Test Cases

Stockalytics

Derek Brown, Daniel Gaisberger, Alyssa Drohan, Horatio Hodge, Jarrett Horton, Phillip Gil-Perea, Isaiah Johnson, Russell Quao, Isaac Silvious

Table of Contents

Introduction…………………………………………………………………...pg 3
Constraints……………………………………………………………...pg 3
Test Environment & Tools……………………………………………..pg 3
Approach……………………………………………………………..pg 3
Roles…………………………………………………...………....pg 3
Schedule……………………………………………………….....pg 4
References…………………………………………………..……pg 4
Test Case ID TC001 Register...………..……………………….......pg 5
Test Case ID TC002 Login……….………………………….………pg 8
Test Case ID TC003 Reset Password……………………………......pg 10
Test Case ID TC004 Logout...…………………………….………....pg 11
Test Case ID TC005 Start Application……………………………...pg 13
Test Case ID TC006 Edit User Settings List………………….…...pg 16
Test Case ID TC007 Search for Stocks …………………………......pg 19
Test Case ID TC008 View Trending Stocks…………………………..pg 21
Test Case ID TC009 View Individual Stock Page……………………....pg 22
Test Case ID TC010 Add to Watchlist…………………………………pg 24
Test Case ID TC011 Add to Portfolio……………………………………pg 25

Introduction

The purpose of testing this Stockalytics is for a user to be able to view, track and add stocks to their portfolio. The user can view each stock on the New York Stock exchange and compare prices against peers. This easy to use application also comes with a trending stocks section which displays the top gainers and losers for each week.

Constraints

Three of the members, Alyssa, Derek, and Horatio will be testing the application for bugs which include text inputs in forms and other user interactions.

Test Items & Environment

We are using Webstorm 2017 2.4 and Brackets 1.12 for our IDE. We will be using Gitkraken for our git commits which will allow us to manage, track, and view each team members commits. For bug reporting purposes, we will using GitKraken Glo. When a team member finds a bug they can go in to the interactive message board and add a bug to the board which will automatically notify each team member. We will keep the most important bugs at the top of the list.

Approach

The type of testing we are using for this stock application is manual black box testing. Black box testing is used to test the input/output behavior of each test case. All of the test cases are found below for the stock application.

Roles

Jarrett Horton: Manager of the team and in charge of writing up everything from Gantt Chart to developer the software. Jarrett’s role in the testing process is to manage the team and assign code to members to test.

Isaac Silvious: Team member in charge of testing for bugs, and to try and break the application. Once application broken or bugs found, will proceed to fix those issues. Worked on numerous amount of JavaScript and CSS with different application processes.

Isaiah Johnson: Team member in charge of coding and css on the stock pages. Helped create Trending Stock, FAQ, gainers and losers table, and settings page.

Russell Quao: Team Member in charge of writing/formatting Use Cases(calculate price alert, calculate trending stocks, search for stocks). Participated in designing the search Gui for the application.

Derek Brown: Manager of the ISM team. Worked on RAD document and Test Cases, as well tested for bugs. Assigned roles as to what needed to be completed for the RAD and Test cases to his team members.

Horatio Hodge: Team member responsible for developing use cases and keeping them up to date. Responsible for developing and updating Test cases as well as testing the appropriate functions of the webpage and searching for bugs.

Daniel Gaisberger: Team member in charge of the database and developer for the homepage.

Phillip Gil-Perea: Responsible for use cases. (Edit user settings, Log out, Forgot Password)

Alyssa Drohan: Team member is responsible for the part of the black box testing.

Schedule

Schedule can be found at https://github.com/StockAppSoftwareEng450/Stock-App/blob/master/Ghant%20Chart/Gantt%20Chart.pdf on page 25.

References

The Requirements Analysis Document (RAD) is available on the stock application website and other project information:
https://github.com/StockAppSoftwareEng450/Stock-App/blob/master/RAD/Rad.pdf

Test Cases

Test Case ID TC001 “Register”

Summary: Verify that a user can create an account as long as the user uses a valid email that is not in the database and the Password and Confirm Password fields are the same.

Prerequisite: Application has the Register screen loaded.

