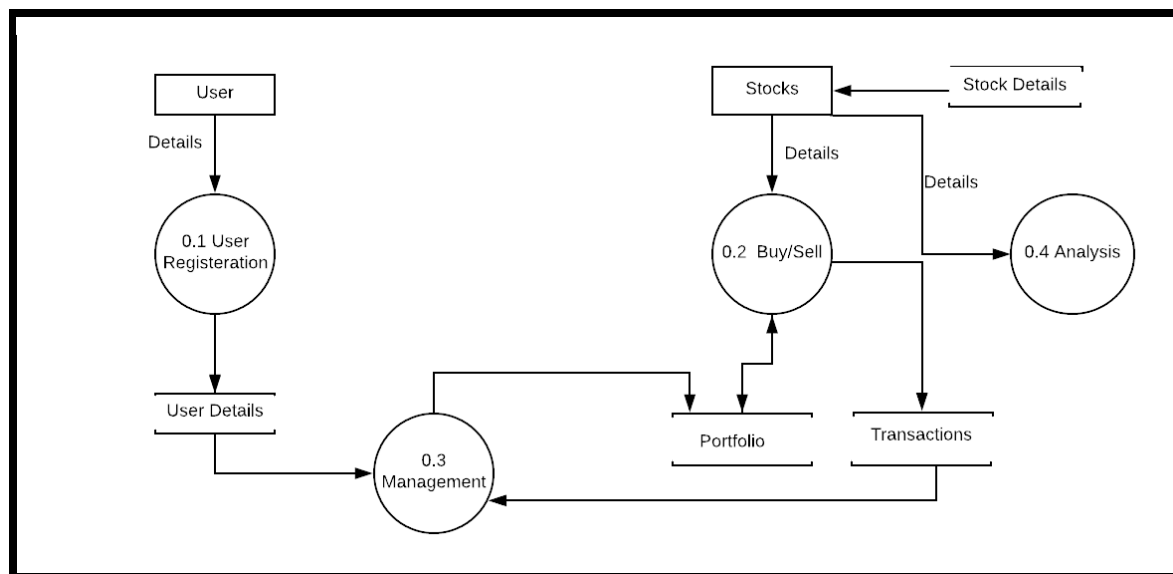
- **Security Requirements:** In this system, sensitive details of the clients are stored. Thus, Information needs to be protected from intrusion and unauthorized modifications. For this, the 3 databases are made locked and can only be edited by the administrator. Also, authorization of login details is a must, and to view Client Details, Client ID is required.
- **Safety Requirements:** The idea of transaction processing is required. The transaction made must reflect in the databases. There should not be any data loss during a transaction. Data Integrity must be preserved.
- **Software Quality Attributes:** The software is reusable and can be extended to other stock markets.

