Software Requirements Specification

for

StockUp

Technical Analysis Application

Version 1.0 approved

Prepared by: TEAM #6

Concordia University

April 12th, 2017

Table of Contents

Team Members

Name	**Student ID**
Mehal Patel	26789234
Dinh Bui	27321651
Jordan Orsini	26471196
Matthew To	40005135
Matthew Verrucci	26795528
Sri Ram Ede	40038478
Vasiliki Boutas	26220304

Revision History

Name	**Date**	**Reason For Changes**	**Version**
Vasiliki Boutas	4/4/2017	Updated Use Case diagram	1.0
-	12/4/2017	Final Revision	1.0

1. Introduction

1.1. Purpose

This Software Requirements Specification (SRS) document identifies and describes the key features of the **StockUp Version 1.0** technical stock analysis software. The application is meant to display price movements for the current **DOW 30** stocks in the stock market. These features include viewing, adding and removing moving averages over an adjustable range, which are displayed in graph form on the application. A login system and user log are also part of the application.

1.2. Document Convention

This document uses the Times New Roman and size 12 font, which is the IEEE standard. Each major section heading is bolded and numbered, and subsections are labelled with their parent section's number with an additional decimal. Key words are also bolded, and any notes or important concepts are italicized.

1.3. Intended Audience and Reading Suggestions

This document is mainly targeted toward developer and testers, but can also be useful for regular users who are interested in the software specifications and functionalities. There are currently five major sections which include the Introduction (§1), Overall Description (§2), System Features (§3), System Architecture (§4), and External Interface Requirements (§5). For the average **User**, it is suggested to begin with the Introduction in order to get a general idea of the purpose of this document, and to move on to the System Features section which describes what the product can do, and how it does it. **Developers** may be interested mainly in System Features (primarily 3.1.1-3) and System Architecture.

1.4. Project Scope

Version 1.0

The software reads data and generates a graph for the user to view specific stock *Price Points* over time, as well as *Moving Averages* for selectable pre-defined period. Users cannot buy or sell stocks through the application. It is simply meant as a visual tool for viewing price points and moving averages over time. The *Buy* and *Sell Indicators* that will be displayed on the graph represent short- and long-term crossovers; the user is completely free to decide on whether or not to buy or sell stocks. As of **Iteration 1**, a very basic GUI has been set up and showcases the main functionality of the software. **Iteration 2** will see the completion of the project.

Note: *DOW **30** is an abbreviation for The Dow Jones Industrial Average of the 30 highest indexed publicly owned stocks based in the United States.*

1.5. References

In order to download the data from the Yahoo Finance an API was used from http://financequotes-api.com/ which was added to our application. On their site we obtained the api and the documentation to properly implement the API into the stock application.

Image used in application was from:
https://g.foolcdn.com/editorial/images/206699/looking-for-stocks_large.jpg

2. Overall Description

2.1. Product Perspective

The StockUp desktop application is a self-contained product thought of by **ProfitRUs** with the intention of providing a user-friendly way of viewing price movements in the stock market. The application will have users login to their accounts which will subsequently direct them to the home tab. From there, a user can view his or her recently viewed stocks, or select a new stock to generate a graph with. The graph represents a stock's price point over time. The user is able to choose stock from a drop-down list, or perform a search in a search bar. The date range can also be selected from a drop-down list of predefined ranges. The user can also add or remove moving averages, which would be displayed as overlapping lines on the same graph. Sell and Buy Indicators will be displayed as green up arrows or red down arrows on the graph.

2.2. Product Features

Feature	Details
Login System	● Enter a username ● Enter a password ● Log out
User Log	● View recently viewed stocks ● Select recently viewed stocks
Select Stocks	● Select stock
Select Range	● Select date range
Generate Graph	● Generate graph with selected features
Historical Data	● View all data ● View data for selected date range
Moving Averages	● Add moving average for 20 days ● Add moving average for 50 days ● Add moving average for 100 days ● Add moving average for 200 days ● Remove moving average
Indicators	● Automatically display buy indicator

	● Automatically display sell indicator

2.3. User Classes and Characteristics

Anyone who has a share in the stock market can use StockUp. It is designed to be user-friendly and provide a simple and easy-to-read graph representation of price movements of **DOW 30** stocks. The data generated will be intuitively understandable to users ranging from beginners to experts, and the software itself will be simple to use. **Beginner Users** include those who have no experience in the stock market, for whom this application will serve as an easy transition into the world of stocks, without being intimidating. **Expert Users** are those who know their way around the market and have previously bought or sold stocks. They will find the application to be a go-to tool for a quick look at stock price movement. Due to the simplicity of the product, it may be considered more useful for beginner to intermediate users, who are therefore of higher priority for the purposes of this project.

2.4. Operating Environment

The software will function with only minimal PC requirements.

- Windows XP/Vista or OS X Snow Leopard
- 2GB RAM
- 512MB Memory

2.5. Design and Implementation Constraints

Due to the short amount of time given for the completion of the project, the major constraint faced by the developing team was time. Additionally, the team was limited to using the Java Swing GUI framework, due to lack of knowledge in other frameworks and lack of time to learn. An internet connection will be vital for the final iteration, which will pull data from **Yahoo Finance**.

2.6. User Documentation

A quick guide will be available within the application for users to read through, with details on how to use the software.

2.7. Budget

The objective is to have a functioning software that will allow the user to view, add, and remove moving average of the stock(s) they have selected. The data will be taken from Yahoo Financial, which is freely available on the internet. The size of the project is not relatively big for the number of options it provides, thus making it an **organic type**. The criteria for our adjustment factor are as follows:

Criteria	Priority	Weight
Required software reliability	Low	0.75
Database size	High	1.15
Product complexity	Low	0.75
Execution time constraint	Nominal	1
Main storage constraint	Nominal	1
Virtual machine volatility	Low	0.87
Computer turnaround time	Low	0.87
Analyst capabilities	Nominal	1
Application experience	High	0.91
Programmer capability	Nominal	1
Virtual Machine experience	Very Low	1.21
Programming language experience	Nominal	1
Use of modern programming practices	High	0.91
Use of software tools(nominal):	Nominal	1
Required development schedule	High	1.04

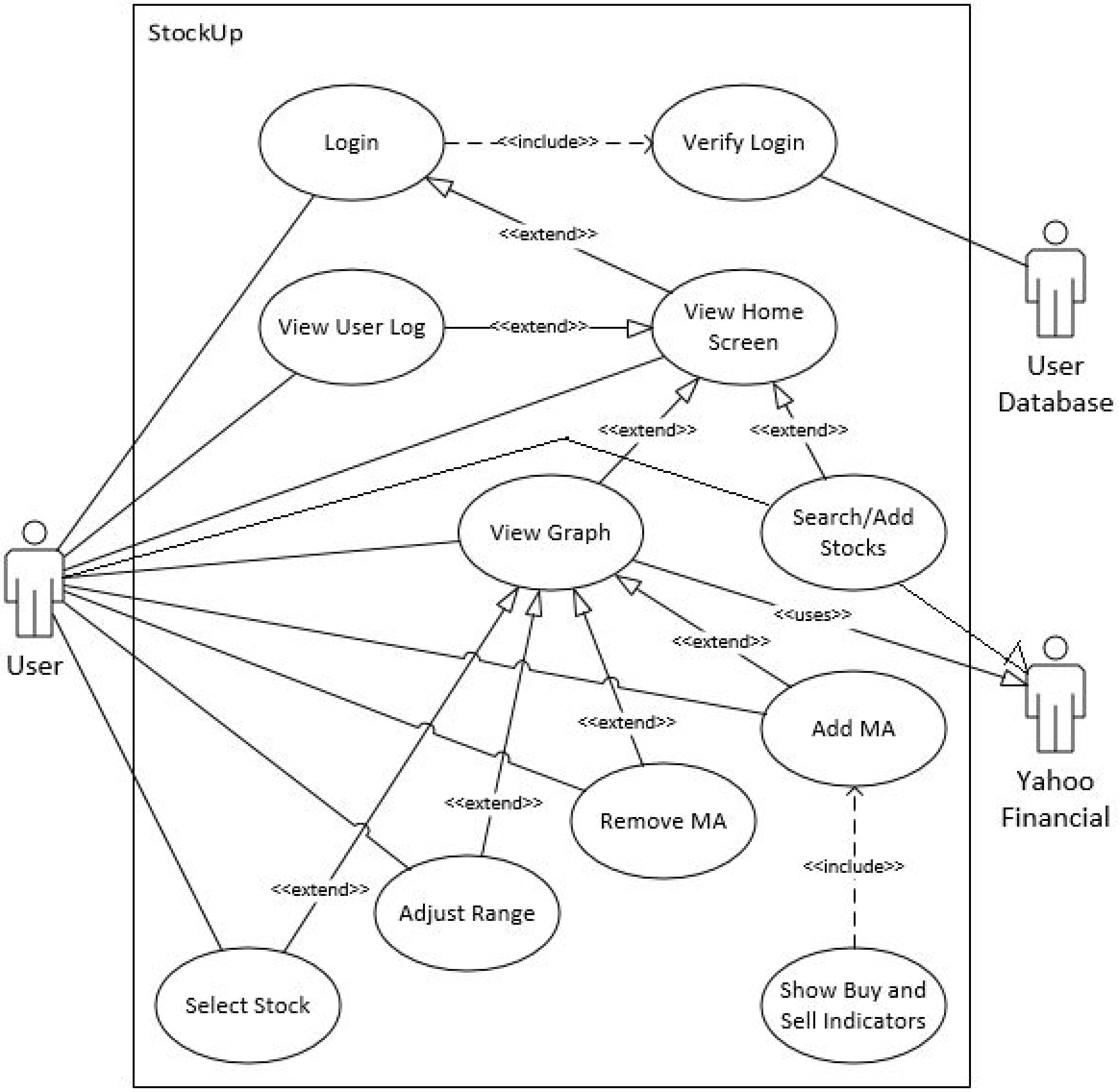
Our adjustment factor is **0.51** so the effort is estimated to be around 2.4*1^1.05*0.51 = 1.224 effort per person/month. Assuming we have a team of 5 with a hourly pay of 20$ and works a total of 40 hours a week, it would take around 16,000(monthly pay) x1.224 = **19,584$**.

3. System Features

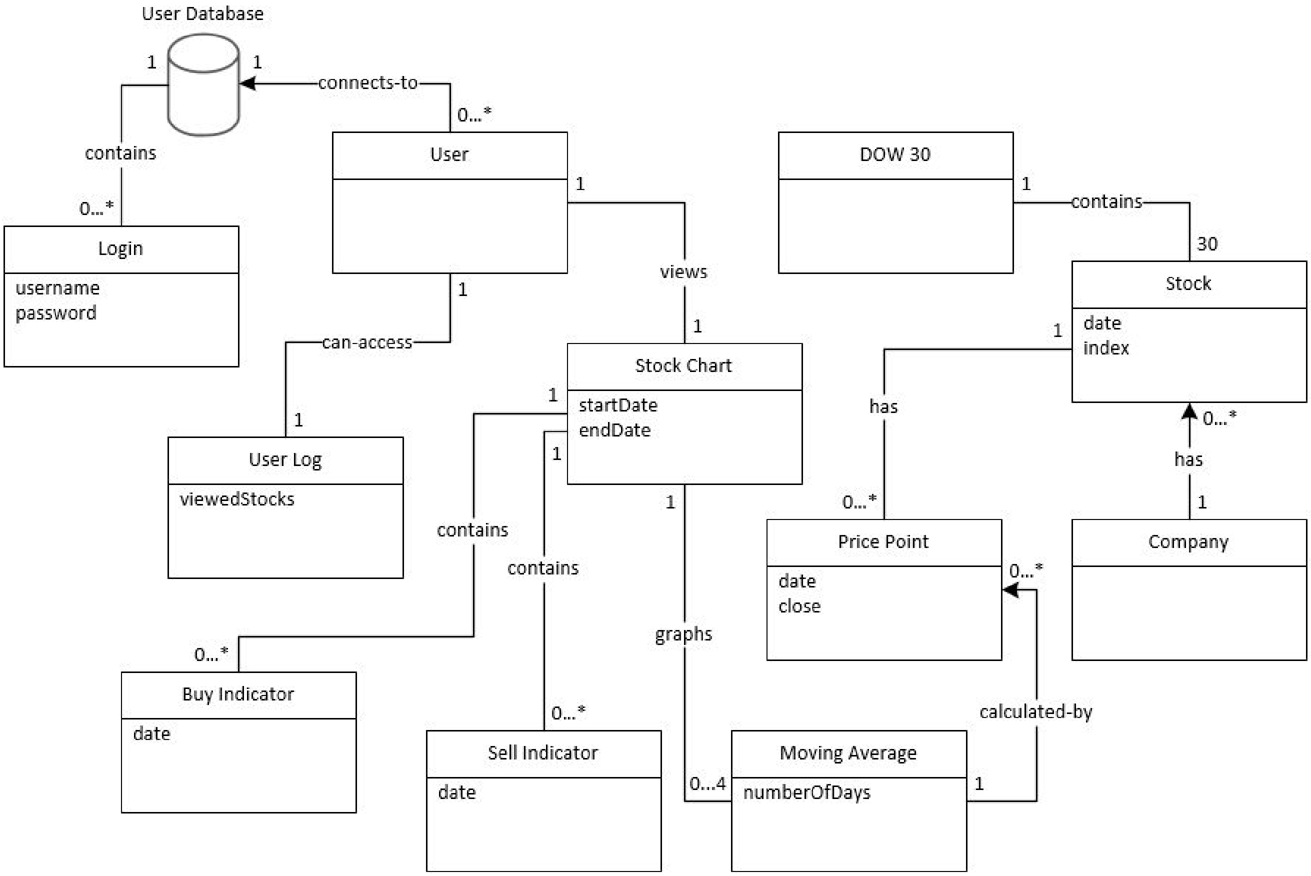
3.1. Project Features

StockUp Version 1.0 is a prototype which will only feature user login, selecting stocks from a drop-down list, viewing the graph, adding moving averages, and changing date ranges. The data will come from a spreadsheet for a handful of currently trending stocks. The final version will get data from *Yahoo Finance* online. The diagrams in §3.1.1-2 are representative of the complete system, while the diagram in §3.1.3 only entails the current prototype version.