Instructions: For this test, the tester will enter data into the email field and press the Create Account Button to test if the application will check for symbols in email addresses.

Test Data and Expected Result:

1. Change the Email field to “bobdylongmail.com” and press the Create Account.
2. Change the Email field to “@gmail.com” and press the Create Account.
3. Change the Email field to “bobdylo/@gmail.com” and press the Create Account.
4. Change the Email field to “bobdylon”@gmail.com” and press Create Account button.

Result: System should display a prompt saying the email is not valid.

Instructions: For this test, the tester will enter data into the password and the confirm password fields and press the Create Account Button to test if the application will check for invalid characters.

Test Data and Expected Result:

1. Change the Password field to “password,” but leave the Confirm Password field
2. Change the Password field to “password2” and press the Create Account button.
3. Change the Password field to “password2,” but leave the Confirm Password field
4. Change the Password field to “password” and press the Create Account button.

Result: System should display a prompt saying the passwords do not match.

Instructions: For this test, the tester will enter data into the First Name field and the and press the Create Account Button to test if the application will check for invalid characters.

Test Data and Expected Result:

1. Change the First Name field to “Bob#” and press the Create Account button.
2. Change the First Name field to “Bob14” and press the Create Account button.
3. Change the First Name field to “Bob*” and press the Create Account button.
4. Change the First Name field to “Bob ” and press the Create Account button.

Result: System should display a prompt saying names may only contain alphabetic symbols.

Instructions: For this test, the tester will enter data into the Last Name field and the and press the Create Account Button to test if the application will check for invalid characters.

Test Data and Expected Result:

1. Change the Last Name field to “Dylon1” and press the Create Account button.
2. Change the Last Name field to “Dylon#” and press the Create Account button.
3. Change the Last Name field to “Dylon*” and press the Create Account button.
4. Change the Last Name field to “Dylon ” and press the Create Account button.

Result: System should display a prompt saying names may only contain alphabetic symbols.

Instructions: For this test, the tester will enter the correct test data above to verify that the application will create an account in the database and redirect to the homepage.

Test Data and Expected Result:

1. Change the first name field to “Bob”.
2. Change the last name field to “Dylon.
3. Change the email field to “bobdylon@gmail.com”.
4. Change the password field to “Testing1234”
5. Change the confirm password field to “Testing1234”

Result: System should create an account in the firebase database and redirect to the login screen.

Test Case ID TC002 "Login"

Summary: Verify that a registered user can log in to the stock application that is connected to firebase authentication.

Prerequisite: Application has the Login screen loaded and a account with the following data has been created.

1. Email: Bobdylon@gmail.com
2. Password: password2

Instructions: For this test, the tester will enter data into the email and password field and press the Login button to check if the user has provided the correct information to access the application.

Test Data and Expected Result:

1. Enter "Bobdylon@gmail.com " in the Email field and "password1" in the password field and press the Login button.
2. Enter "fakeemail@hotmail.com " in the Email field and "password2" in the password field and press the Login button.
3. Enter "Bobdylon@gmail.com " in the Email field and "password2" in the password field and press the Login button.
4. Enter "Bobdylon2@gmail.com " in the Email field and "pas32" in the password field and press the Login button.

Result: System should display an error message stating the username or password is incorrect.

Instructions: For this test, the tester will enter the correct data into the email and password field and press the Login button to check if the user has provided the correct information to access the application. The user should be transferred to the home screen.

Test Data and Expected Result:

1. Enter “Bobdylon@gmail.com ” in the Email field and “password2” in the password field and press the Login button.

Result: The application should display an confirmation message to the user and the user should be transferred to the home screen.

Test Case ID TC003 “Reset Password”

Summary: The user can reset their password by entering their email address. Firebase will send the user an email link and the user can change their password.

Prerequisite: The user is connected to the internet and is on the reset password page. The has created a valid account.

Instructions: For this test, the tester will enter the incorrect data into the email field press the reset password button.

Test Data and Expected Result:

1. Tester enters “@@@bobdylan@gmail.com” in the email field and presses the reset password button.
2. Tester enters “bobdyla34n@gmail.com” in the email field and presses the reset password button.
3. Tester enters “bob!!!!ylan@gmail.com” in the email field and presses the reset password button.
4. Tester enters “bobdylangmail.com” in the email field and presses the reset password button.