## 2.3 Use Case Scenarios

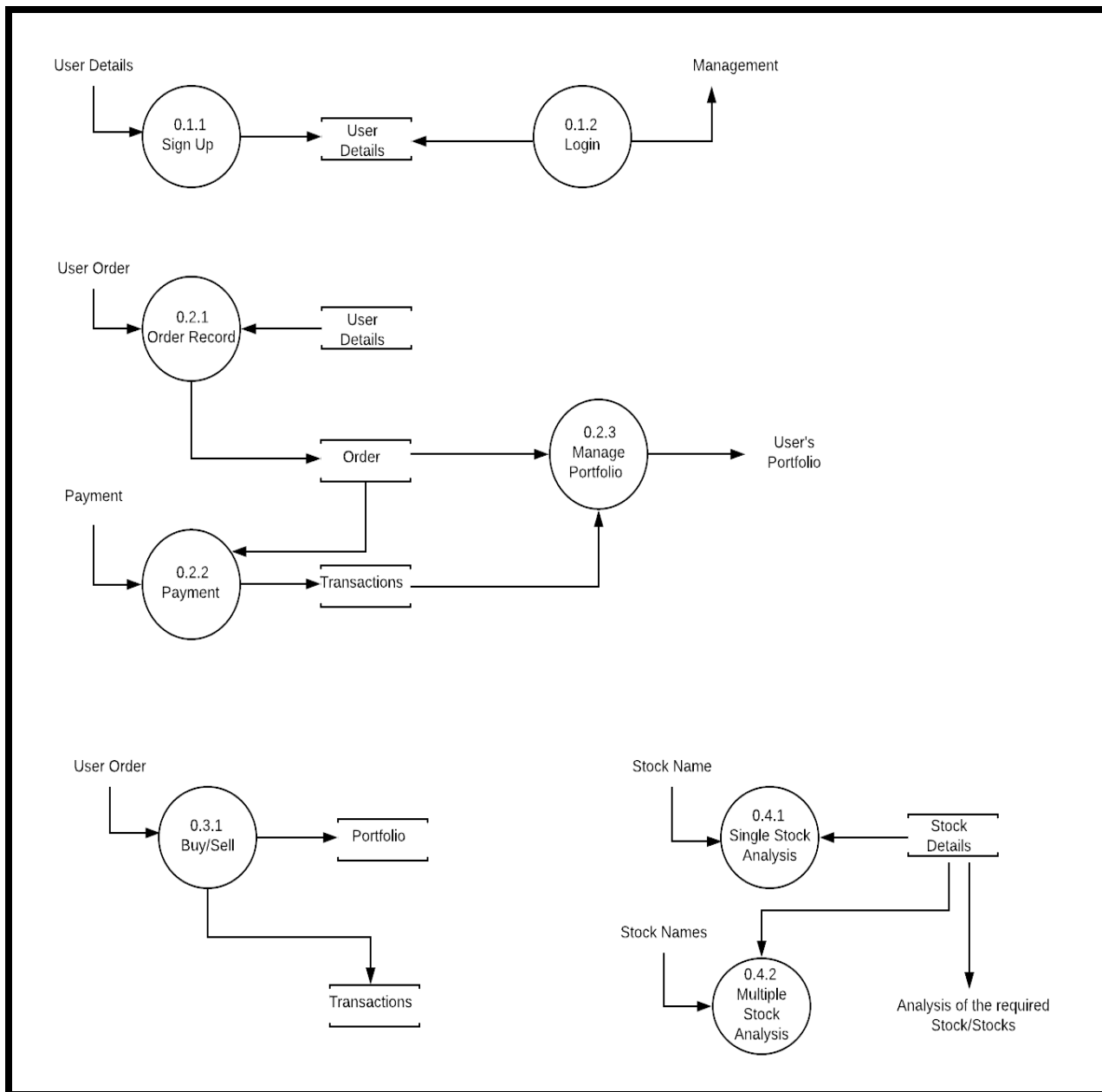




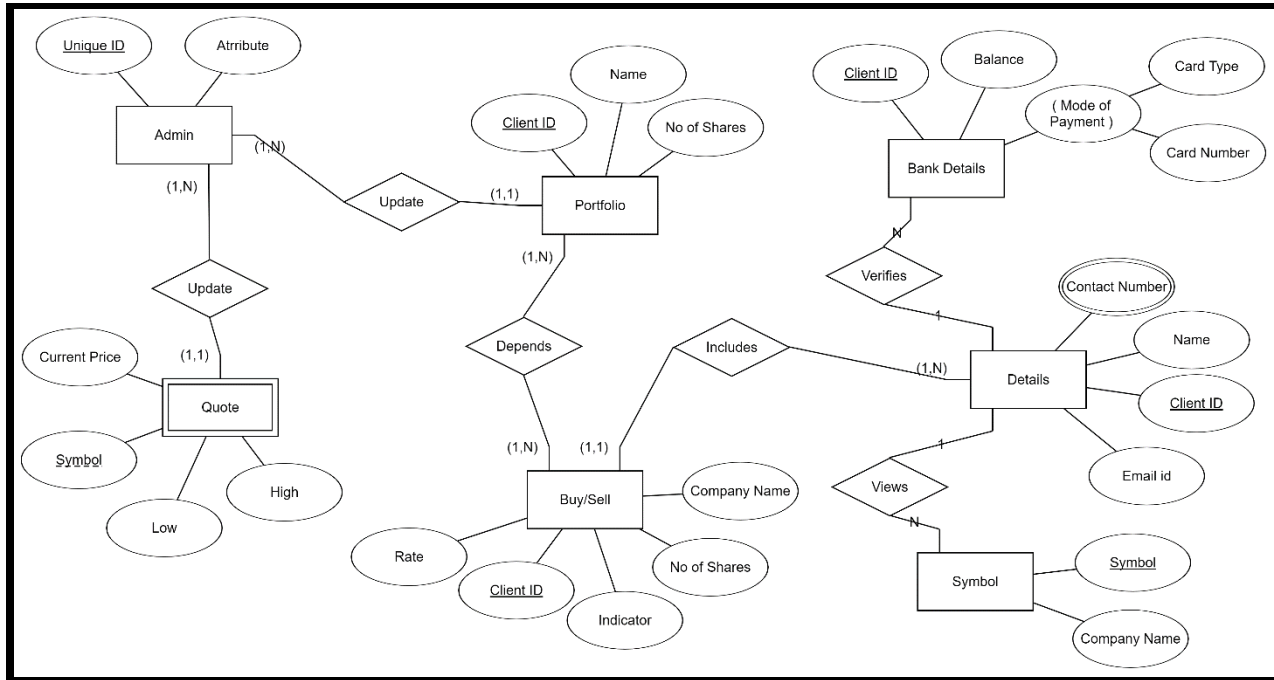
**Zero/Context Level DFD**



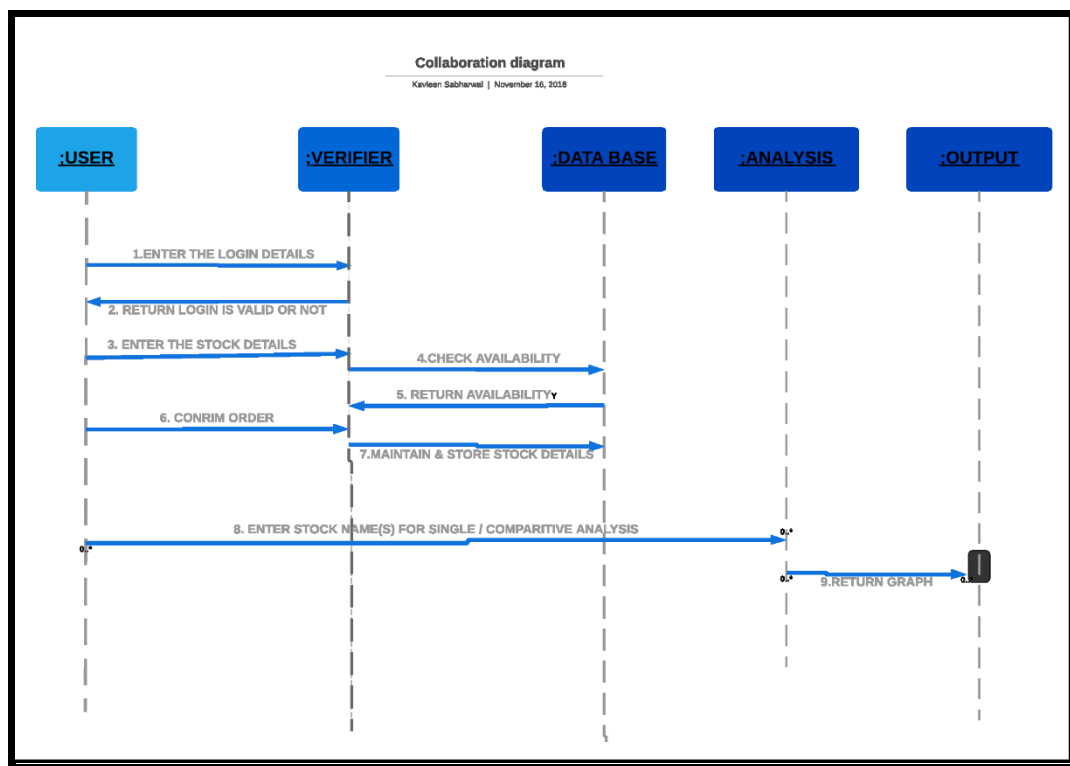