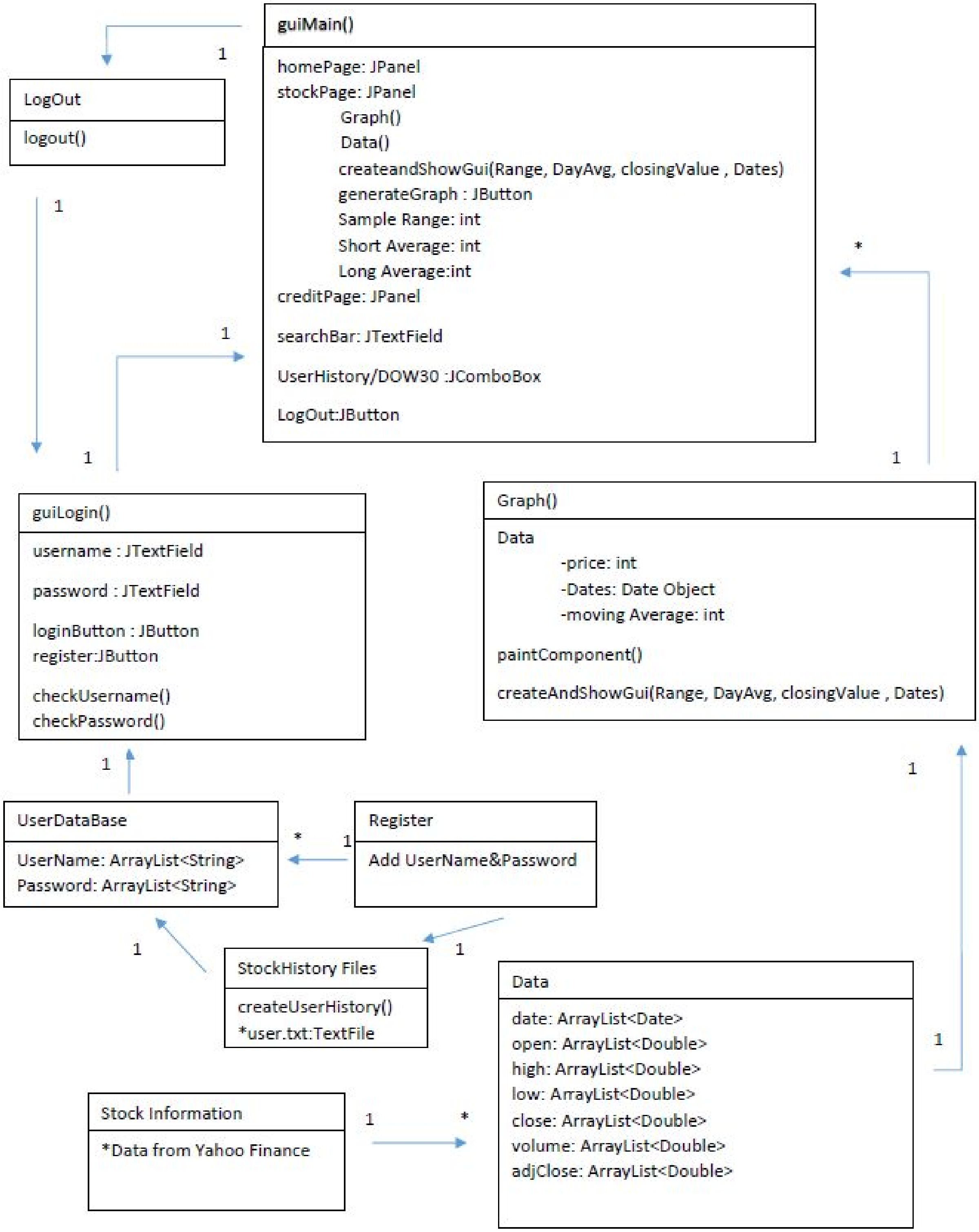
3.1.1. Project Use Case Diagram



3.1.2. Project Domain Model



3.1.3. Class Diagram



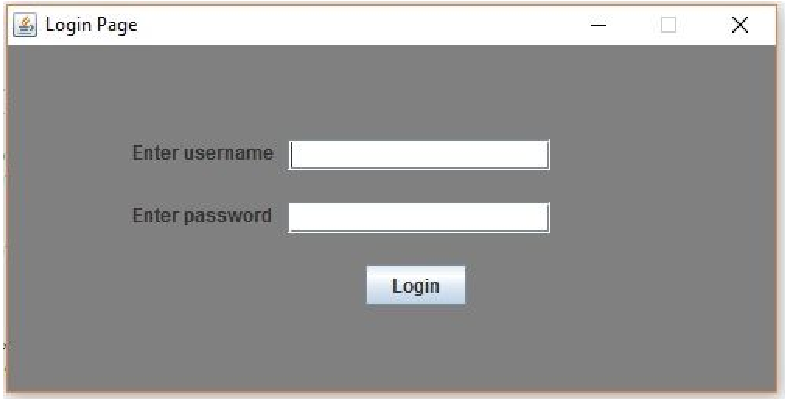
3.2 User Login

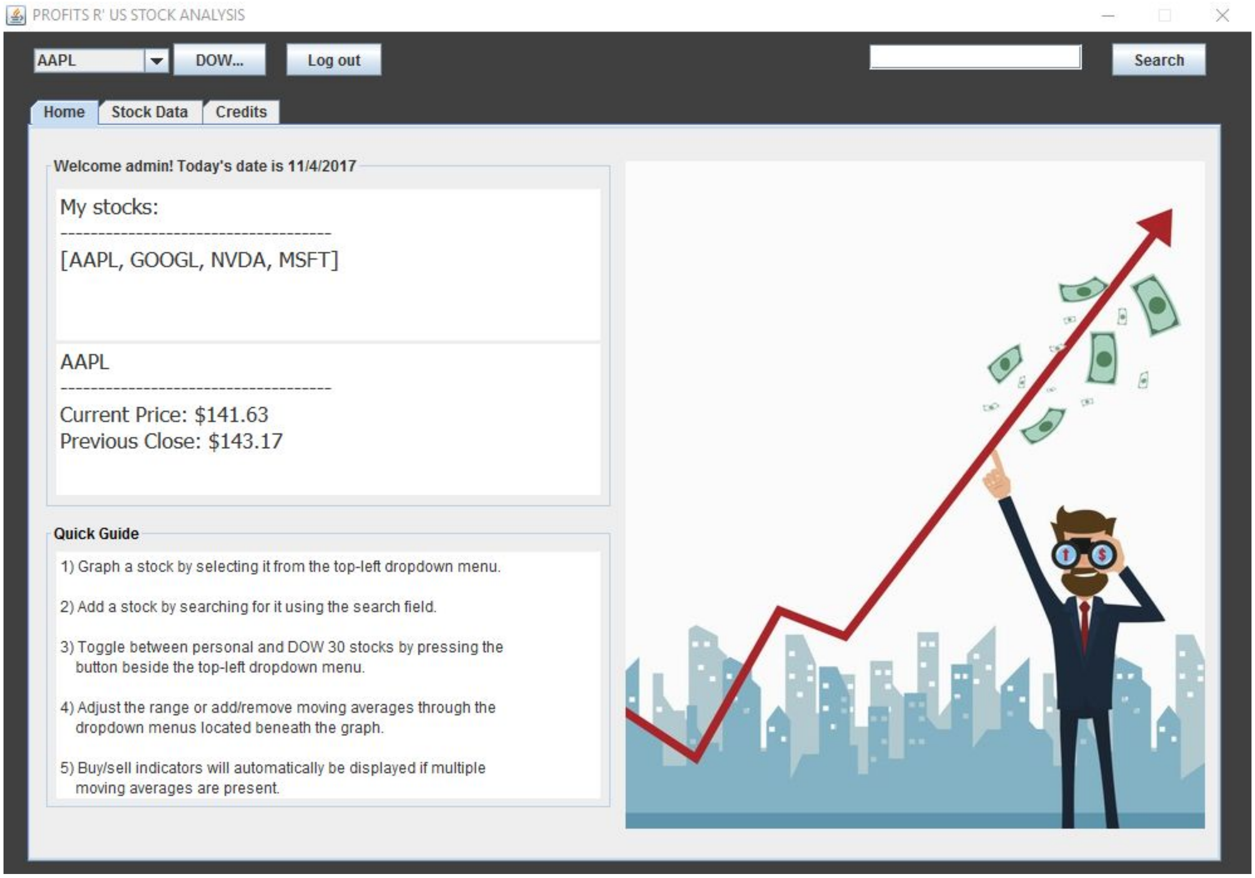
3.2.1 Use Case Specification

Number	1	
Name	Login	
Summary	User logs into the system	
Priority	High	
Preconditions	User created a username and password	
Postconditions	User can access stock application features	
Primary Actor(s)	ProfitsRUS Employee	
Secondary Actor(s)	Yahoo Finance Database	
Trigger	User has started the program	
Main Scenario	**Step**	**Action**
	1	User begins the program
	2	System asks for username and password
	3	User enters username and password
	4	System checks if username exists and given password is correct.
	5	User waits for system to respond
	6	Systems approves and displays main UI for program
Extensions	**Step**	**Action**
	4a	System notifies user that username or password is incorrect
	4b	System asks user to re-enter information
Open Issues	1	Should the system ask if the user wants to create an account?

3.2.2 Scenario

Once the program is started, the user will be prompted to login by entering a username and password in the corresponding text fields. The credentials will be verified, and if the username exists and matches with the associated password, the user will be provided access to the main program. The main program GUI will then be displayed, as seen in the images below.





Welcome admin! Today's date is 11/4/2017

My stocks:

[AAPL, GOOGL, NVDA, MSFT]

AAPL

Current Price: $141.63
Previous Close: $143.17

1) Graph a stock by selecting it from the top-left dropdown menu.

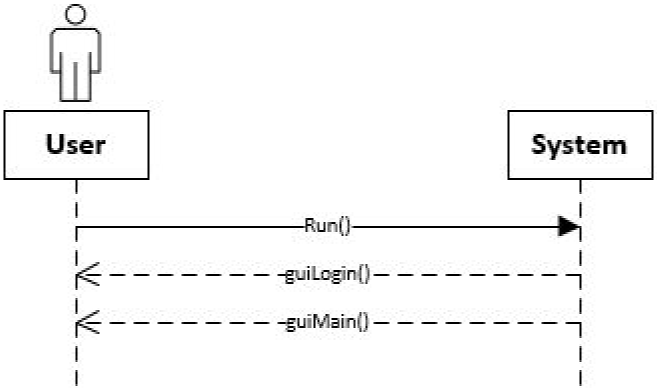
2) Add a stock by searching for it using the search field.

3) Toggle between personal and DOW 30 stocks by pressing the button beside the top-left dropdown menu.

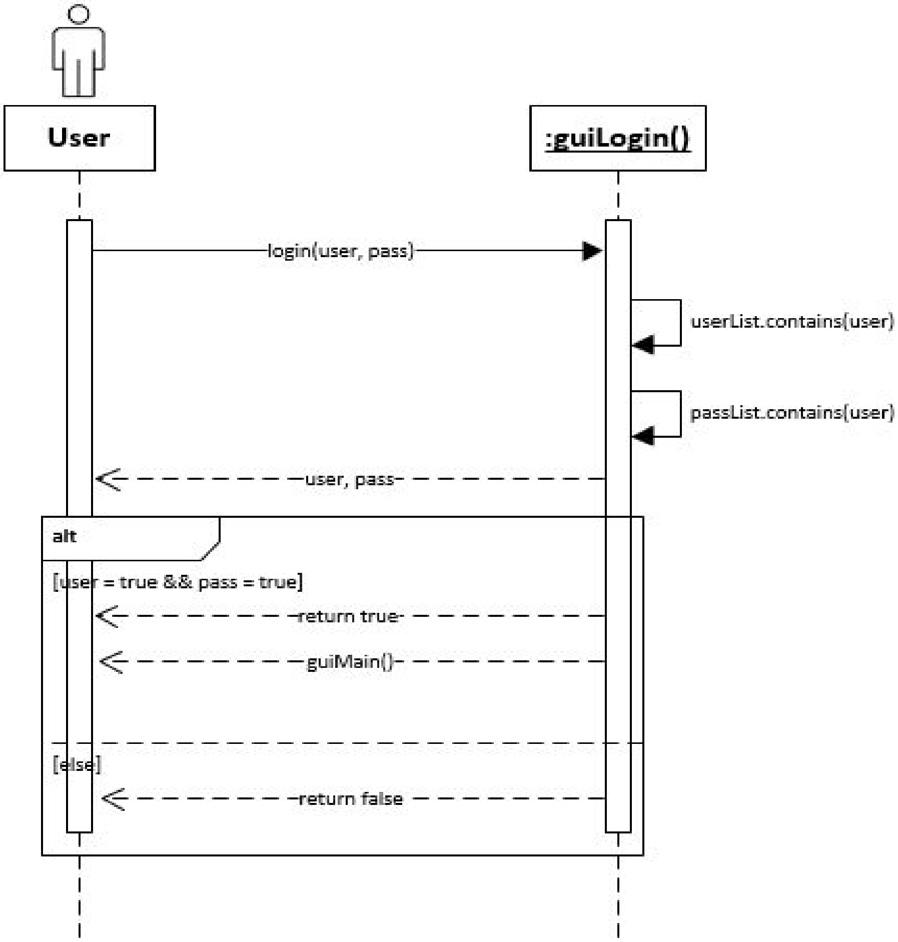
4) Adjust the range or add/remove moving averages through the dropdown menus located beneath the graph.

5) Buy/sell indicators will automatically be displayed if multiple moving averages are present.

