

## **WORKBOOK**

**Project Title:** Stock Virtual Machine

**Team:** Sujal Patel, Linisha Basu, Sujay Patel

**Client:** Benjamin Hughes

Link to the git repository:

<https://github.com/sujaypatel/CS422Project>

### **Description:**

The Stock Virtual Machine is a web application that is designed to simulate buying and selling stocks in the stock market in a very realistic manner. Using the Stock Virtual Machine, user will get the full experience of buying and selling stocks without having to invest any actual currency.

### **Requirements:**

<b>Use case name</b>	<b>UserRegistration</b>
<b>Participating actors</b>	New Users
<b>Flow of events</b>	<ol style="list-style-type: none"><li>1. User opens the browser and visits the URL localhost.</li><li>2. The system displays the Home screen with a Signup, Login, and Help button.</li><li>3. User clicks the "Sign up" button.</li><li>4. The system takes user to the registration page and asks to enter a username, password and an email address.</li><li>5. Once user completes the form, and presses "Create Account" button, the user gets redirected to Portfolio page.</li><li>6. The system displays user's portfolio, which contains a username and virtual bank account with \$1000 balance.</li></ol>
<b>Entry condition</b>	User access the website through a web browser
<b>Exit condition</b>	User successfully creates the account.
<b>Quality/Nonfunctional Requirements</b>	The application should be loaded in 5 seconds after the user accesses it on a web browser

<b>Use case name</b>	<b>Login</b>
	<ol style="list-style-type: none"><li>1. User opens the browser and visits the URL localhost.</li><li>2. The system displays the Home screen with a Signup, Login, and Help button.</li><li>3. User clicks the "Login" button.</li></ol>

<b>Flow of events</b>	<ol style="list-style-type: none"> <li>4. The system takes user to Login page and ask user to enter username and password.</li> <li>5. User enters the required information and clicks the “Submit” button.</li> <li>6. The system verifies the username and password from database.</li> <li>7. If the user enters incorrect information, then system asks user to reenter the information.</li> <li>8. If the user enters correct information then system takes the user to portfolio page.</li> </ol>
<b>Entry condition</b>	User access the website through a web browser
<b>Exit condition</b>	User successfully is able to login.
<b>Quality/Nonfunctional Requirements</b>	Authentication should be done within 2 seconds.