**First Level DFD**



**Second Level DFD**

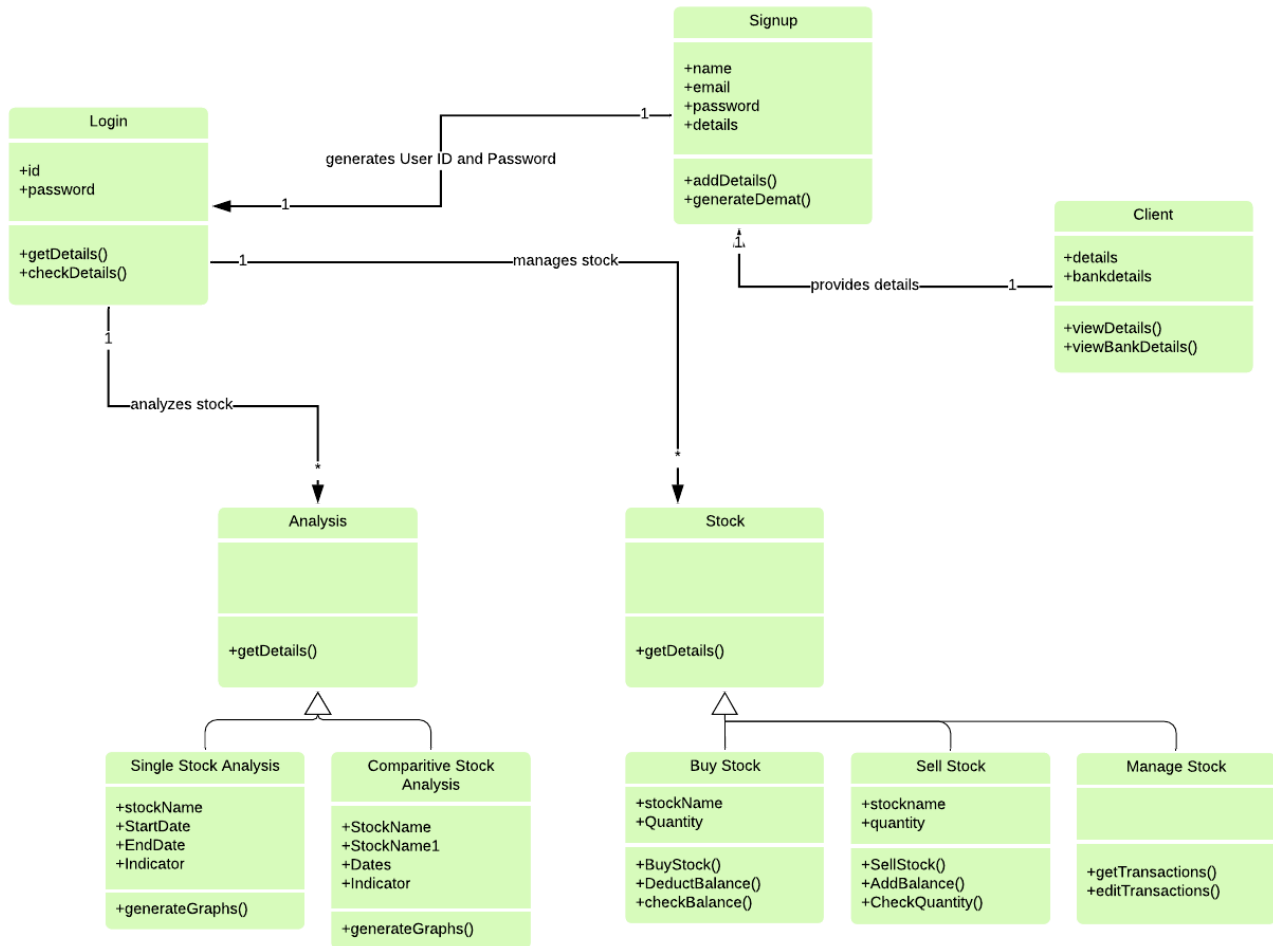


**E-R Diagram**



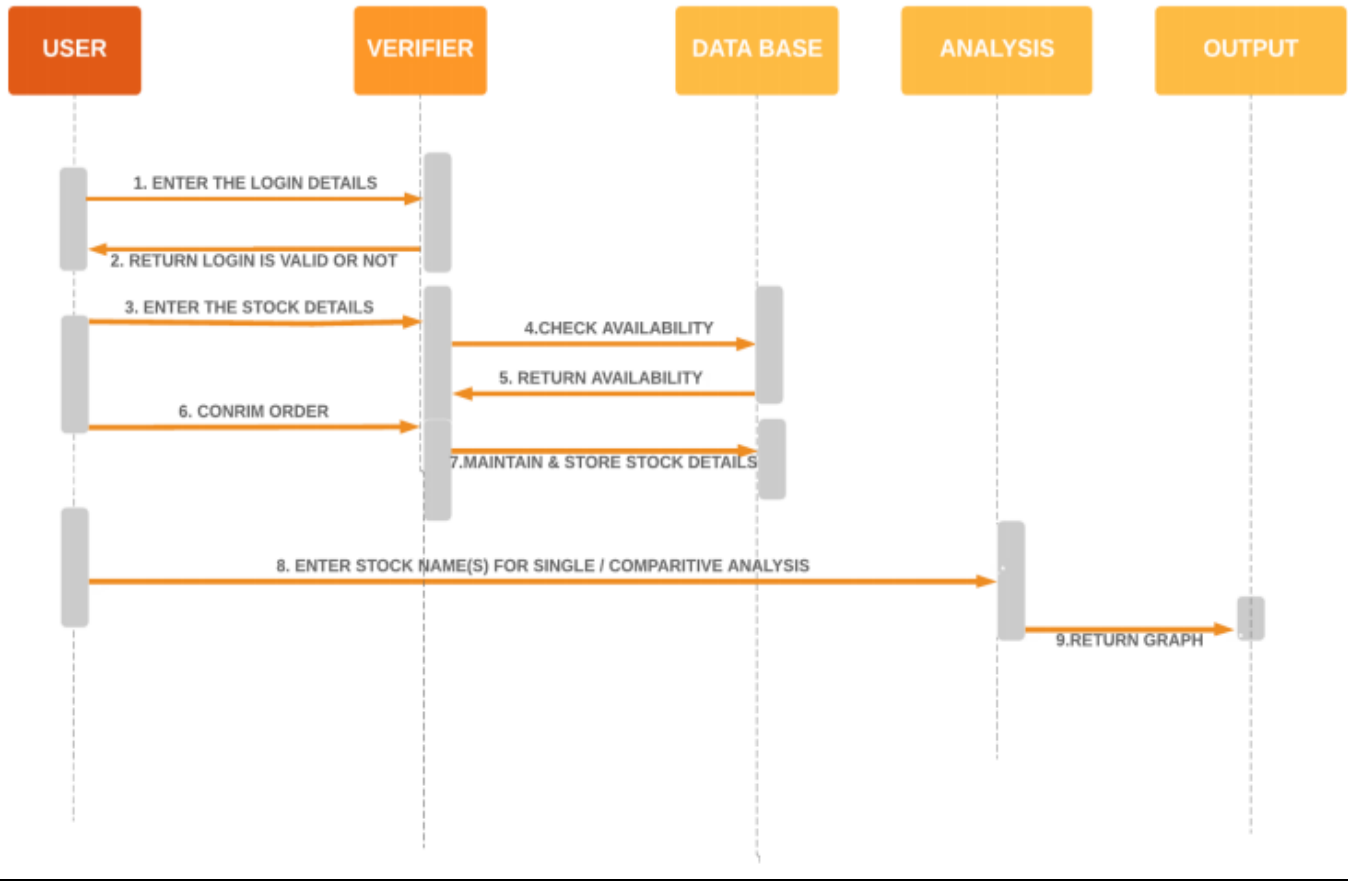
**Collaboration Diagram**

## Class Diagram for SMAS



**Class Diagram**

Sequence diagram Stock Management  
and Analysis System  
Kavleen Sahasrwal | October 12, 2018