3.2.3 System Sequence Diagram



3.2.4 Sequence Diagram



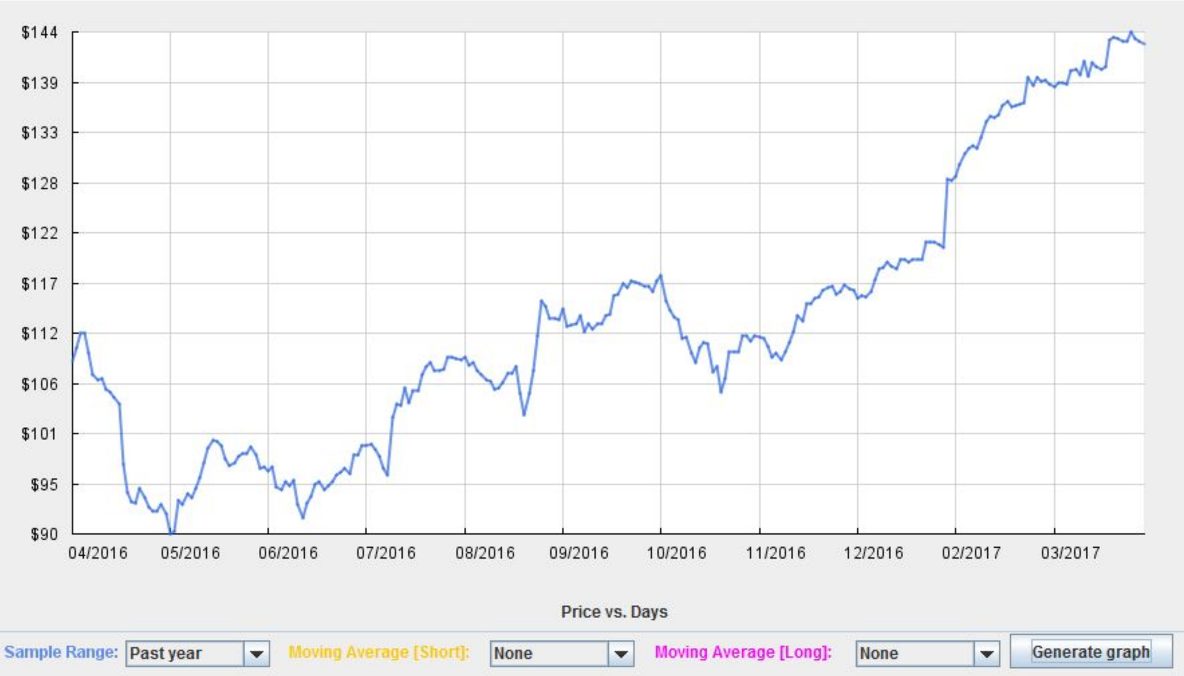
3.3 Add/Remove Moving Average

3.3.1 Use Case Specification

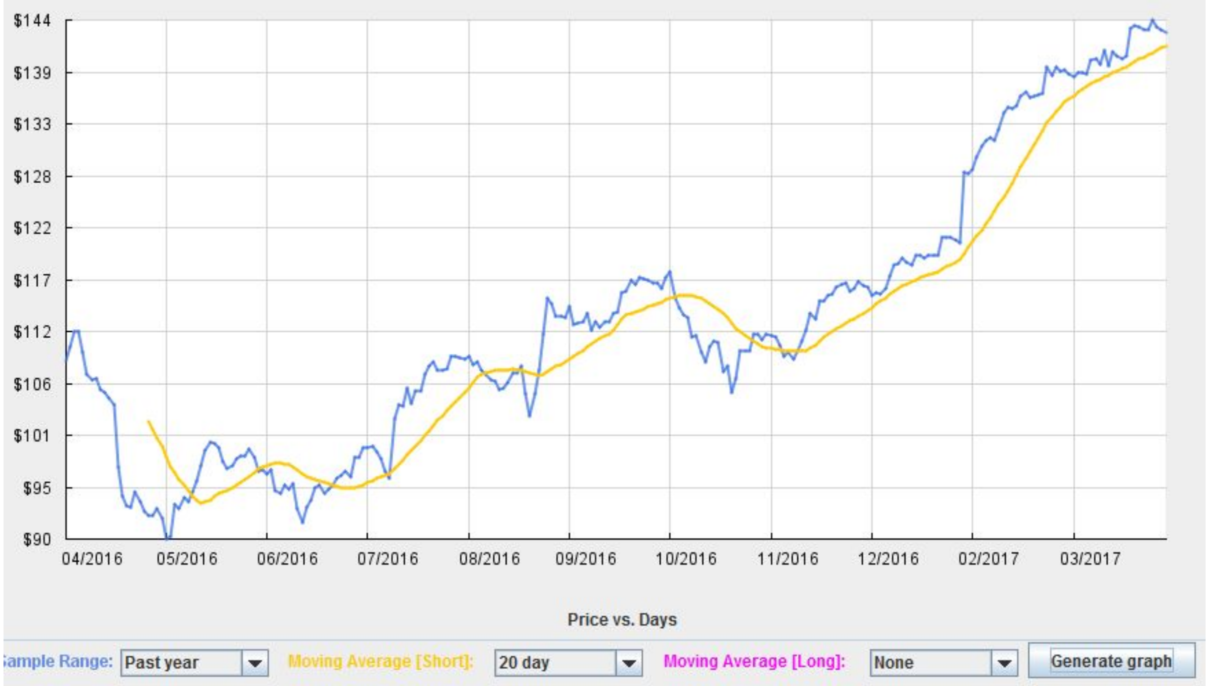
Number	2	
Name	Add/Remove Moving Average	
Summary	User can add or remove one or both of the Moving Averages (short/long)	
Priority	Medium	
Preconditions	User is logged in and selected Stock Data	
Postconditions	User adds or removes selected stock moving average(s)	
Primary Actor(s)	ProfitsRUS Employee	
Secondary Actor(s)	User Database	
Trigger	User has selected Moving Average values	
Main Scenario	**Step**	**Action**
	1	User selects "Stock Data" tab
	2	User selects sample range (default: all)
	3	User may add Short term Moving Average (20 or 50 days) or remove Short term Moving average (select "none")
	4	User may add Long term Moving Average (100 or 200 days) or remove Long term Moving average (select "none")
	5	User presses "Generate Graph" button
	6	System displays graph
	7	Repeat step 1-6 (Optional)

3.3.2 Scenario

After selecting Stock Data, user chooses range of data (default: all) and selects from the drop down menu each value for short term moving average (20/50 days) and long (100/200 days) or “none” to remove an existing moving average function. After pressing *Generate graph*, the System displays the graph accordingly and changes may be re-entered.

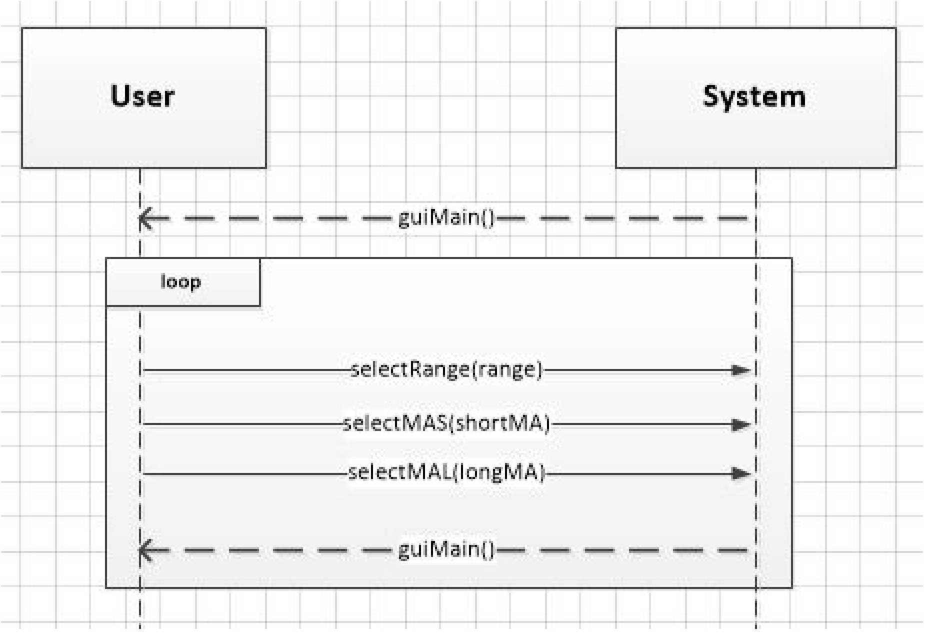


Sample Range: Past year Moving Average [Short]: None Moving Average [Long]: None

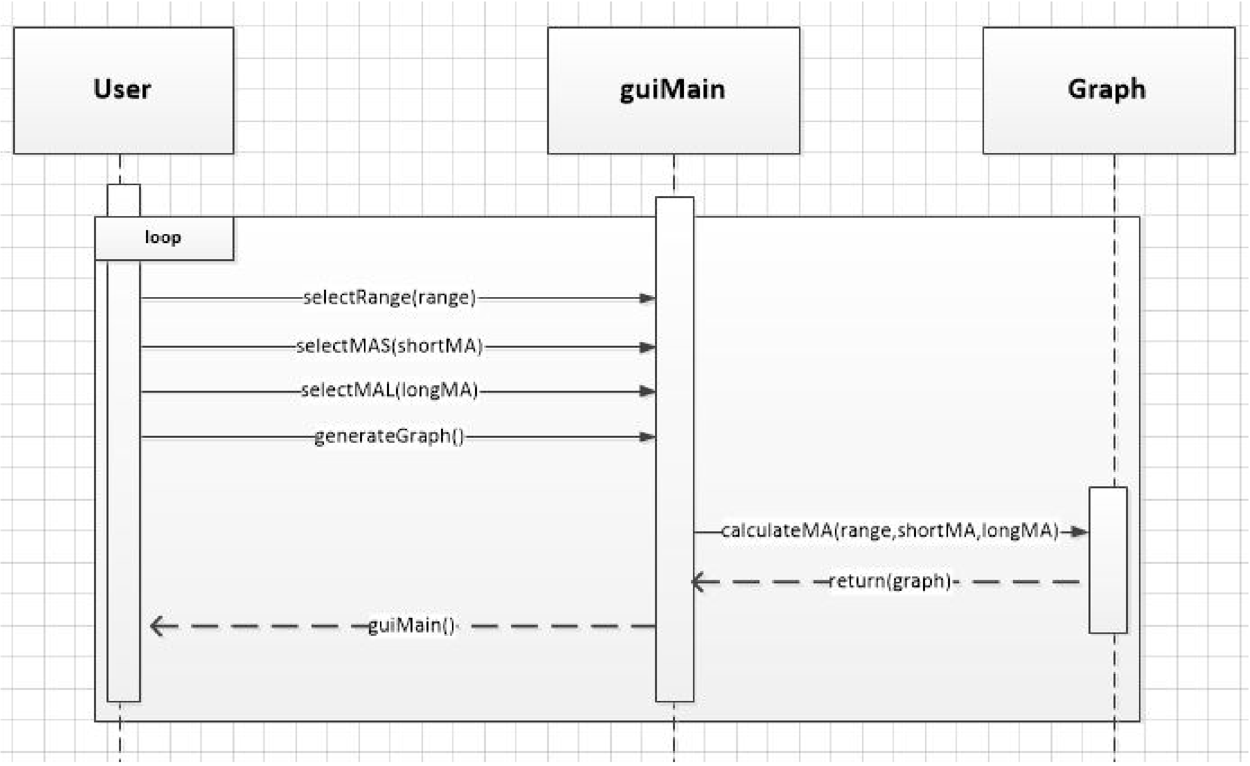


Sample Range: Past year Moving Average [Short]: 20 day Moving Average [Long]: None

3.3.3 System Sequence Diagram



3.3.4 Sequence Diagram



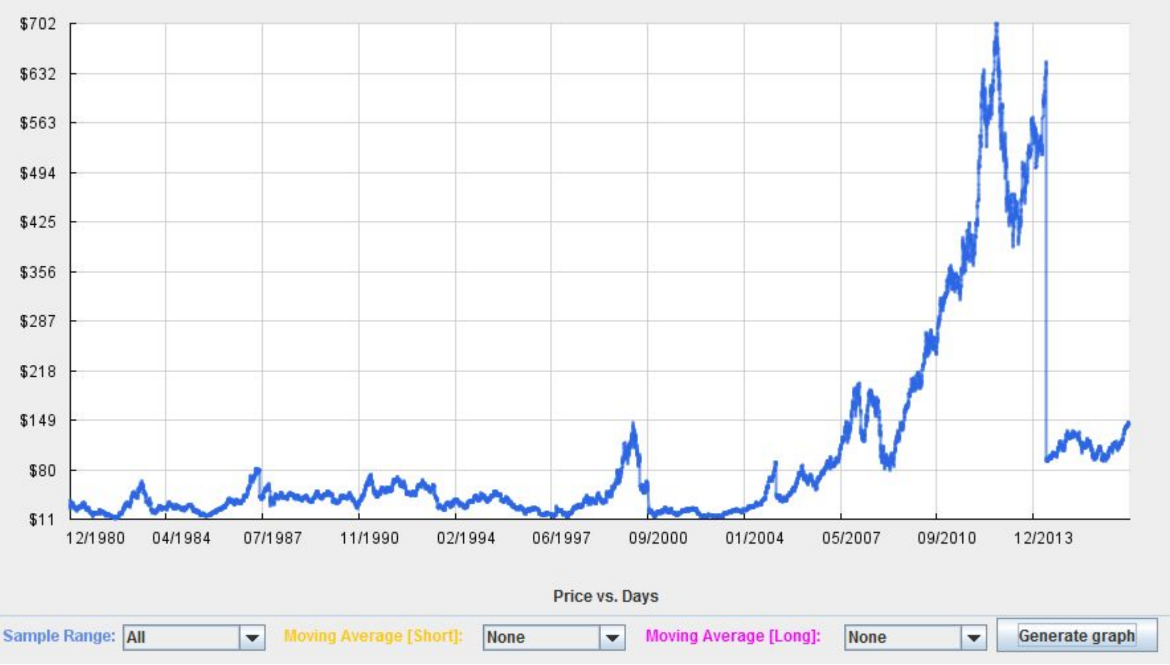
3.4 Change Range

3.4.1 Use Case Specification

Number	3	
Name	Change Range	
Summary	User selects a range of display options (All Data, Past Year, Past 2 Years, Past 5 Years)	
Priority	Medium	
Preconditions	User is logged in and selected Stock Data	
Postconditions	User changes range of graph	
Primary Actor(s)	ProfitsRUS Employee	
Secondary Actor(s)	User database	
Trigger	User has selected range value	
Main Scenario	**Step**	**Action**
	1	User selects “Stock Data” tab
	2	User selects sample range (All, past year, 2 years or 5 years)
	3	User may add Short term Moving Average (20 or 50 days) or remove Short term Moving average (select “none”)
	4	User may add Long term Moving Average (100 or 200 days) or remove Long term Moving average (select “none”)
	5	User presses “Generate Graph” button
	6	System displays graph
	7	Repeat step 1-6 (Optional)