Result: The system displays an error telling the user that the email is not found.

Instructions: For this test, the tester will enter the correct data into the email field press the reset password button.

Test Data and Expected Result:

1. Tester enters “bobdylan@gmail.com” in the email field and presses the reset password button.

Result: The system displays an confirmation that the email has been sent to the user.

Test Case ID TC004 “Logout”

Summary: Verify that the user can log out and that the user will be transferred to the login page. Verify that if the user goes back to the homepage that the user will be transferred to login page instead of the homepage.

Prerequisite: The user has created an account and is logged in and currently on the homepage, trending stocks page, individual stock page, or the FAQ page.

Instructions: For this test, the tester will click on the logout button in the top right of the page.

Test Data and Expected Result:

1. Within the top navigation bar, click the logout button.

Result: System should display a pop menu with “Ready to Leave? Select “Logout” below if you are ready to end your current session. Cancel, Logout”.

Instructions: For this test, the tester will click on the logout button in the top right of the page and select the cancel button when the popup dialog box appears.

Test Data and Expected Result:

1. The user clicks on the logout button in the top right of the screen.
2. The user then clicks selects the cancel button when the dialog box appears.

Result: System should return the user back to the home screen.

Instructions: For this test, the tester will click on the logout button in the top right of the page and select the logout button when the popup dialog box appears.

Test Data and Expected Result:

1. The user clicks on the logout button in the top right of the screen.
2. The user clicks on the Logout button after the dialog button appears

Result: System should return the user back to the home screen.

Test Case ID TC005 “Start Application”

Summary: Will determine if the user has logged on to the website previously.

Prerequisite: The user is connected to the internet and has access to the firebase database. The user has created an account and is logged in. The user is on the homescreen.

Instructions: For this test, the tester will click on the the links on the page.

Test Data and Expected Result:

Dashboard Button:

1. Click on the dashboard button

Result: The user is transferred to the homepage.

Settings:

1. Click on the setting button button at the top of the page.

Result: The user is transferred to the trending stocks page.

Trending Stocks:

2. Click on the trending stocks button at the top of the page.

Result: The user is transferred to the trending stocks page.

My Portfolio Stock Items:

1. Add stocks to the portfolio.

2. Click on the stock symbol link in the portfolio table located on the homepage.

Result: The user should be transferred to the individual stock page for that company.

My Watchlist Stock Items:

1. Add stocks to the watchlist.
2. Click on the stock symbol in the watchlist table located on the homepage.

Result: The user should be transferred to the individual stock page for that company.

Instructions: For this test, the tester will add stocks to their portfolio.

Test Data and Expected Result:

Percentage Change Bar Chart:

1. Hover over the percentage bar chart.

Result: Tooltip should appear with the stock symbol and the percentage number.

Stock Portfolio Diversity:

1. Hover over the Stock Portfolio Diversity

Result: Tooltip should appear with the quantity of stocks in the users portfolio.

Delete Stocks in Portfolio

1. The user observes stocks in the portfolio.
2. The user selects the delete button in the portfolio.

Result: The stock is deleted from the portfolio and from the database

Delete Stocks in Watchlist

3. The user observes stocks in the Watchlist.
4. The user selects the delete button in the Watchlist.

Result: The stock is deleted from the watchlist and from the database

Test Case ID TC006 “Edit User Settings”

Summary: Allows the user to edit email, password, currency, and name which will save the database.

Prerequisite: The user is logged in and the gear icon on the top left of web page is clicked.

Instructions: Change the email to an existing email in the gmail

Test Data and Expected Result:

1. Change the Email field to “test@gmail.com” and press the save button.
2. Change the Email field to “existingUser@gmail.com” and press the save button.
3. Change the Email field to “existerUser1@gmail.com” and press the save button.
4. Re-enter the users email.

Result: System should display an error message saying the email entered is not valid.

Instructions: Change the email to invalid email with excessive symbols.