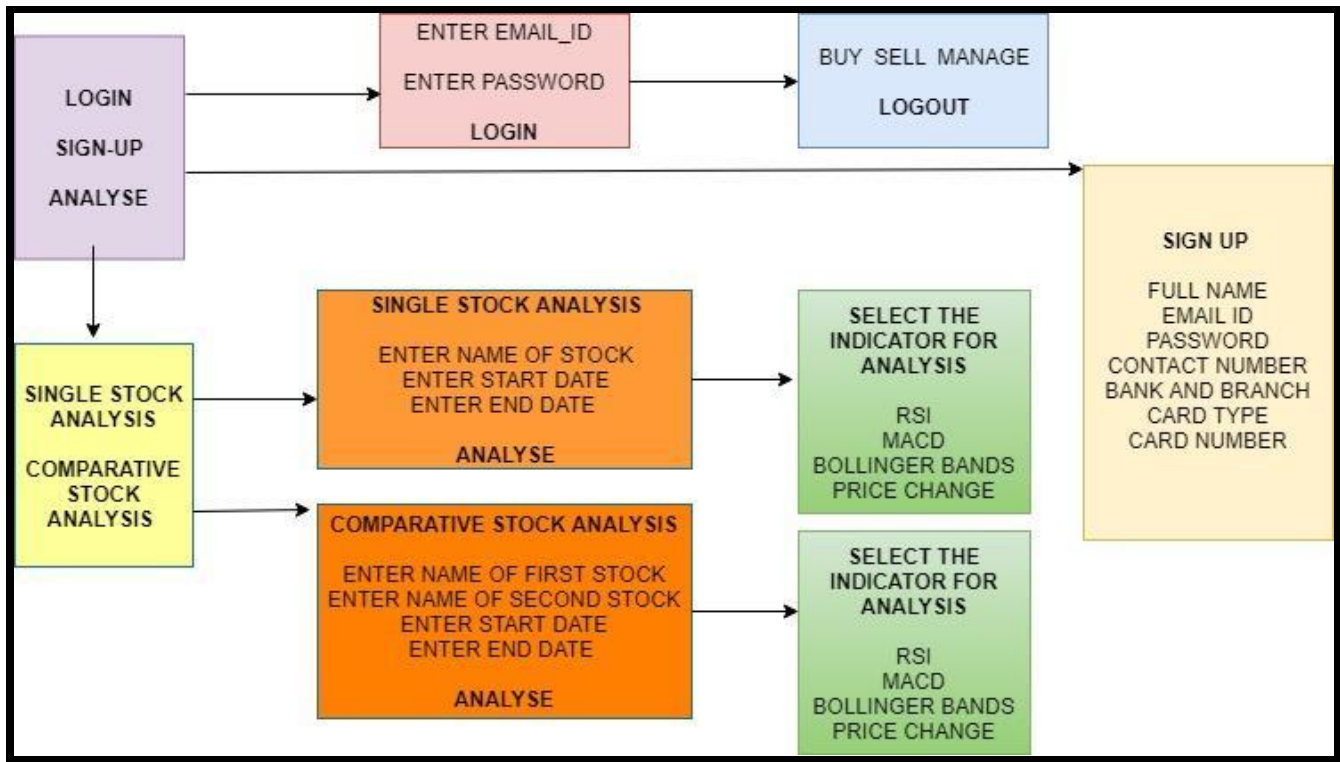


Sequence Diagram

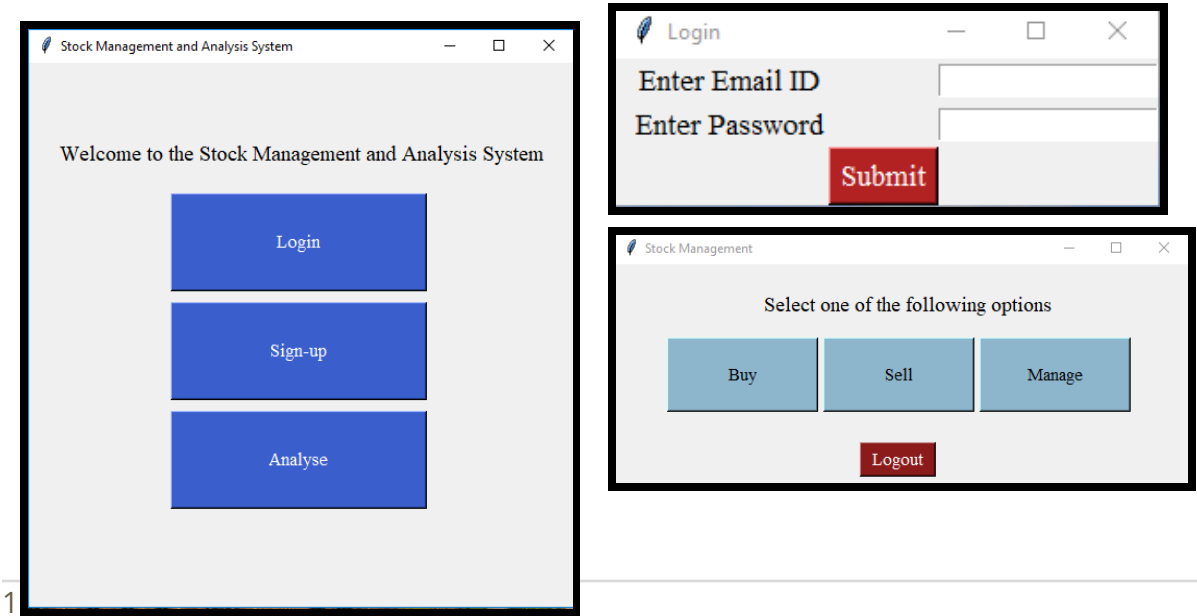


# 3. System Design

## 3.1. Design Goals



## 3.2. Design



Sign-up

Sign-up

Full Name

Email ID

Password

Contact Number

Bank and Branch

Card Type

Card Number

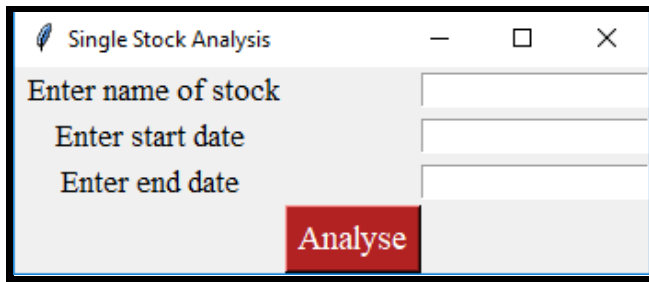
Submit

Stock Analysis

Select the type of analysis to be done

Single Stock Analysis

Comparative Stock Analysis



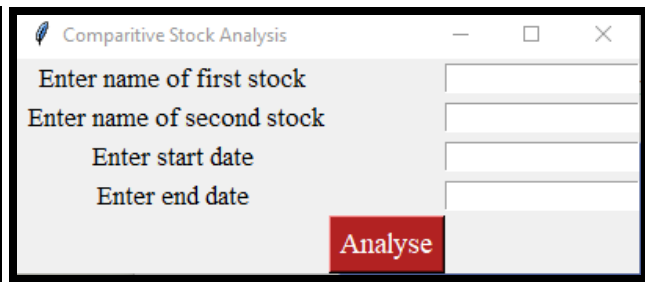
Single Stock Analysis

Enter name of stock

Enter start date

Enter end date

Analyse



Comparative Stock Analysis

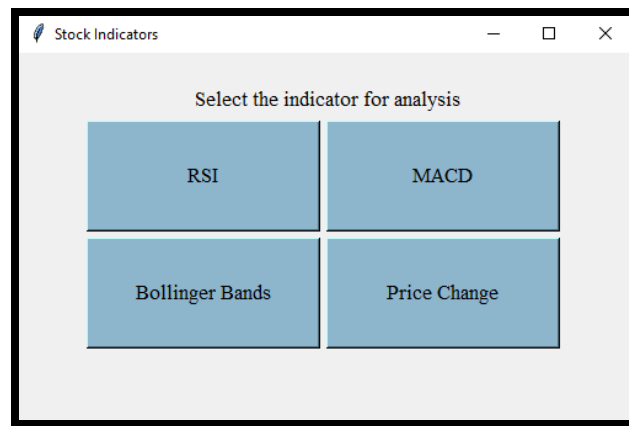
Enter name of first stock

Enter name of second stock

Enter start date

Enter end date

Analyse



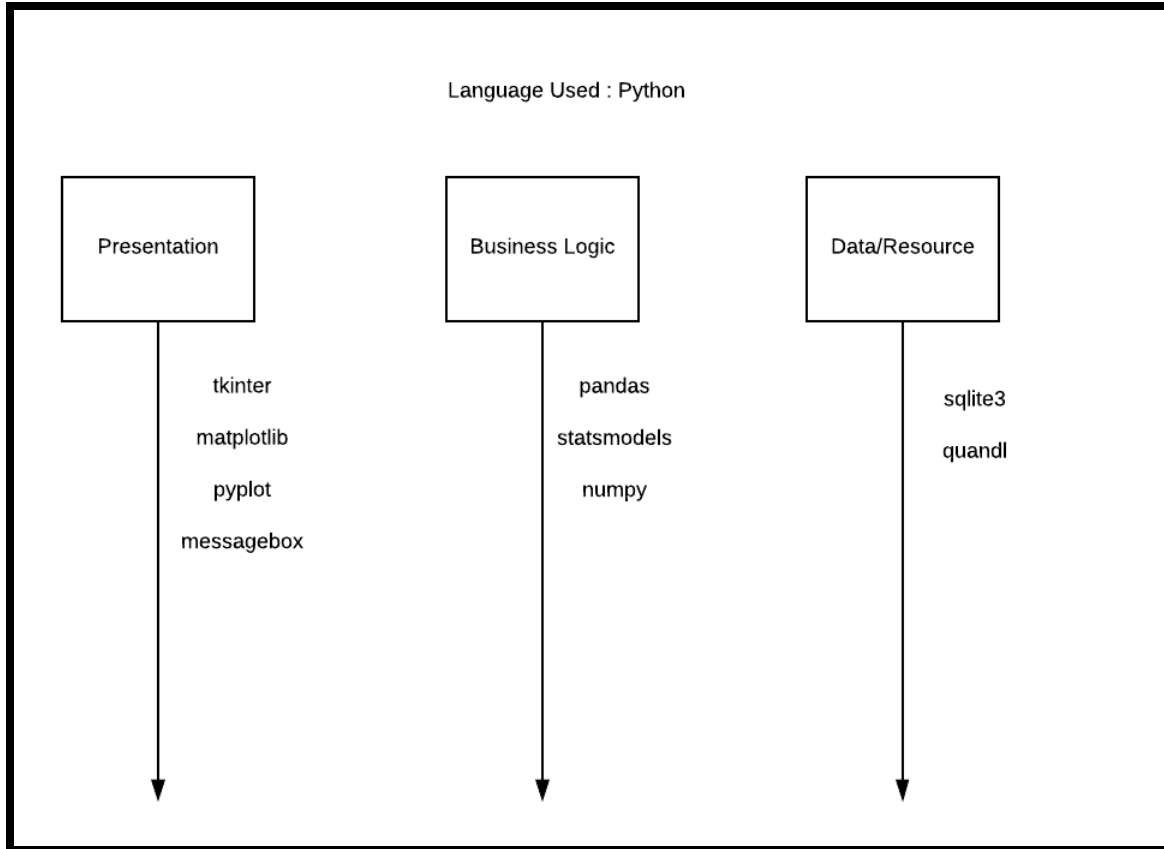
Stock Indicators

Select the indicator for analysis

RSI	MACD
Bollinger Bands	Price Change

### 3.3. System Architecture

Our application follows three tier architecture which has presentation, business logic and data base layer. The presentation layer explains about the user interface, the business layer explains the various logics that are used to implement our functionalities and the database is on SQLite and Quandl is used to fetch the data.