3.4.2 Change Range Scenario

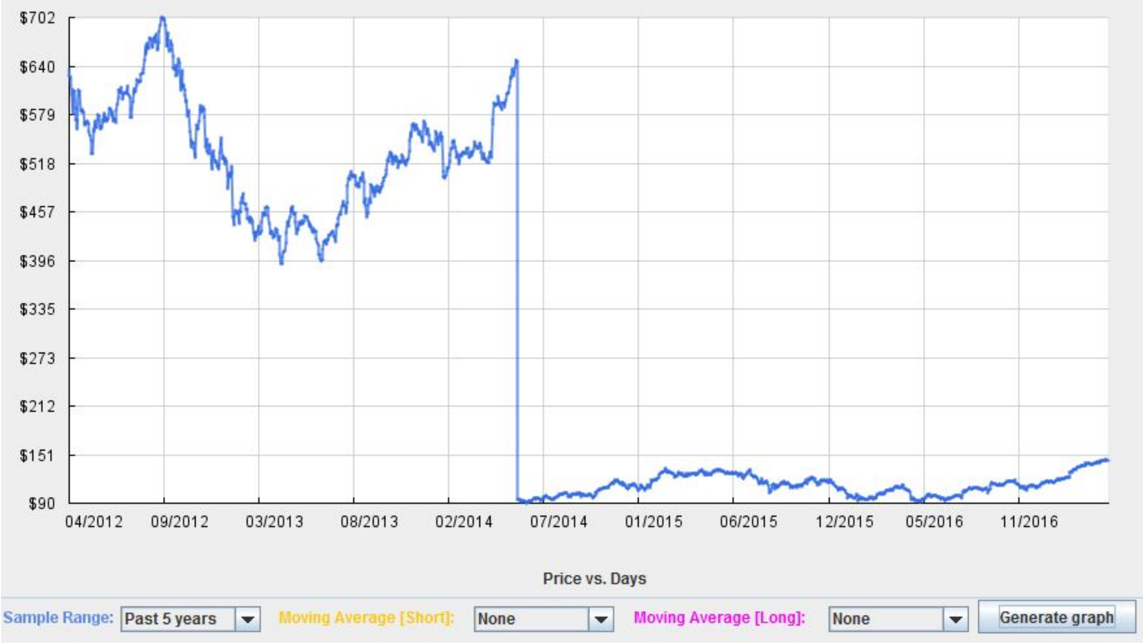
After selecting Stock Data, user chooses range of data (default: all) or changes the range (all, past year, 2 years or 5 years) if already set. User then selects from the drop down menu each value for short term moving average (none//20/50 days) and long (none/100/200 days). After pressing *Generate graph*, the System displays the graph accordingly and changes may be re-entered.



Sample Range: All

Moving Average [Short]: None

Moving Average [Long]: None

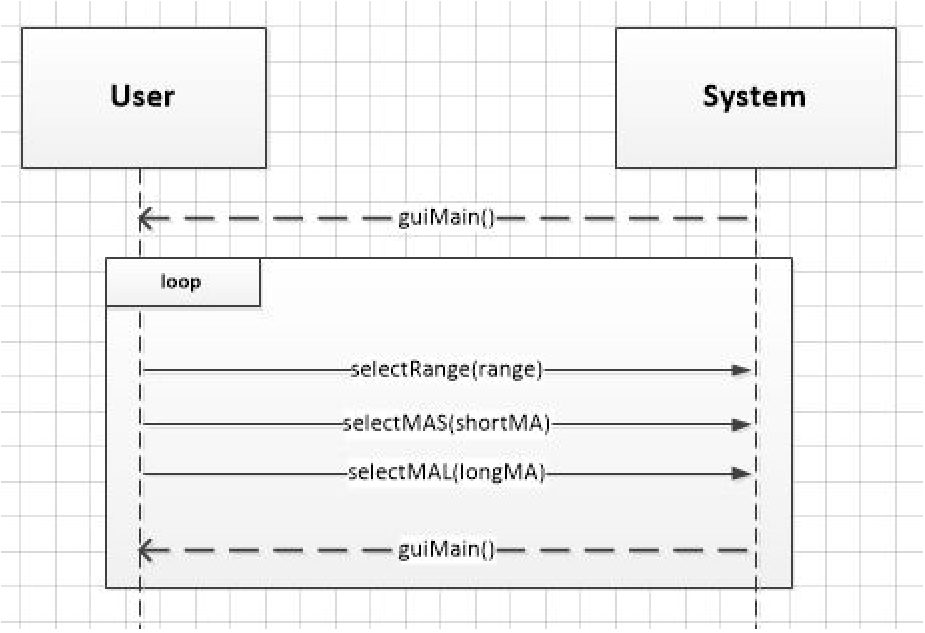


Sample Range: Past 5 years

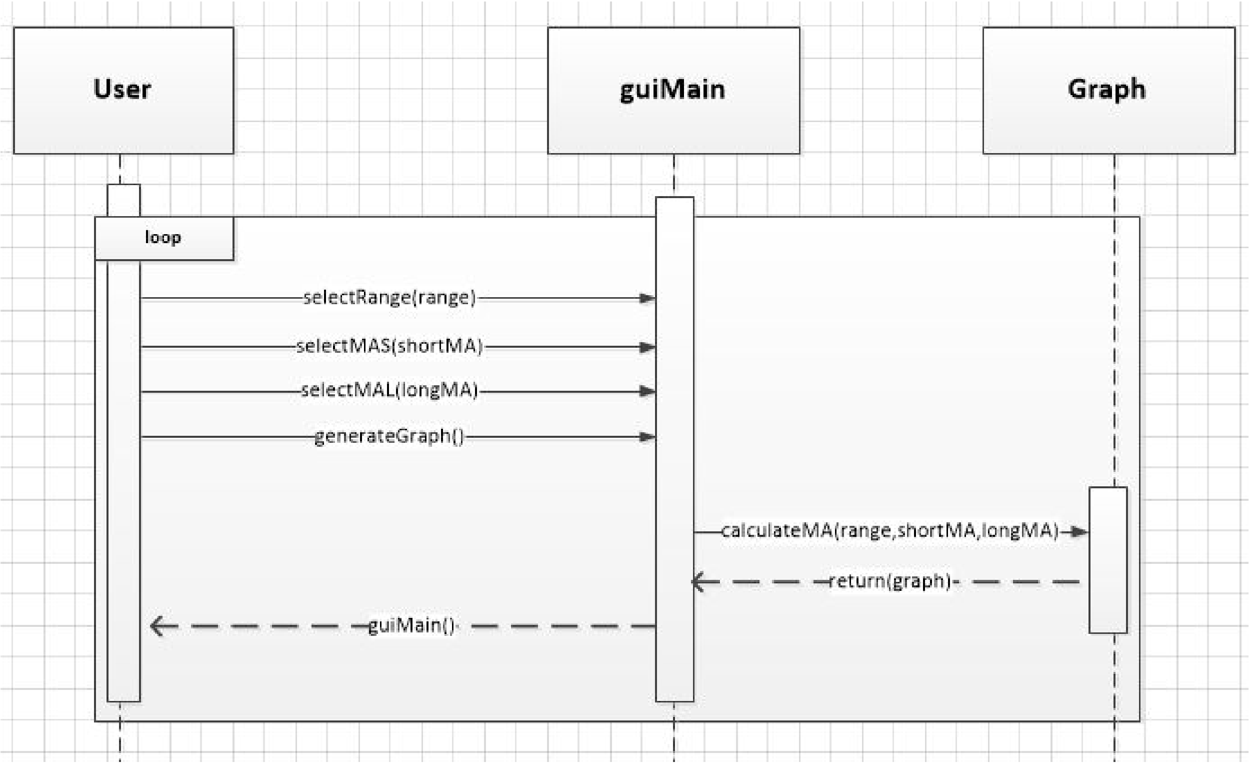
Moving Average [Short]: None

Moving Average [Long]: None

3.4.3 System Sequence Diagram



3.4.4 Sequence Diagram



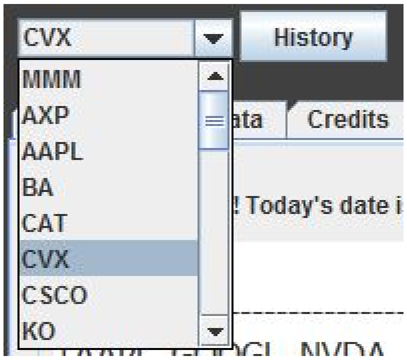
3.5 Graph Stock

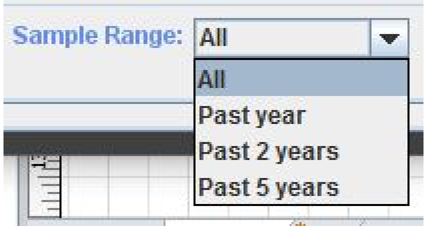
3.5.1 Use Case Specification

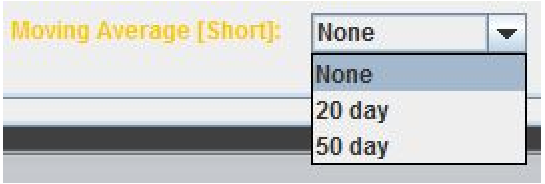
Number	4	
Name	Graph Stock	
Summary	Prints the graphical output for selected stock (from history or DOW 30)	
Priority	High	
Preconditions	User is logged in	
Postconditions	User can see graphs of selected stock closing day (including range)	
Primary Actor(s)	ProfitsRUS Employee	
Secondary Actor(s)	User Database	
Trigger	User presses “Generate graph” button	
Main Scenario	**Step**	**Action**
	1	User selects stock type (from History or from DOW30)
	2	User selects stock
	3	User selects “Stock Data” tab
	4	User selects sample range (default: all)
	5	User selects Short term Moving Average (default: none)
	6	User selects Long term Moving Average (default: none)
	7	User presses “Generate Graph” button
	8	System displays graph
	9	Repeat step 1-8 (Optional)
Extention	**Step**	**Action**
	7a	Stock has not been selected
	7b	Return to step 2

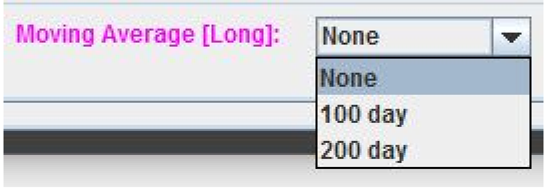
3.5.2 Graph Stock Scenario

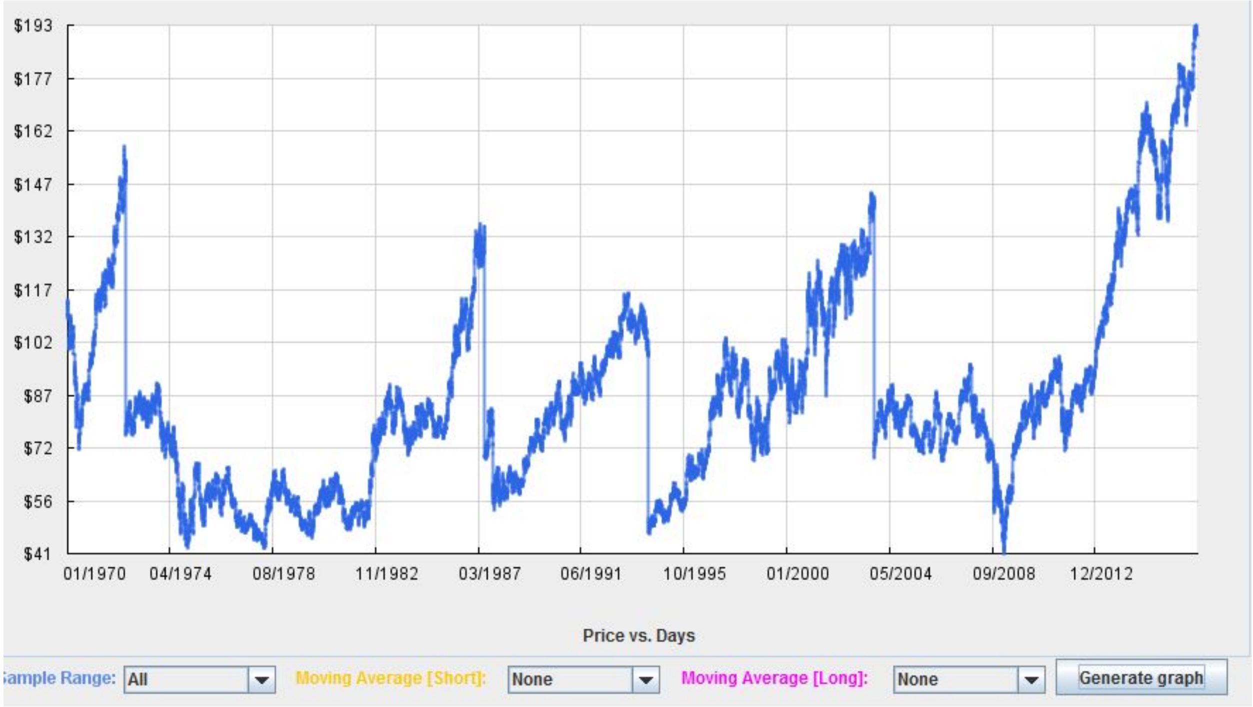
After successfully logging in, the user may select a stock either directly from the DOW30 option or through their history using the drop down scroll. From here, user may select the sample range, moving average (short) and moving average (long) settings or otherwise keep their default values. User then presses “Generate graph” button to view the graph displayed by the system.


