Ben Swanson
Nick Clason
Jack Komrska

Overall Test Plan

The testing that will be conducted for our web application will be split into four separate components. These components include Account Authentication, Data Retrieval, Database, and Compatibility. The first component is Account Authentication; by conducting the tests in this area, we can verify that users are granted a functional and secure account service. Testing the Data Retrieval and Database areas of the web application will ensure that users are getting the correct corresponding data to their requests. Finally, the Compatibility of the application will be tested in order to assure users will be able to utilize this application across a variety of different browsers/devices/platforms, so that there will be a consistent user experience across all users.

Test Case Descriptors

AA1.1	Account Authentication Test 1
AA1.2	This test will ensure that a user can successfully register for a newly created account
AA1.3	This test will involve whoever is testing to create a new account using the register tab that is available in the navigation bar.
AA1.4	Inputs: first name, last name, email, password, and confirmation password
AA1.5	Outputs: The user will be pushed to a login page to input their log-in information
AA1.6	Normal
AA1.7	Blackbox
AA1.8	Functional
AA1.9	Unit Test

AA2.1	Account Authentication Test 2
AA2.2	This test will ensure that entering a user’s credentials will perform authentication successfully
AA2.3	The user will go to the log-in page and will enter credentials to log-in to the application
AA2.4	Inputs: email, password
AA2.5	Outputs: User will be pushed to stocks-info page
AA2.6	Normal
AA2.7	Blackbox
AA2.8	Functional
AA2.9	Unit Test

AA3.1	Account Authentication Test 3
AA3.2	This test will ensure that a user can successfully log out and the system performs the correct deauthentication process
AA3.3	The user will click the logout button in the navigation bar, and click the confirmation button to log out.
AA3.4	Inputs: 2 separate left clicks
AA3.5	Outputs: User is logged out and put in the home screen
AA3.6	Normal
AA3.7	Blackbox
AA3.8	Functional
AA3.9	Unit Test

AA4.1	Account Authentication Test 4
AA4.2	This test will ensure that a user will be denied authentication to an account if attempting to log in with incorrect credentials
AA4.3	The user will go to the login screen and enter invalid credentials
AA4.4	Inputs: Invalid email and/or password
AA4.5	Outputs: No login granted, and display of invalid credentials
AA4.6	Normal
AA4.7	Blackbox
AA4.8	Functional
AA4.9	Unit Test

DR1.1	Data Retrieval Test 1
DR1.2	This test will ensure that a user can select a financial entity and its corresponding price history and information is retrieved from the database and displayed to the user
DR1.3	The user will select a stock or cryptocurrency and input how much of it they have, and the application will display how much it is worth
DR1.4	Inputs: Financial entity and how much of it
DR1.5	Outputs: Worth of the number of said entities
DR1.6	Normal
DR1.7	Blackbox
DR1.8	Functional
DR1.9	Unit

DR2.1	Data Retrieval Test 2
DR2.2	This test will ensure that when a user navigates to their portfolio within the application, their assets’ corresponding data for prices and amounts display successfully
DR2.3	The user will navigate to the page of their finances, and will display the information of their assets’.
DR2.4	Inputs: Credentials
DR2.5	Outputs: Financial information based on their credentials
DR2.6	Normal
DR2.7	Blackbox
DR2.8	Functional
DR2.9	Unit

DR3.1	Data Retrieval Test 3
DR3.2	This test will ensure that a user can request the model’s price projection and a generated price outlook will be returned to the user
DR3.3	When a user requests a price prediction for a given asset, they will be returned a projection that they can then view
DR3.4	Inputs: Stock to view prediction for
DR3.5	Outputs: a graphical/numerical value for projected price at a given time point/period
DR3.6	Normal
DR3.7	Whitebox
DR3.8	Functional
DR3.9	Unit

DB1.1	Database Test 1
DB1.2	This test will ensure that the database query when logging in returns the correct response
DB1.3	The query will be issued via api call containing the email as a parameter, then, if successful, a valid AccessToken and RefreshToken will be issued.
DB1.4	Inputs: Email
DB1.5	Outputs: AccessToken, RefreshToken
DB1.6	Normal
DB1.7	Whitebox
DB1.8	Functional
DB1.9	Unit

DB2.1	Database Test 2
DB2.2	This test will ensure that the database query for registering a user returns the correct response and commits the user to the database
DB2.3	The query will be issued via api call containing the users name, email, and password, if successful, the user will be committed to the database and the backend will issue a successful registration message
DB2.4	Inputs: First Name, Last Name, Email, Password
DB2.5	Outputs: User added to database and success message
DB2.6	Normal
DB2.7	Whitebox
DB2.8	Functional
DB2.9	Unit

C1.1	Compatibility Test 1
C1.2	This test will ensure that a user can access the web application successfully across different browsers and platforms, such as Chrome and Firefox on PC, Mac, and Mobile
1.3	The website and all pages will be loaded on each browser, if the website loads and is displayed without issue, the test is successful
1.4	Inputs: N/A
1.5	Outputs: Website and all elements properly load and render on all platforms
1.6	Normal
1.7	Blackbox
1.8	Performance
1.9	Integration

Test Case Matrix

	Normal/ Abnormal	Blackbox/ Whitebox	Functional/ Performance	Unit/ Integration
AA1	Normal	Blackbox	Functional	Unit
AA2	Normal	Blackbox	Functional	Unit
AA3	Normal	Blackbox	Functional	Unit
AA4	Normal	Blackbox	Functional	Unit
DR1	Normal	Blackbox	Functional	Unit
DR2	Normal	Blackbox	Functional	Unit
DR3	Normal	Blackbox	Functional	Unit
DB1	Normal	Whitebox	Functional	Unit
DB2	Normal	Whitebox	Functional	Unit
C1	Normal	Blackbox	Functional	Unit