## 4. Work Done

### 4.1 Development Environment

- Hardware used: HP Probook 440 G3
- Operating System: Ubuntu 18.04 LTS, Windows 10
- IDE's: IDLE, Spyder
- Language used: Python
- Libraries used: Matplotlib, Pandas, Statsmodels, Tkinter, Quandl

### 4.2 Results and Discussion

- As a result of coding process, we developed a desktop app that analyzes the stocks, the app fetches data based on various given dates.
- We were able to implement 4 Technical Indicators, for Single Stock and Comparative Stock Analysis.
- We were also able to connect with Backend and implement Login and Signup Modules.

### 4.3 Individual Contribution of Project Members

- Mehil: Team Leader and Back-End Developer.
- Maheeka: Front End and GUI designer.
- Kavleen: Documentation, User Stories and Testing.
- Mehul: Complete Requirement Analysis and all related diagrams.



# 5. Conclusion and Future

## 5.0. Conclusion

We have created an app which is deployable in the current market. It has the functionalities lacking in other such Stock Analysis and Management apps and is user friendly on account of the simple and efficient user interface. The app has been tested using both white and black box testing methods and has passed the test cases for all functional requirements and performance parameters.

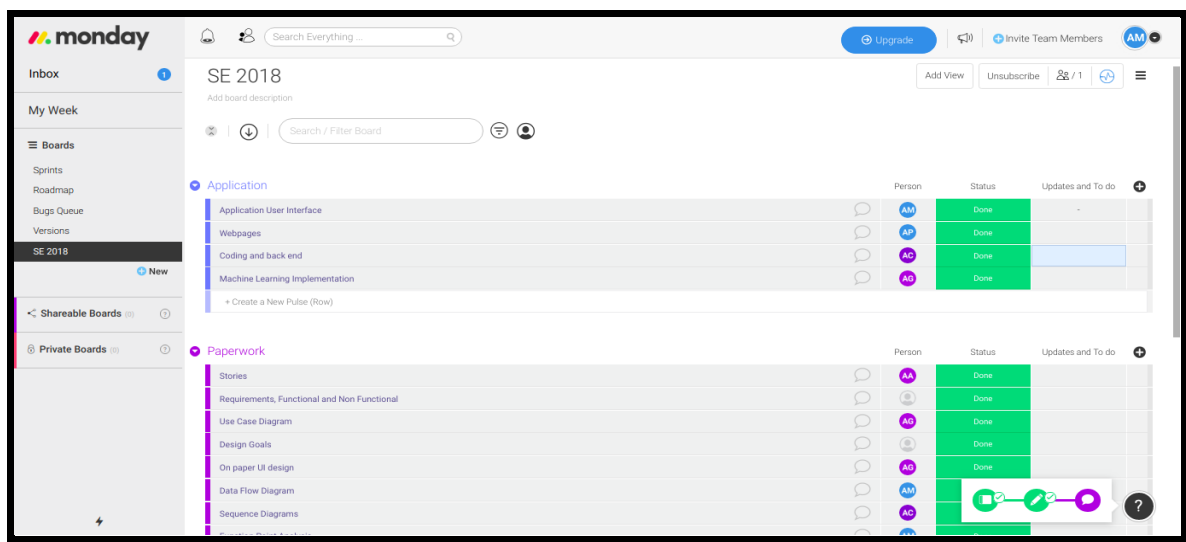
## 5.1. Future Scope

There is a scope for adding additional functionalities in the future such as commodity-wise analysis which is scalable to other markets. It also has scope for development into a mobile application which analyses real-time stocks and notifies the user when his/her stock prices fall or rise. A suggestive chat bot can also be made which on the basis of user restrictions and preferences, suggests the stock which will profit the user.

## 5.2 Task board

Using the Agile methodology, we went through a systematic approach of using a task board to be updated with the current tasks in hand, task completed and task yet to be done in the future.