CVX


Sample Range: All

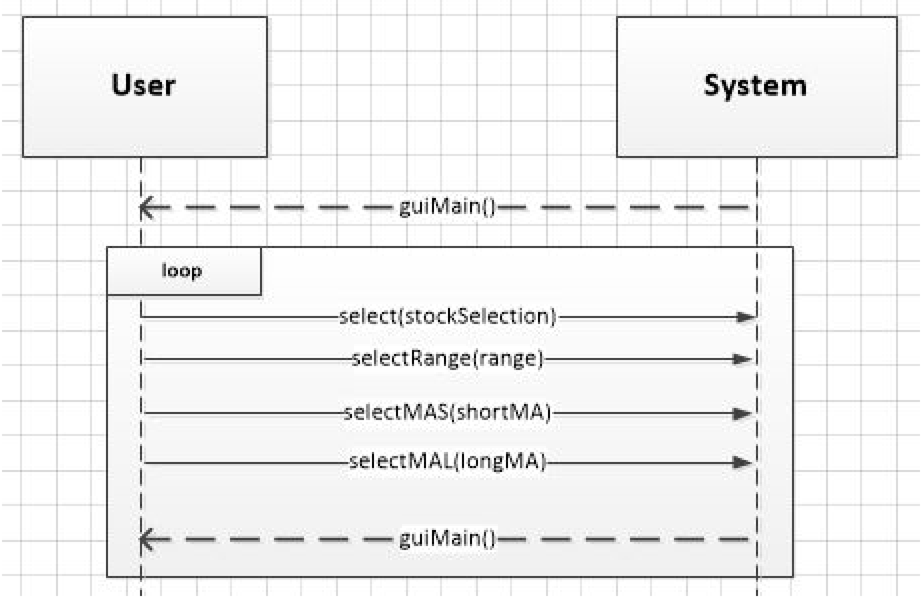

Moving Average [Short]: None


Moving Average [Long]: None

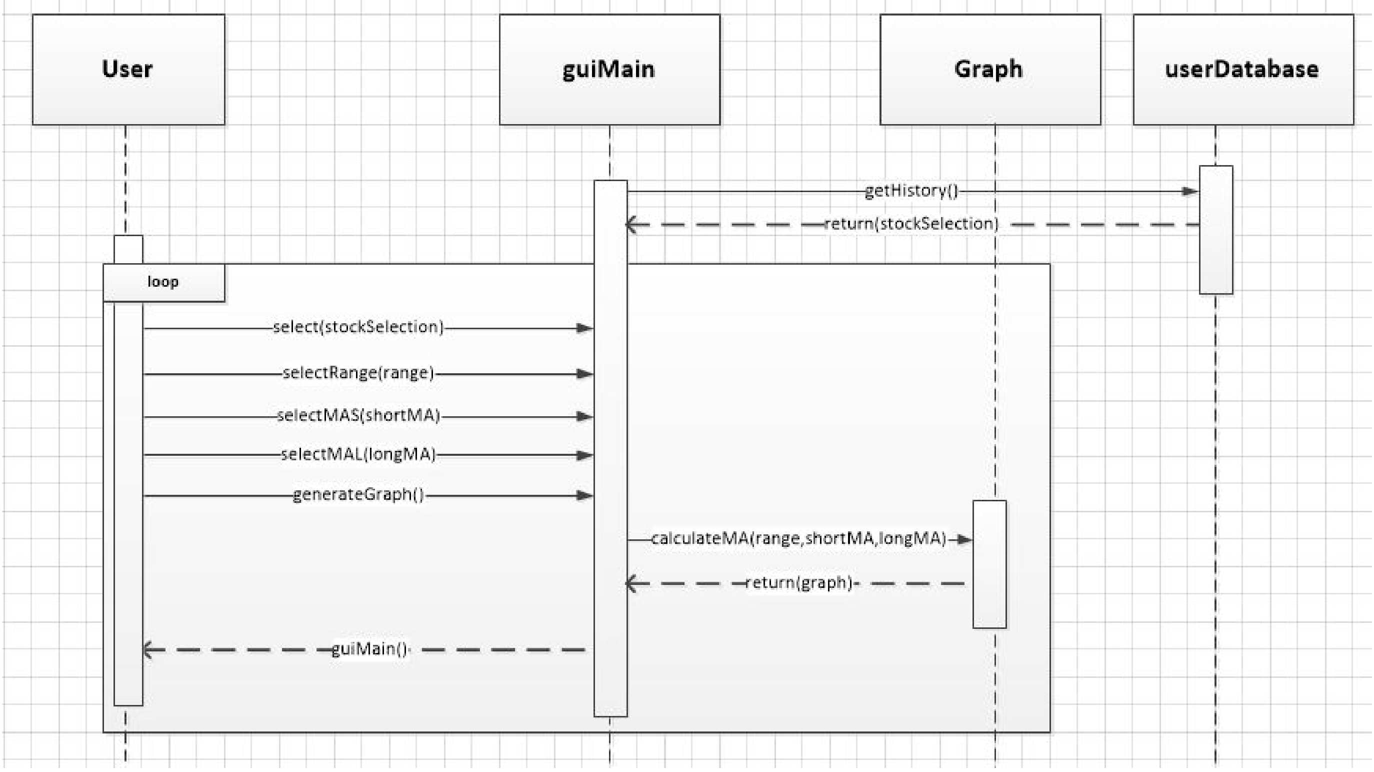


Sample Range: All Moving Average [Short]: None Moving Average [Long]: None

3.5.3 System Sequence Diagram



3.5.4 Sequence Diagram



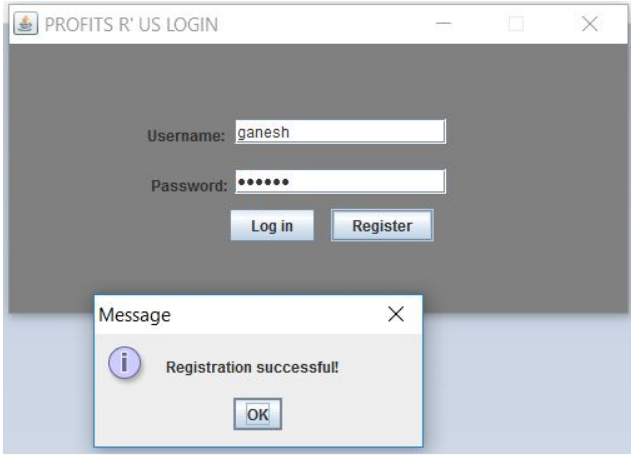
3.6 Create New User

3.6.1 Use Case Specification

Number	5	
Name	Register New User	
Summary	A new user is registered and can access all the functionalities in the application.	
Priority	High	
Preconditions	User should enter his user id and password and click on register .	
Postconditions	User automatically logins once he registered successfully.	
Primary Actor(s)	ProfitsRUS Employee	
Secondary Actor(s)	User Database	
Trigger	User has selected Register	
Main Scenario	**Step**	**Action**
	1	User enters username
	2	User enters Password
	3	User presses “Register” button
	4	User can see a welcoming message.
	5	User clicks on “OK” button.
	6	Registered user gets logged in.
Extention	**Step**	**Action**
	3a	User has entered an existing username
	3b	Return to step 1.

3.6.2 New user Scenario

Once the program is launched, the user will be prompted to login by entering a username and password in the corresponding text fields. The credentials will be verified, and if the username exists and matches with the associated password, the user will be provided access to the main program. If the user name does not exists then the user should register using register with unique user id and password one the user id is allowed then the program displays the main GUI .



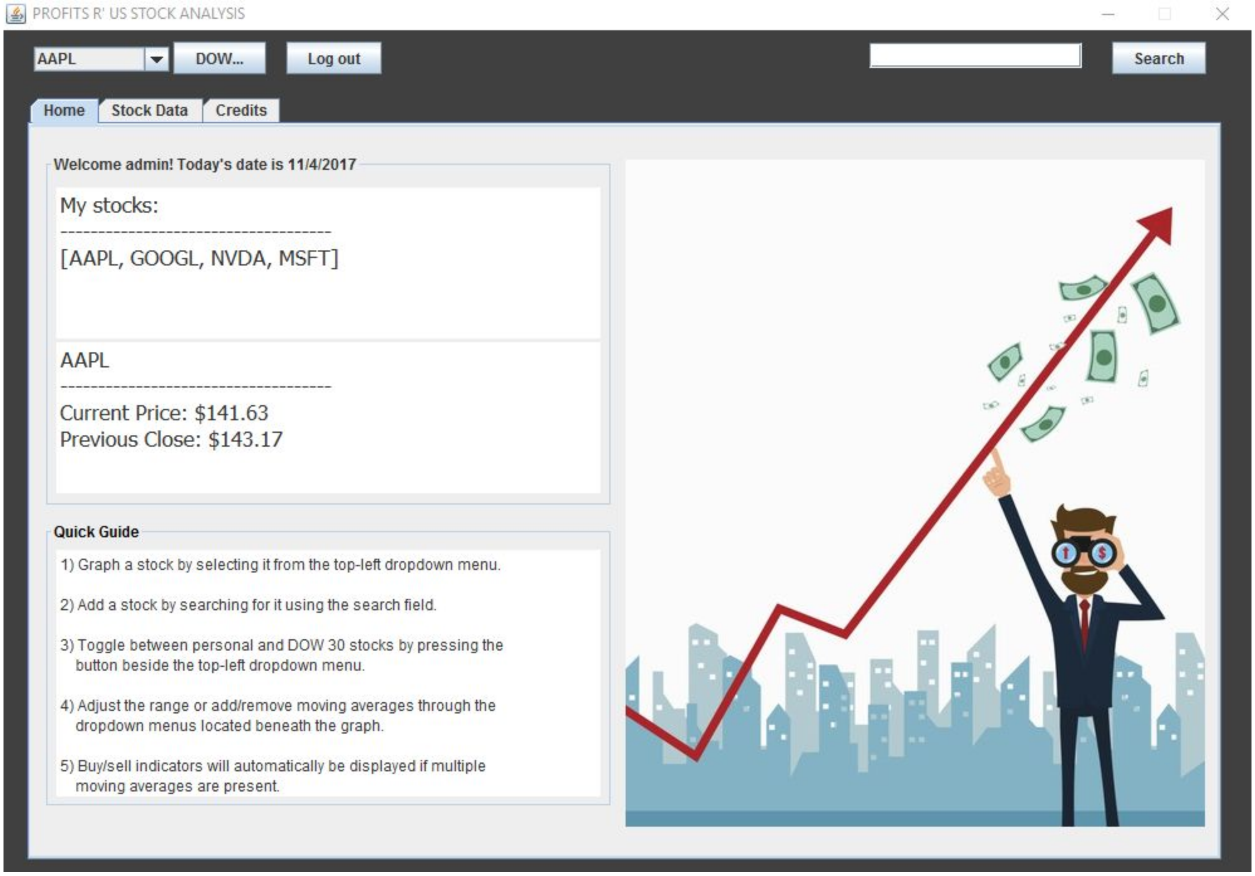
PROFITS R' US LOGIN

Username: ganesh

Password: ••••••

Message

Registration successful!



PROFITS R' US STOCK ANALYSIS

Welcome admin! Today's date is 11/4/2017

My stocks:

-------------------------------------

[AAPL, GOOGL, NVDA, MSFT]

AAPL

-------------------------------------

Current Price: $141.63
Previous Close: $143.17

Quick Guide

1) Graph a stock by selecting it from the top-left dropdown menu.

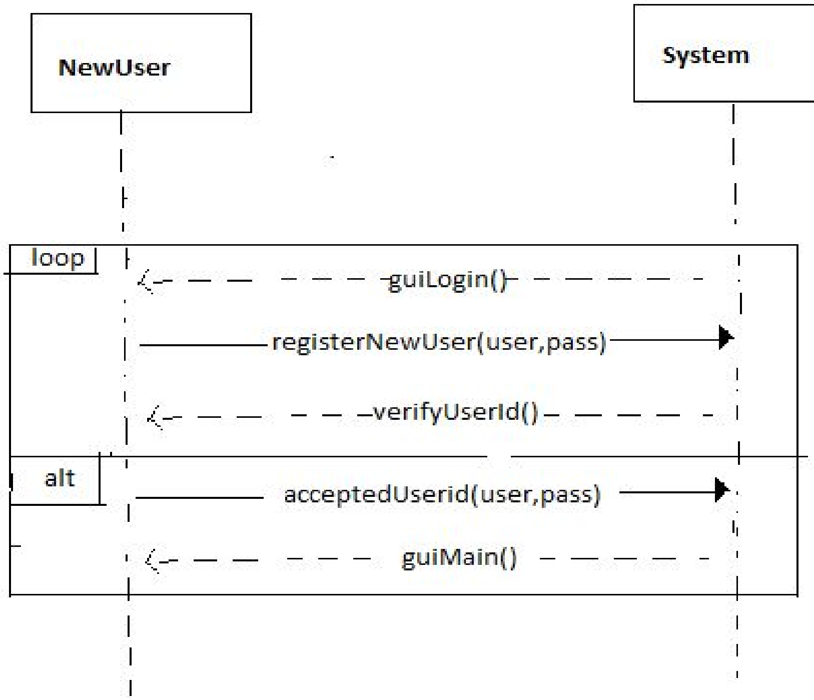
2) Add a stock by searching for it using the search field.

3) Toggle between personal and DOW 30 stocks by pressing the button beside the top-left dropdown menu.