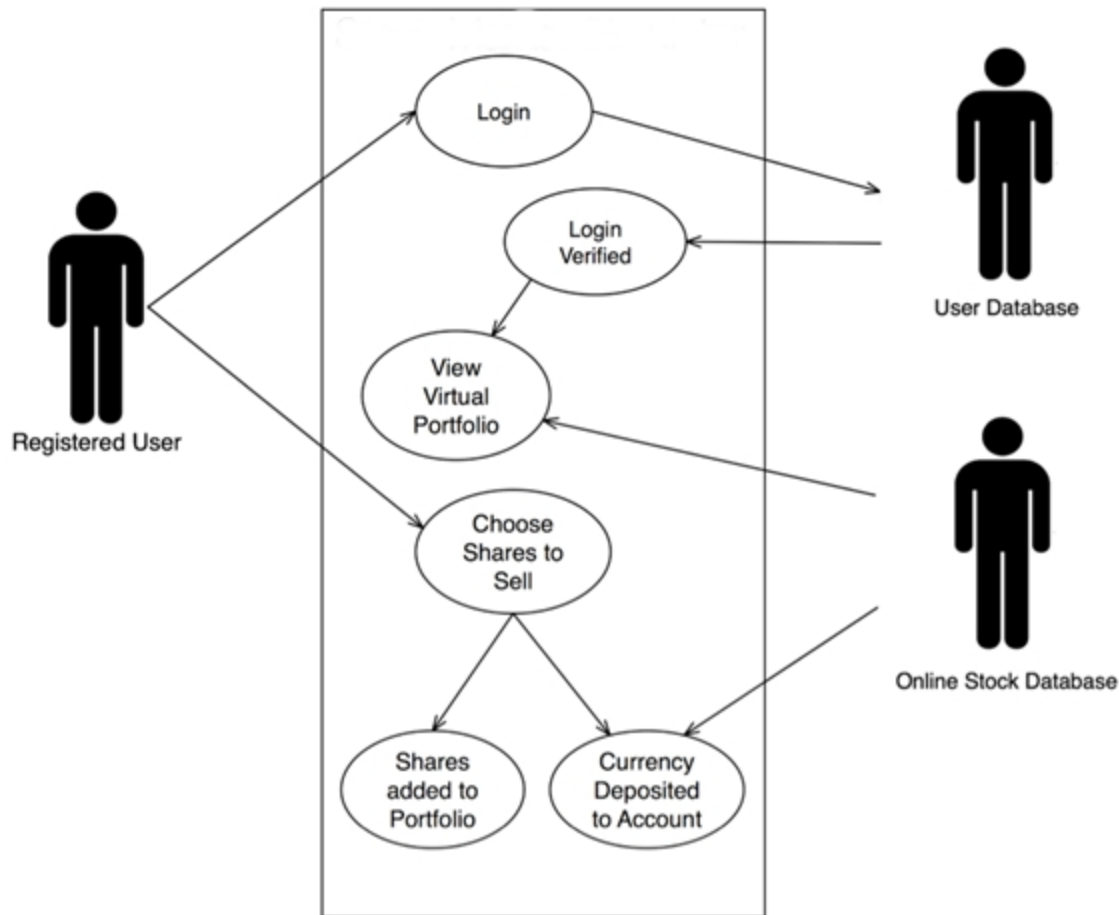
<b>Use case name</b>	<b>BuyStock</b>
<b>Participating actors</b>	Registered Users
<b>Flow of events</b>	<ol style="list-style-type: none"> <li>1. User opens the browser and visits the URL localhost.</li> <li>2. Once the user Login successfully, then the system takes the user to portfolio page.</li> <li>3. On portfolio page, the system provides user with the username, virtual balance, and three buttons i.e. BuyStock, SellStock, SearchStock.</li> <li>4. User click on “BuyStock” button.</li> <li>5. The system takes user to BuyStock page where user will be able to select the company name and enter the quantity of stock that user want to buy for a select company.</li> <li>6. The user click on “Purchase” button.</li> <li>7. The system makes sure that user has enough amount of virtual money to purchase the stock.</li> <li>8. If the user has enough virtual money then the system allows user to purchase the stock and takes the user back to portfolio page.</li> <li>9. On the portfolio page, it would display the list of stocks that the user owns and would update the virtual balance.</li> </ol>
<b>Entry condition</b>	User access the website through a web browser
<b>Exit condition</b>	User successfully able to purchase the stock.
<b>Quality/Nonfunctional Requirements</b>	Once the user purchase the stock, their virtual balance and their portfolio should be updated within 5 seconds of the transaction.

<b>Use case name</b>	<b>SellStock</b>
<b>Participating actors</b>	Registered Users
<b>Flow of events</b>	<ol style="list-style-type: none"> <li>1. User opens the browser and visits the URL localhost.</li> <li>2. Once the user Login successfully, then the system takes the user to portfolio page.</li> <li>3. On portfolio page, the system provides user with the username, virtual balance, and three buttons i.e. BuyStock, SellStock, SearchStock.</li> <li>4. User click on "SellStock" button.</li> <li>5. The system takes user to SellStock page where user will be able to select the company name and enter the quantity of stock that user want to sell for a select company.</li> <li>6. The user clicks on "Sell" button.</li> <li>7. The system makes sure that user is not able to sell more stock than he owns.</li> <li>8. Once the user sells the stocks then the system takes the user back to portfolio page.</li> <li>9. On the portfolio page, it would display only the list of stocks that the user currently owns and would update the virtual balance.</li> </ol>
<b>Entry condition</b>	User access the website through a web browser
<b>Exit condition</b>	User successfully able to sell the stock.
<b>Quality/Nonfunctional Requirements</b>	Once the user sells the stock, their virtual balance and their portfolio should be updated within 5 seconds of the transaction.