This represent the assignment of the task to the team member and its current status.



[+ Invite Team Members](#)

Inbox 1

My Week

Boards

Sprints

Roadmap

Bugs Queue

Versions

SE 2018

New

Shareable Boards (0)

Private Boards (0)

Application

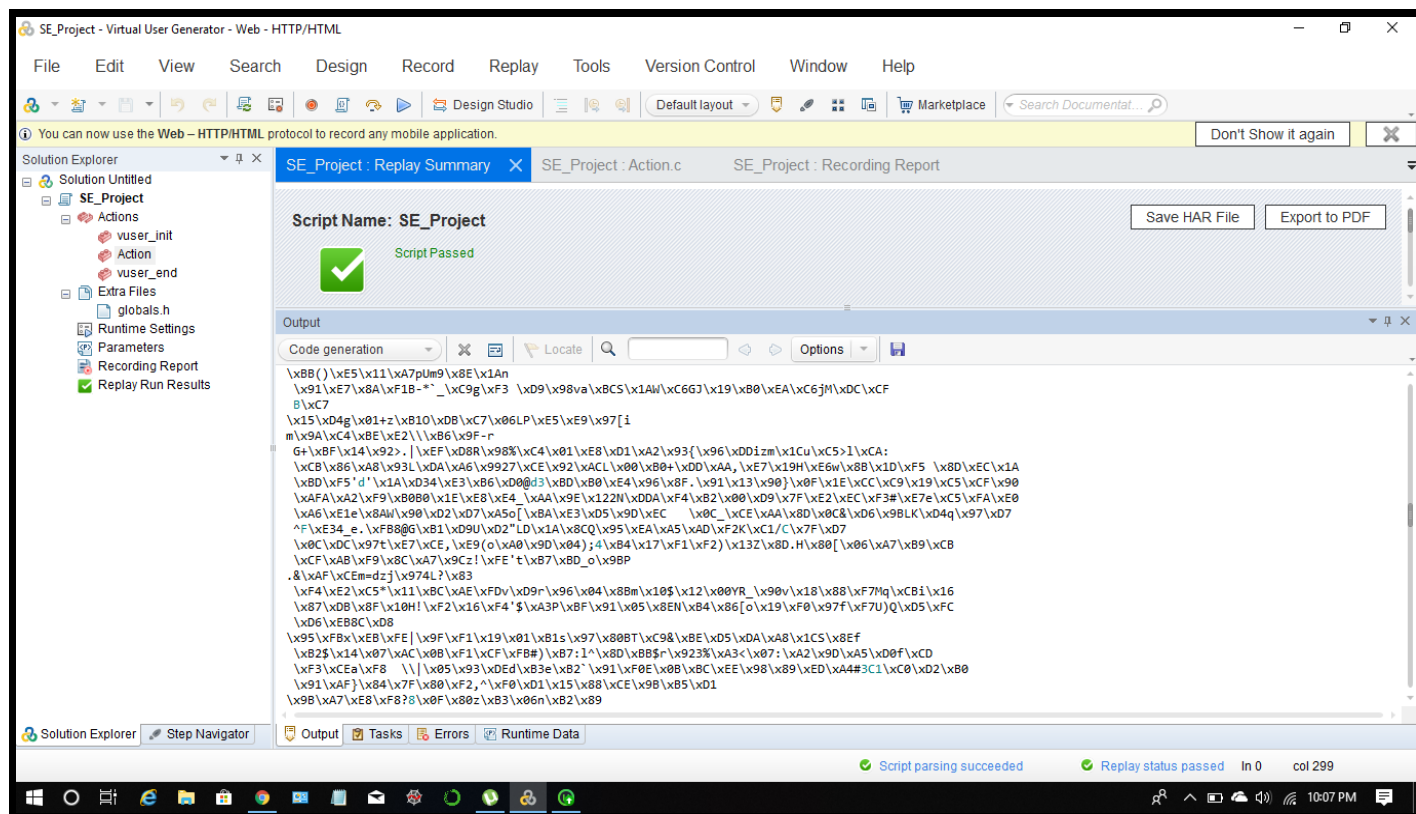
Paperwork

Stories		AA	Done	
Requirements, Functional and Non Functional			Done	
Use Case Diagram		AG	Done	
Design Goals			Done	
On paper UI design		AG	Done	
Data Flow Diagram		AM	Done	
Sequence Diagrams		AC	Done	
Function Point Analysis		AM	Done	
Class Diagram ( Needs Edits )		AA	Done	
Class Diagram ( Final )		AM	Done	
Entity Relationship Diagram		AM	Done	
Class Collaborator Cards		AG	Done	
Cocoma		AG	Done	
Test Cases : Black box		AA	Done	
Testing White box		AA	Done	
Load Runner Report		AA	Working on it	
Report		AM		
+ Create a New Pulse (Row)				

Person	Status	Updates and To do
Person	Status	Updates and To do



## 6. Testing Tool Report



We recorded our actions on the application using LoadRunner and the result of the script generated is attached above. We used LoadRunner to test the behaviour and performance of our application under load. This is done by recording and simulating interactions with the user interface such as keypresses or mouse movements. The messages and interactions to be generated are stored in scripts. Our application was tested against load and the test turned out to be successful.