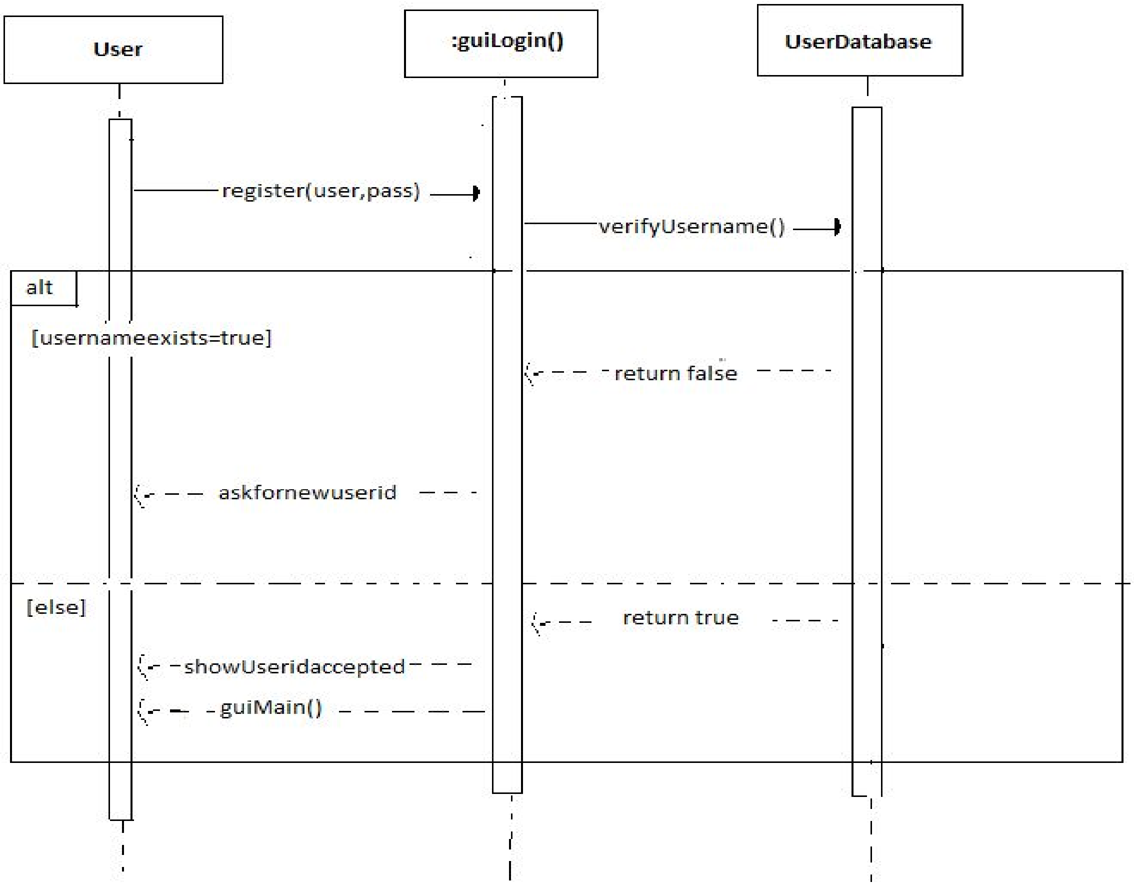
4) Adjust the range or add/remove moving averages through the dropdown menus located beneath the graph.

5) Buy/sell indicators will automatically be displayed if multiple moving averages are present.

3.6.3 System Sequence Diagram



3.6.4 Sequence Diagram

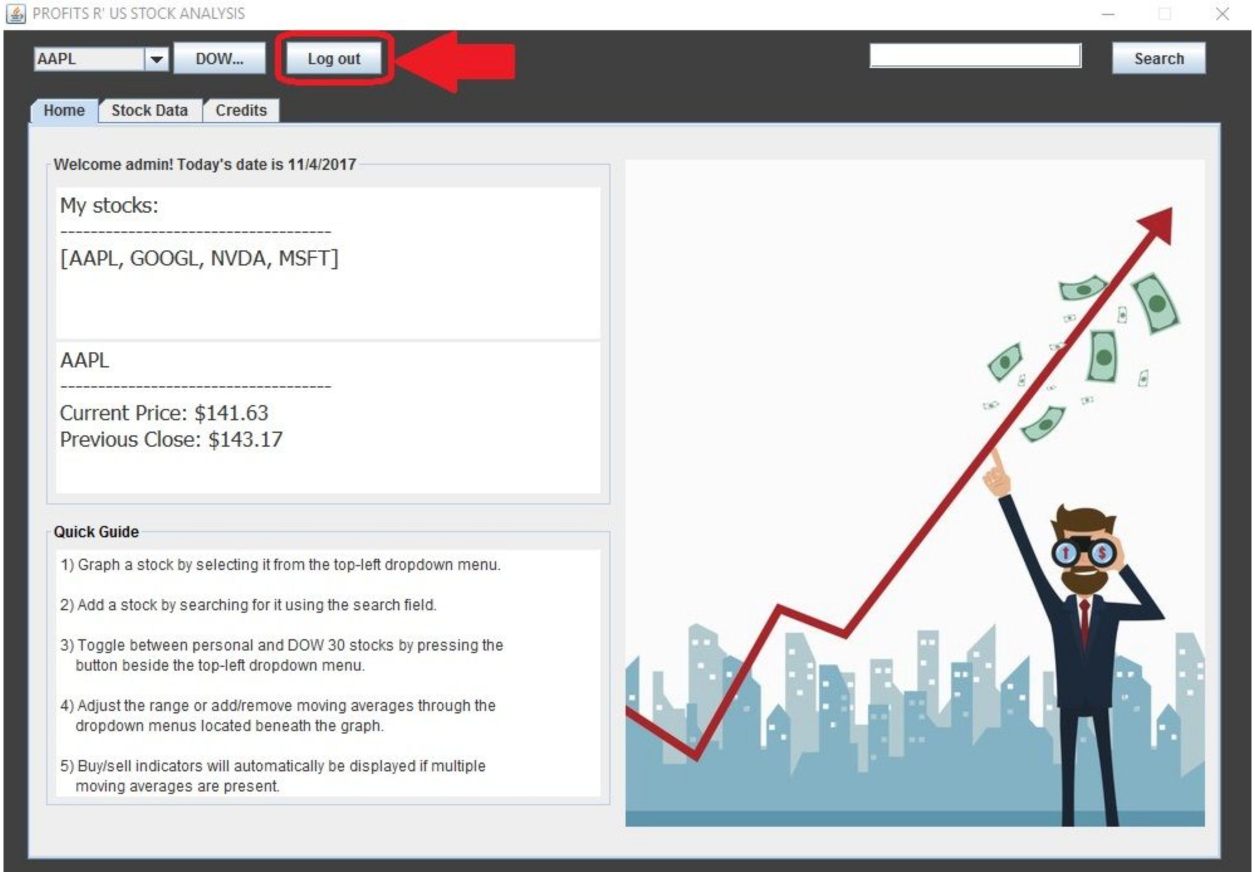


3.7 Logging Out

3.7.1 Use Case Specification

Number	6	
Name	Log Out	
Summary	The user is logged out of his account.	
Priority	Low	
Preconditions	User is already logged in.	
Postconditions	The user is logged out.	
Primary Actor(s)	The User	
Secondary Actor(s)	User Database	
Trigger	User Selects the Log Out button	
Main Scenario	**Step**	**Action**
	1	The user is already logged in.
	2	The user selects the Log out button.
	3	The user is logged out.
	4	The user is returned to the logging screen
Extention	**Step**	**Action**
	3a	User is already logged out.

3.7.2 Logging Out Scenario



Welcome admin! Today's date is 11/4/2017

```
My stocks:
------------------------------------
[AAPL, GOOGL, NVDA, MSFT]

AAPL
------------------------------------
Current Price: $141.63
Previous Close: $143.17
```

Quick Guide

1) Graph a stock by selecting it from the top-left dropdown menu.

2) Add a stock by searching for it using the search field.

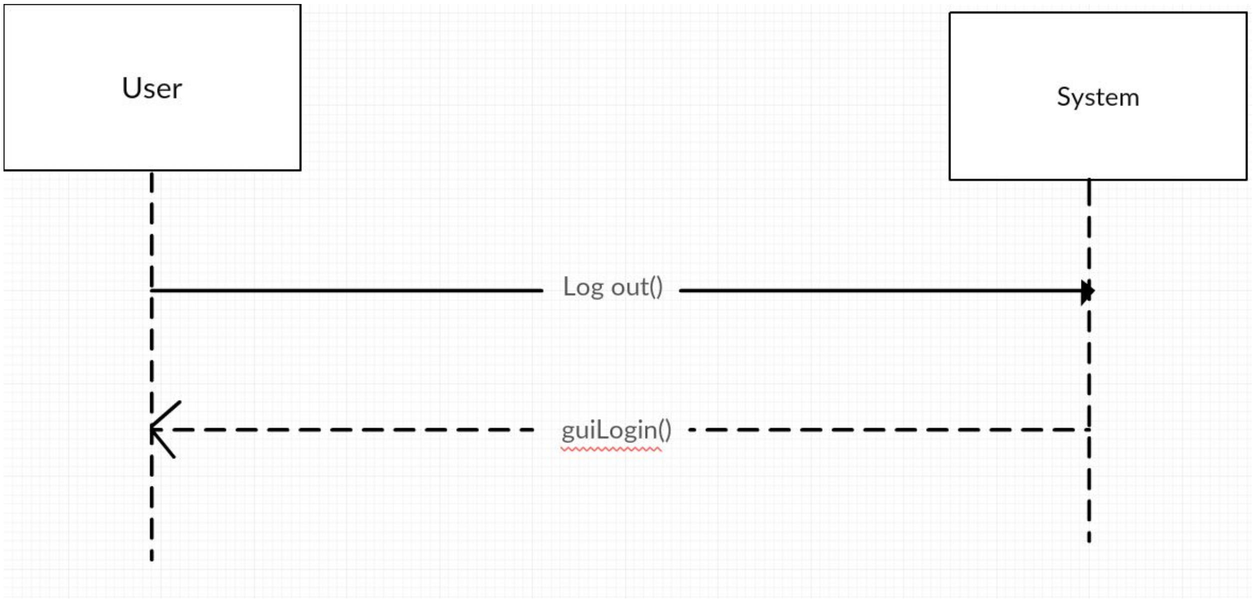
3) Toggle between personal and DOW 30 stocks by pressing the button beside the top-left dropdown menu.

4) Adjust the range or add/remove moving averages through the dropdown menus located beneath the graph.

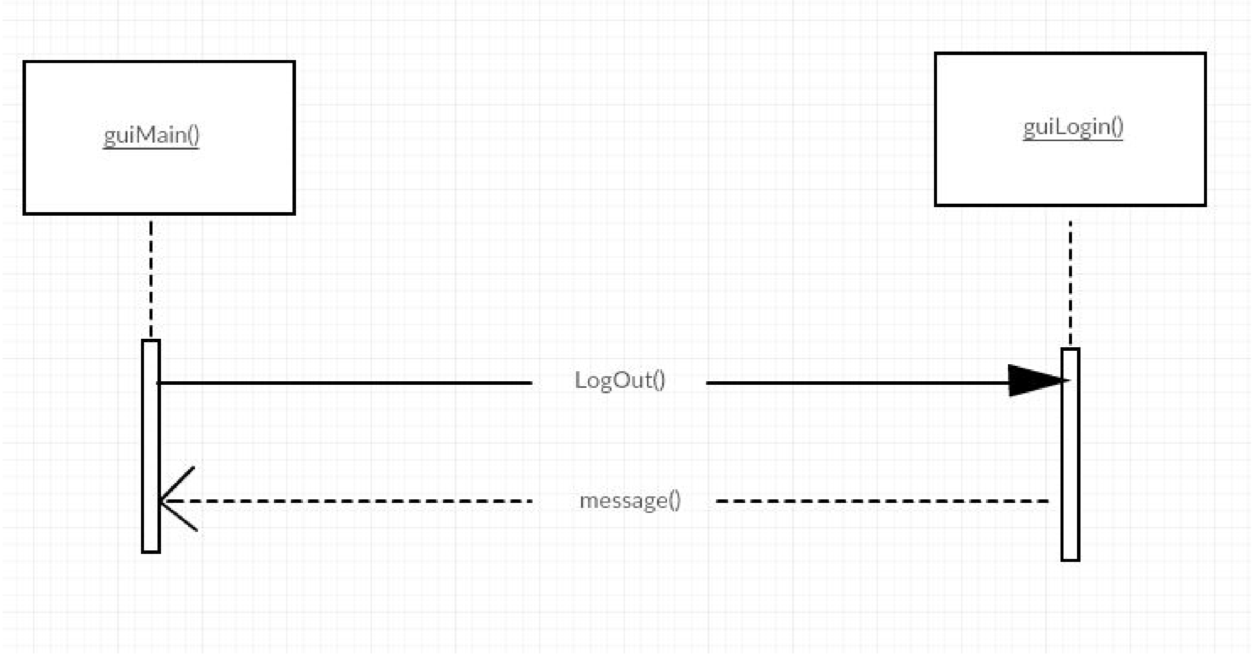
5) Buy/sell indicators will automatically be displayed if multiple moving averages are present.

When the user is done using the program and wishes to logout of their account they may do so by selecting the logout button which will bring them back to the logging screen.