Test Data and Expected Result:

1. Change the Email field to “chuckb@#$%!illychoe@gmail.com” and press the save button.
2. Change the Email field to “@gmail.com” and press the save button.
3. Change the Email field to “@#$%!illychoegmail.com” and press the save button.
4. Change the Email field to “bobDylangmail.com” and press the save button.

Result: System should display an error message saying the email entered is not valid.

Instructions: Change the email to one that is valid.

Test Data and Expected Result:

1. Change the Email field to “newEmail@gmail.com” and press the save button.

Result: System should display a confirmation message and the updated string will be sent to the database.

Instructions: Change the password that is invalid .

Test Data and Expected Result:

1. Change the password and confirm password to “password” and press the save button.
2. Change the password and confirm password to “e#” and press the save button.
3. Change the password and confirm password to “!@#” and press the save button.
4. Change the password and confirm password to “123” and press the save button.

Result: System should display a error message that lets the user know the password must contain one uppercase, eight characters, one number, and no symbols.

Instructions: Change the password to one that is invalid .

Test Data and Expected Result:

1. Change the password and confirm password to “password” and press the save button.
2. Change the password and confirm password to “e#” and press the save button.
3. Change the password and confirm password to “!@#” and press the save button.
4. Change the password and confirm password to “123” and press the save button.

Result: System should display an error message that lets the user know the password must contain one uppercase, eight characters, one number, and no symbols.

Instructions: Change the password to one that is valid.

Test Data and Expected Result:

1. Change the password to and confirm password to “Testing1234” and press the save button.

Result: System should display a confirmation message that updates the database.

Instructions: Change the first name and last name to one that has numbers and symbols.

Test Data and Expected Result:

1. Change the first name and last name to “Joe4 Wilson” and press the save button.
2. Change the first name and last name to “Joe Wilson4” and press the save button.
3. Change the first name and last name to “JoeWilson 4” and press the save button.
4. Change the first name and last name to “Joe -Wilson” and press the save button.

Result: System should display an error that lets the user know the first name and last name fields are invalid.

Instructions: Change the first name and last name to one that is valid

Test Data and Expected Result:

1. Change the first name and last name to “Joe Wilson” and press the save button.

Result: System should display a confirmation button that updates the database with the new information.

Test Case ID TC007 “Search for Stocks”

Summary: Allows the user to search for a public company in the NYSE either through a company name or through the companies stock symbol.

Prerequisite: The user is logged in and has located the search for stocks.

Instructions: Enter a stock symbol that does not exist and hit enter.

Test Data and Expected Result:

1. Enter “AAPL!” into the search field.
2. Enter “MSFT@” into the search field.
3. Enter “GOOG#” into the search field.
4. Enter “AM”D” into the search field.

Result: The autocomplete will return nothing to select in the input field.

Instructions: Enter a company name that does not exist and hit enter.

Test Data and Expected Result:

1. Enter “Apple!” into the search field.
2. Enter “Microsot” into the search field.
3. Enter “gooles#” into the search field.
4. Enter “AMAssD” into the search field.

Result: The autocomplete will return nothing to select in the input field.

Instructions: Enter a stock symbol that exists and hit enter.

Test Data and Expected Result:

1. Enter “AAPL” into the search field.

Result: The autocomplete will return transfer the user to the respective individual stock page.

Instructions: Enter the name of a public company on the NYSE and hit enter.

Test Data and Expected Result:

1. Enter “Apple Inc.” into the search field.

Result: The autocomplete will return transfer the user to the respective individual stock page.

Test Case ID TC008 “View Trending Stocks”

Summary: Will allow user to view the trending stocks (gainers and losers).

Prerequisite: The user is logged in and the user has clicked on the trending stocks page at the top.

Instructions: Click on the links to make sure the links transfer to the respective individual stock page.

Test Data and Expected Result:

1. Click on stock symbol “aapl” in the gainers section.

Result: The user is transferred to the individual stock page of the selected stock.

Instructions: Click on the links to make sure the links transfer to the respective individual stock page.

Test Data and Expected Result:

1. Click on stock symbol “aapl” in the losers section.

Result: The user is transferred to the individual stock page of the selected stock.