<b>Use case name</b>	<b>SearchStock</b>
<b>Participating actors</b>	Registered Users

<b>Flow of events</b>	<ol style="list-style-type: none"> <li>1. User opens the browser and visits the URL localhost.</li> <li>2. Once the user Login successfully, then the system takes the user to portfolio page.</li> <li>3. On portfolio page, the system display user with the username, virtual balance, and three buttons i.e. BuyStock, SellStock, SearchStock.</li> <li>4. User clicks on "SearchStock" button.</li> <li>5. The system takes user to Search page where user will be able to select the company name and click "Search" button.</li> <li>6. The system displays the stock information and current price of the selected company.</li> <li>7. If the user want to go back to portfolio page. then the user can click on "BackPortfolio" button.</li> </ol>
<b>Entry condition</b>	User access the website through a web browser.
<b>Exit condition</b>	User successfully able to sell the stock.

<b>Use case name</b>	<b>ExitApplication</b>
<b>Participating actors</b>	Registered Users
<b>Flow of events</b>	<ol style="list-style-type: none"> <li>1. User opens the browser and visits the URL localhost.</li> <li>2. Once the user Login successfully, then the system takes the user to portfolio page.</li> <li>3. The system provides a "Logout" Button on the right side of the screen.</li> <li>4. User clicks on the "Logout" button.</li> <li>5. The system saves the user data and goes to login page.</li> </ol>
<b>Entry condition</b>	User access the website through a web browser.
<b>Exit condition</b>	User successfully able to logout from the application.
<b>Quality/Nonfunctional Requirements</b>	The game should terminate within 2 seconds.



## Specifications:

### Goals:

#### Low Target:

Our team goal is to accomplish several things before March 16. For example, Ted is new to the stock market and wants to get a real world experience before venturing into the actual stock market. Using the Stock Virtual Machine, Ted shall be able to create an account; the system must allow Ted to login. Ted should be able to access his portfolio page.

**Use Cases:** UserRegistration, Login, & ExitApplication

#### Your desirable target:

Our next step then would be to implement the functionality of buy, sell, and search stocks feature. Ted will get the full experience of buying and selling stocks without having to invest any actual currency. On the user's portfolio webpage, the system shall update the table of the stocks and virtual balance once Ted makes any transaction. The system must prevent Ted from buying more stocks than the amount of virtual balance he has.

**Use Cases:** BuyStock, SellStock, & SearchStock

### Your high target:

At the end, our team goal would be to design the user-interface to make our web application look simple and attractive to the users.

### Your extras:

Extra feature that could be implemented in the future would be RSS Feed, and Forget Password Recovery.

### Non-Goals:

- User authentication should be done within 2 seconds.
- The user should be able to view their latest transactions no later than 5 seconds.
- The system should not crash more than once in a week.
- The system shall ensure that the data is protected from unauthorized access.
- In case of an error, the time needed to get the system back to running should not exceed an hour.
- The system should support multiple users simultaneously.

### Group Prototypes

**Prototype #1**

## User Registration

User **MUST** register to use the application. Once they register, the system takes the user to the Login page.

HOME SIGNUP LOGIN HELP

Welcome to Stock Virtual Machine.

WELCOME

Stock Virtual Machine Web application which provides a learning opportunity for new users to become experts in Stock Market without spending actual money.

To use the application users have to Sign Up.

....

HOME SIGNUP LOGIN HELP

Create an Account

REGISTER NOW

The user will get the full experience of buying and selling stock on the web without having to invest actual money.

USERNAME:

PASSWORD:

CONFIRM PASSWORD:

EMAIL ADDRESS:

REGISTER

HOME SIGNUP LOGIN HELP

Login to Stock Virtual Machine

LOGIN

To use the program users must register with a Username, Email, and password.

USERNAME:

PASSWORD:

LOGIN

## User Login

Once user enter valid username and password then system takes the user to the portfolio page.

HOME SIGNUP LOGIN HELP

Welcome to Stock Virtual Machine

WELCOME

Stock Virtual Machine Web application which provides a learning opportunity for new users to become experts in stock market without spending actual money. To use the application users have to Sign Up. ....