3.7.3 System Sequence Diagram



3.7.4 Sequence Diagram



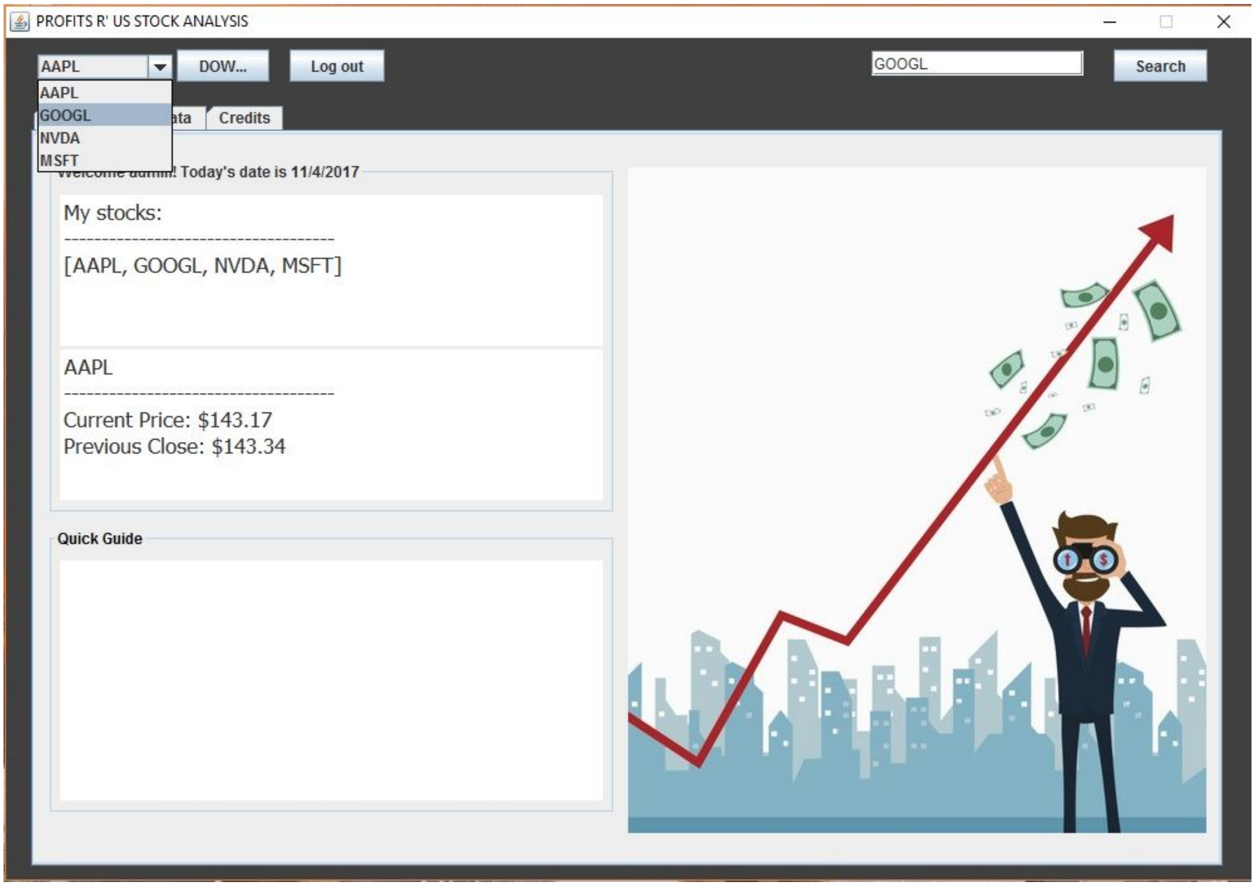
3.8 Search/Add Stock

3.8.1 Use Case Specification

Number	7	
Name	Search/Add Stock	
Summary	The user is searches or adds a stock to be viewed on the graph	
Priority	Medium	
Preconditions	User is already logged in.	
Postconditions	Stock is added to User's list of stocks.	
Primary Actor(s)	The User	
Secondary Actor(s)	Yahoo Finances	
Trigger	User Searches in the Search Bar and clicks Add	
Main Scenario	**Step**	**Action**
	1	System displays main window
	2	User types in search bar name of stock they want and click add
	3	System adds stock to the User's list of stocks
Extention	**Step**	**Action**
	2a	System notifies user that stock does not exist in Yahoo database
	2b	System asks user to input a new stock

3.8.2 Search/Add Stock Scenario

When the user has successfully logged in, they can now search for stocks via the search bar at the top right of the window. Once entered, the system searches for a matching stock of the same name within the Yahoo Finance database and if it finds one, adds that stock to the list of stocks the User has already searched and added.



! Today's date is 11/4/2017

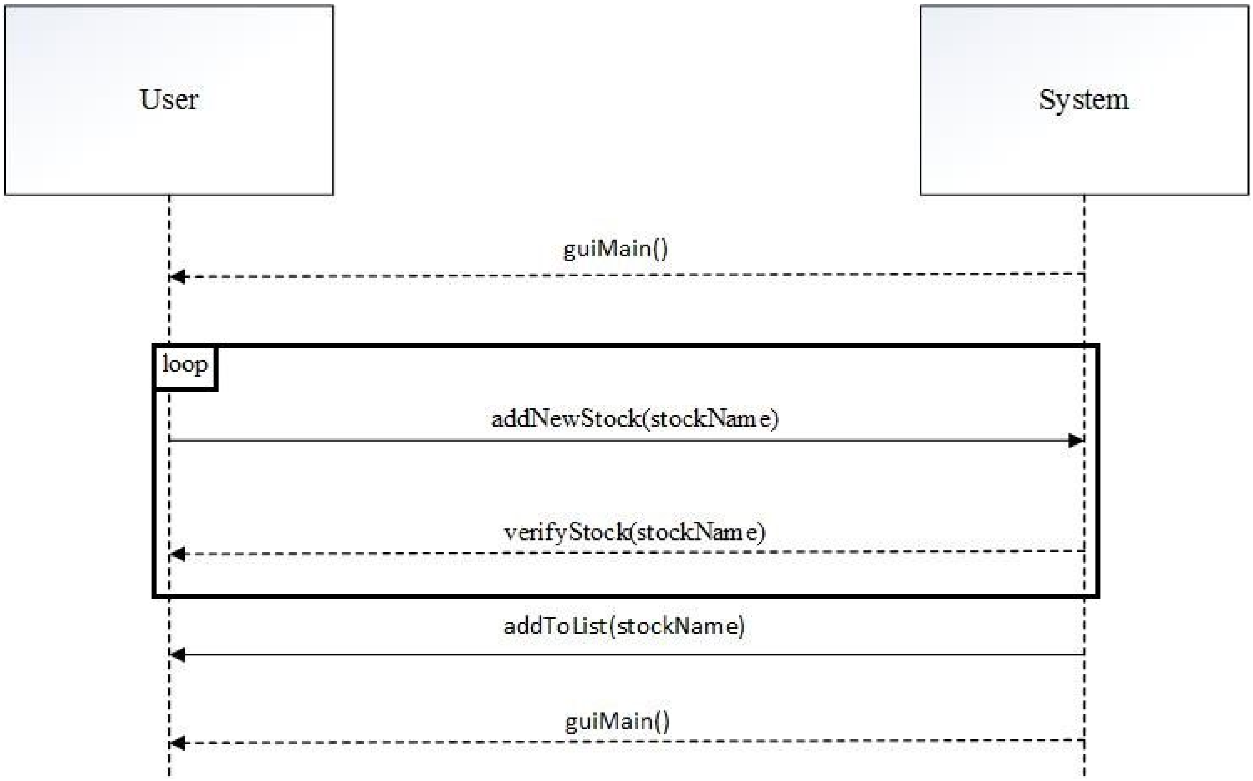
My stocks:

[AAPL, GOOGL, NVDA, MSFT]

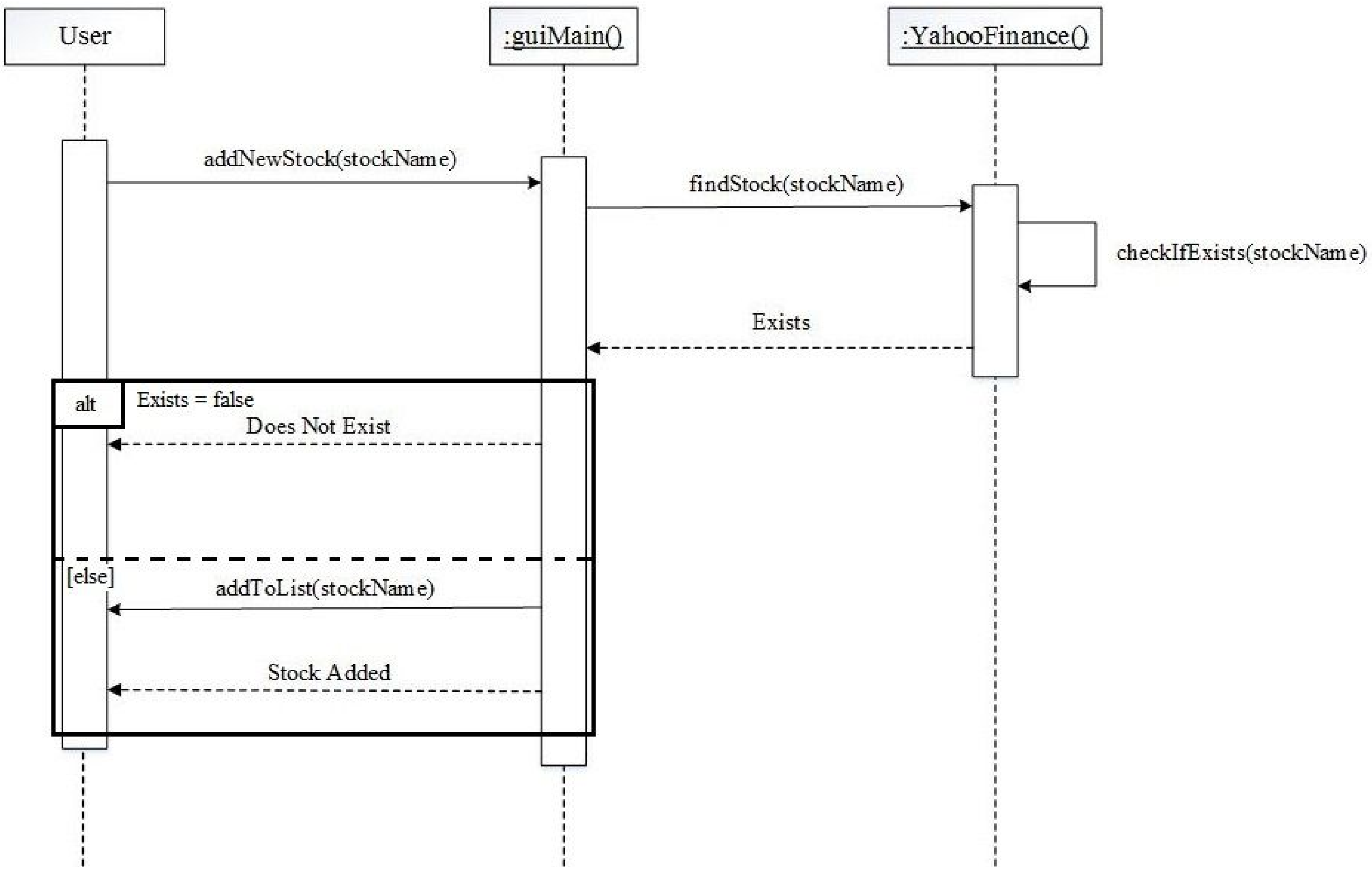
AAPL

Current Price: $143.17
Previous Close: $143.34

3.8.3 System Sequence Diagram



3.8.4 Sequence Diagram



4. System Architecture

4.1. Principle

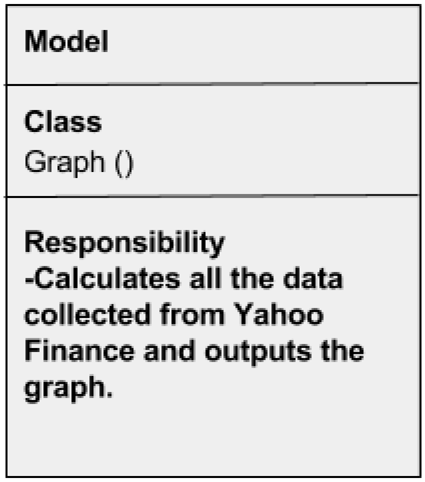
The Model-View-Controller architectural pattern (MVC) divides an interactive application into three components. Model contains the core functionality and data. Views display information to the user. Controllers handle user input. Views and controllers together comprise the user interface. A change-propagation mechanism ensures consistency between the user interface and the model.

In our project model is main logic which give access to registered users and gives requested data to GUI, which acts as an interface between user and data and controller will be the control functions that are given in the GUI at which user can choose different functionalities available.

In this project, we are using GUI which is especially prone to change requests. The user might request data multiple times on different options available like stock market history, moving averages of them based on time etc. They can access tabs for required information which gives us a clear idea that it acts as an interface between user and some fixed logic which is mode which will display requested data through controller.

Model

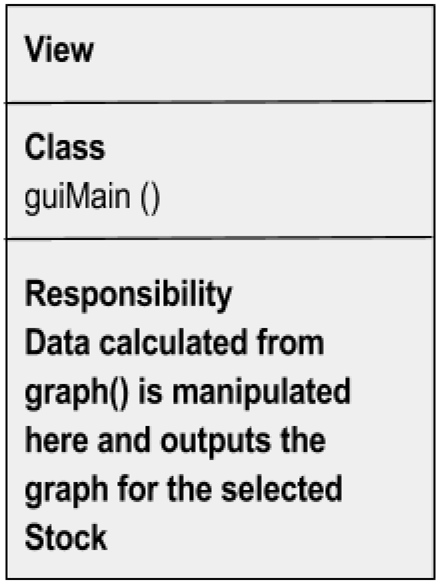
The model component contains the functional core of the application. It encapsulates the appropriate data, and exports procedures that perform application-specific processing. Controllers call these procedures on behalf of the user. The model also provides functions to access its data that are used by view components to acquire the data to be displayed.



Model
Class Graph ()
Responsibility **-Calculates all the data collected from Yahoo Finance and outputs the graph.**

View

View components display information to the user. A view obtains the data from the model. There can be multiple views of the model. Each view has an associated controller component. Controllers receive input, usually as events that encode mouse movement, activation of mouse buttons, or keyboard input. Events are translated to service requests for the model or the view. The user interacts with the system solely through controllers.