Test Case ID TC009 “View Individual Stock Page”

Summary: This will allow the user to get an in depth look at the respective stock. Price, key stats, peer price, company overview, and related news article are a few items list on this page.

Prerequisite: The user is logged in and the user has clicked on a stock to view more information or to add to their portfolio.

Instructions: Hover over the graph in the middle of screen

Test Data and Expected Result:

1. Hover over the 5 year graph
2. Hover over the 2 year graph
3. Hover over the 1 year graph
4. Hover over the year to date graph
5. Hover over the 6 month graph
6. Hover over the 3 month graph
7. Hover over the 1 month graph
8. Hover over the 1 day graph

Result: The system will display the date and the price when hovering hovering the line graph.

Instructions: Click on the buttons beneath the graph in the middle of screen

Test Data and Expected Result:

1. Click on the 5 year graph
2. Click on the 2 year graph
3. Click on the 1 year graph
4. Click on the year to date graph
5. Click on the 6 month graph
6. Click on the 3 month graph
7. Click on the 1 month graph

8. Hover over the 1 day graph

Result: The system will display the date and the price when hovering hovering the line graph.

Instructions: Click on the buttons beneath the graph in the middle of screen

Test Data and Expected Result:

1. Click on the 5 year graph
2. Click on the 2 year graph
3. Click on the 1 year graph
4. Click on the year to date graph
5. Click on the 6 month graph
6. Click on the 3 month graph
7. Click on the 1 month graph
8. Hover over the 1 day graph

Result: The system will change the graph to update the respective timeline

Instructions: Click on the news article links down at the bottom in the news section

Test Data and Expected Result:

1. Click on the first card in the news section
2. Click on the second card in the news section
3. Click on the third card in the news section

Result: The link will redirect the user to the news site where the article was written.

Instructions: Click on the links in the peers section

Test Data and Expected Result:

1. Click one of the links in the peers section.

Result: The link will redirect the user to a related companies individual stock page.

Test Case ID TC010 “Add to Watchlist”

Summary: This will allow the user add a particular stock to their watchlist. This will allow the user to the current price, 3 month percent, 6 month percent, and 1 year percent.

Prerequisite: The user is logged in and is on the individual stock page.

Instructions: The tester clicks on the add to watchlist button on the individual stock page.

Test Data and Expected Result:

1. The user clicks on the “add to watchlist” plus button at the of the individual stock page.

Result: The stock is added to the watchlist table in the database.

.

.

Test Case ID TC011 “Add to Portfolio”

Summary: This will allow the user add a particular stock to their portfolio. This will allow the user to view the date of purchase, purchase price, current price, quantity, purchase cost, current equity, net profit/ loss, after tax profit, and current percentage change.

Prerequisite: The user is logged in and is on the individual stock page.

Instructions: The tester clicks on the add to portfolio button on the individual stock page and enters incorrect data for date.

Test Data and Expected Result:

1. The user clicks on the “add to portfolio” plus button at the of the individual stock page.
2. The user selects “1/!/2018” in the date field
3. The user keeps the current price field.
4. The user enters “1” in the quantity field.

Result: The stock is not added to the portfolio and an error message appears.

Instructions: The tester clicks on the add to portfolio button on the individual stock page and enters incorrect data for the current price.

Test Data and Expected Result:

1. The user clicks on the “add to portfolio” plus button at the of the individual stock page.
2. The user selects “1/1/2018” in the date field.
3. The user enters “bob” in the current price field.
4. The user enter as “1” quantity

Result: The stock is not added to the watchlist table and an error message appears

Test Data and Expected Result:

1. The user clicks on the “add to portfolio” plus button at the of the individual stock page.
2. The user leaves the date field blank
3. The user keeps the current price
4. The user enter as

Result: The stock is not added to the watchlist table and an error message appears

Instructions: The tester clicks on the add to portfolio button on the individual stock page.

Test Data and Expected Result:

1. The user clicks on the “add to portfolio” plus button at the of the individual stock page.
2. The user selects “02/18/18” in the purchase date field.
3. The user selects “$123.45” in the price field.
4. The user enters “45” in the quantity field.

Result: The stock is added to the watchlist table in the database.