HOME SIGNUP LOGIN HELP

Login to Stock Virtual Machine

LOGIN

To use the Program users must register with a Username, Email, and Password.

USERNAME: SUTALMTEL

PASSWORD: KKKK

LOGIN

Portfolio

Sujal Patel

Virtual Balance: \$1000

User Preferences

Logout

List of Owned Stock

Symbol	High	Low	Close	Quantity
AAP	120.25	119.5	119.75	5
AAT	150.34	145.7	148.8	2
YHOD	45.70	44.5	44.8	5

BUY STOCK

SELL STOCK

SEARCH

## Help Menu

If user needs help using the application. If user click on "Help" then the system takes the user to the Help page.

HOME SIGNUP LOGIN HELP

Welcome to Stock Virtual Machine

WELCOME

Stock Virtual Machine Web application which provides a learning opportunity for new users to become experts in stock market without spending actual money. To use the application users have to Sign Up. ....

HOME SIGNUP LOGIN HELP

Help Menu

INFORMATION

FAQ

CONTACT US

## User Registration

USERNAME   
 EMAIL   
 CONFIRM EMAIL   
 PASSWORD

**STOCK VIRTUAL MACHINE**

USERNAME   
 PASSWORD

## User Login

**STOCK VIRTUAL MACHINE**

USERNAME   
 PASSWORD

LINISHA, WELCOME!

BALANCE: \$235.00

MY STOCKS	
YAHOO	\$150↑
APPLE	\$75↓
GOOG	\$30↑
DAX	\$26↑



## Help Menu

Hand-drawn prototype of a login and registration form. It features a yellow background with horizontal lines. At the top right, there are three buttons: 'LOGIN', 'SEARCH', and 'HELP'. Below these, there are input fields for 'USERNAME' (containing 'LINISHA'), 'EMAIL' (containing 'lpatil2@vic.edu'), 'CONFIRM EMAIL' (containing 'lpatil2@vic.edu'), and 'PASSWORD' (containing 'xxxxxx'). A 'SIGNUP!' button is located at the bottom right.

Hand-drawn prototype of a help menu. It features a yellow background with horizontal lines. There are three buttons: 'FAQ', 'ABOUT US', and 'CONTACT US'.

## User Registration

Hand-drawn prototype of a user registration system, titled 'User Registration'. It consists of four screens labeled #1, #2, #3, and #4.

- #1:** A 'Welcome Page' with a 'Stock virtual Machine' header. It has a navigation bar with 'SignUp', 'Login', 'Help', and 'Contact Us' buttons.
- #2:** A 'Sign UP Page' with a navigation bar where 'SignUp' is highlighted. It contains input fields for 'First Name', 'Last Name', 'Gender' (with radio buttons for 'Male' and 'Female'), 'Email Address', 'Username', and 'Password'. A 'Create Account' button is at the bottom.
- #3:** A 'Sign UP Page' with a navigation bar where 'SignUp' is highlighted. It contains input fields for 'First Name' (filled with 'SURAJ'), 'Last Name' (filled with 'PATEL'), 'Gender' (with radio buttons for 'Male' and 'Female'), 'Email Address' (filled with 'lpatil2@vic.edu'), 'Username' (filled with 'SURAJ'), and 'Password' (filled with 'xxxxxx'). A 'Create Account' button is at the bottom.
- #4:** A 'Login Page' with a navigation bar where 'Login' is highlighted. It contains input fields for 'Username' and 'Password', and a 'Login' button.

# User Login

## Login

#1

Sign Up **Login** Help Contact Us

Login Page

Username:

Password:

#2

Sign Up **Login** Help Contact Us

Login Page

Username:

Password:

#3

Portfolio Page

Hello, SUJAY

YAHOO	<input type="button" value="Buy Stock"/>
GOOGLE	<input type="button" value="Sell Stock"/>
FACEBOOK	<input type="button" value="Watch Stock"/>
AMAZON	
TESLA	