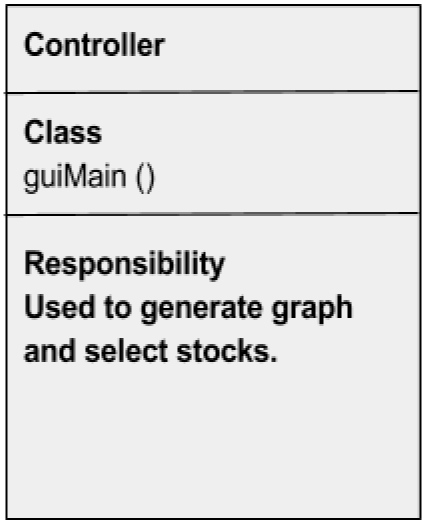
View

Class
guiMain ()

Responsibility
Data calculated from graph() is manipulated here and outputs the graph for the selected Stock

Controller

The controller components accept user input as events. How these events are delivered to a controller depends on the user interface platform. If the behavior of a controller depends on the state of the model, the controller registers itself with the change-propagation mechanism and implements an update procedure.

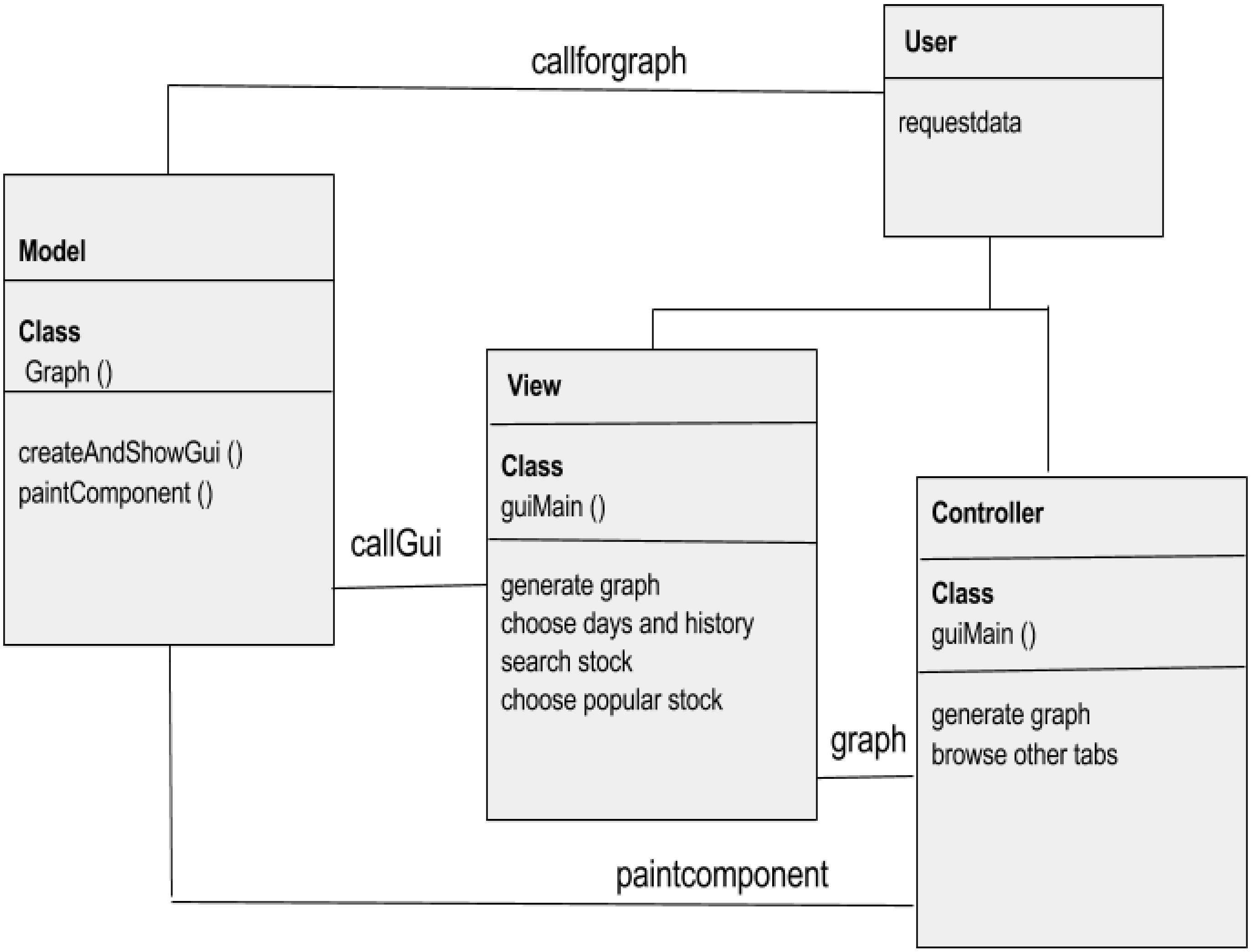


Controller

Class
guiMain ()

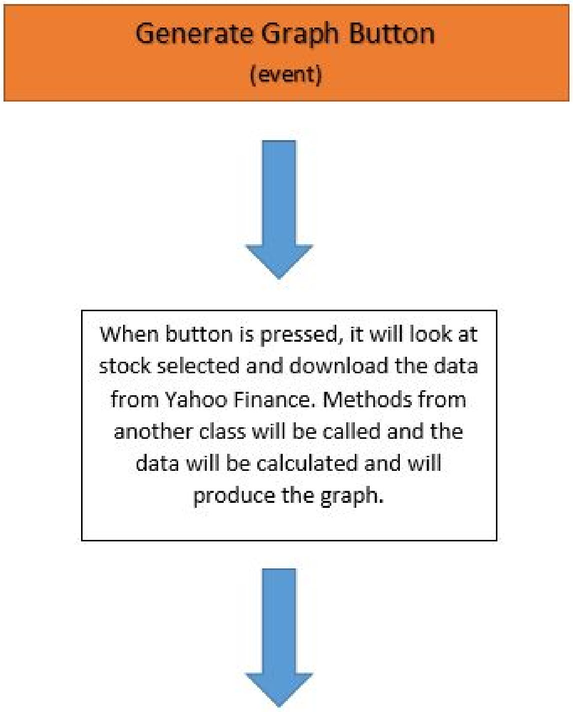
Responsibility
Used to generate graph and select stocks.

4.2. MVC Class Diagram



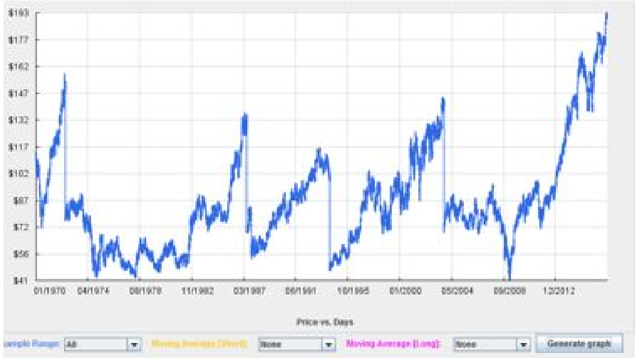
4.3 Secondary Architecture

Although the MVC architecture is heavily used in this application, the Event-Based architecture is also implemented. The button “Generate Graph” on the Stock Data tab is the event handler and signals the application to calculate the data collected from Yahoo Finance(an external repository of numbers). When the button is pressed, the data will be calculated and the graph shown is the result of the signaled event.



Generate Graph Button
(event)

When button is pressed, it will look at stock selected and download the data from Yahoo Finance. Methods from another class will be called and the data will be calculated and will produce the graph.



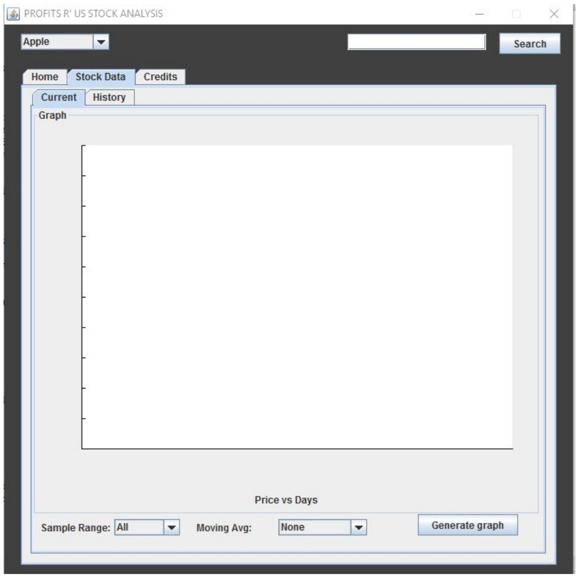
ample Range: All

Moving Average (Short): None

Moving Average (Long): None

5. External Interface Requirements

5.1. User Interfaces



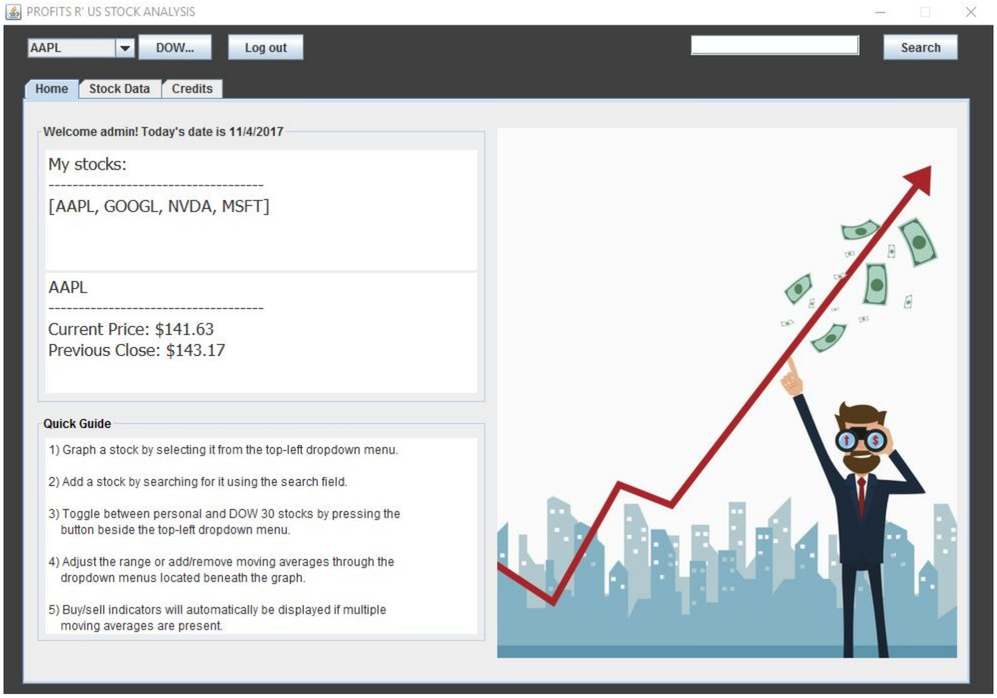
Sample Range: All

Moving Avg: None

Early version

The image above is the main tab the user will be on for most of the time the software is running. On this page the graph will be shown to the user.

Final Version



Welcome admin! Today's date is 11/4/2017

My stocks:

[AAPL, GOOGL, NVDA, MSFT]

AAPL

Current Price: $141.63
Previous Close: $143.17

Quick Guide

1) Graph a stock by selecting it from the top-left dropdown menu.

2) Add a stock by searching for it using the search field.

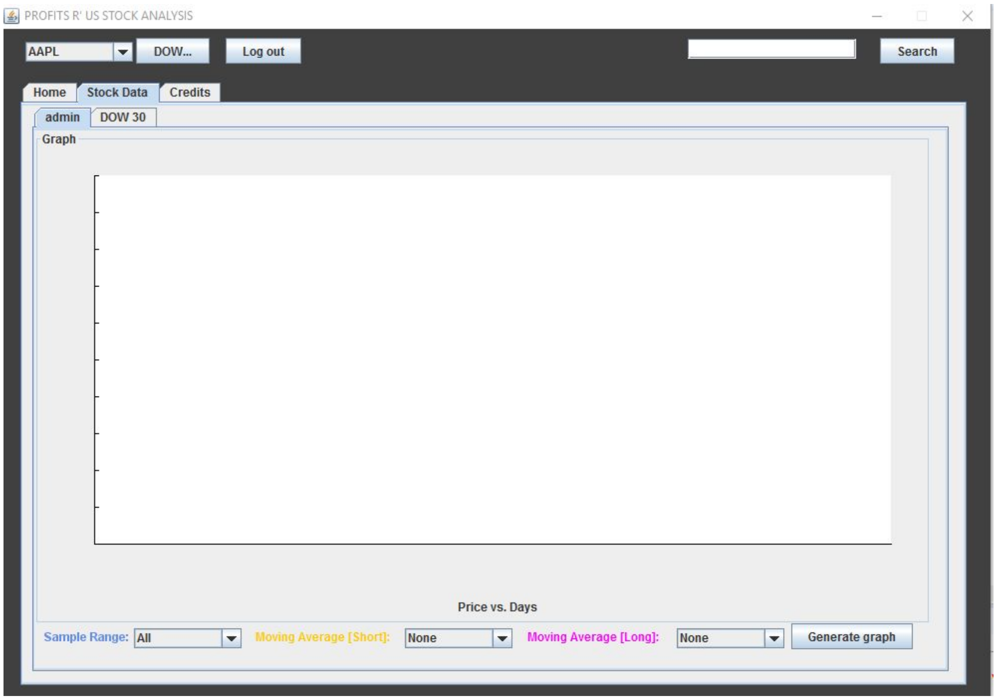
3) Toggle between personal and DOW 30 stocks by pressing the button beside the top-left dropdown menu.

4) Adjust the range or add/remove moving averages through the dropdown menus located beneath the graph.

5) Buy/sell indicators will automatically be displayed if multiple moving averages are present.

The image above is the home page that the user will be greeted with at the start. Here the application offers a quick guide to use the app, a list of previously viewed stocks and the current and previous prices of the stock selected.

From here the user can either visit the “Stock Data” tab or “Credits” tab.



Sample Range: All

Moving Average [Short]: None

Moving Average [Long]: None

The user will spend most of the application life time on this “Stock Data” tab. Here the selected stock in the drop down menu will be graphed when the button is hit. The user can then view the stock history, select various averages and determine whether to buy or sell a stock. If the user is confused about the DOW 30 acronyms they can select the DOW 30 tab and see visual pictures of the DOW 30

5.2 Hardware Interfaces

Computers running OS X, Windows or Linux will be able to run this application

5.3 Software Interfaces

Yahoo Finance API