Data and Algorithms

Fortunately, are system doesn’t use many complex data structures. Since we store the data in an SQL database. We can retrieve the data from the database and store it in simple container objects which we then use to generate the display for the user. Also, since the front end is primarily responsible for interacting with the user the complexity of the algorithms is low. Most of the functions involve storing and retrieving text-based data which is a common and well-known type of system.

Front-End Tests

**Display Tests**

Test case 1: Display View

Function Tested: displayView()

Success/fail: an html page containing the details of a clicked challenge

Expected Result: A detailed page contain all relevant challenge information is displayed

Expected Failure: No page is displayed

Test case 2: Display Challenge

Function Tested: displayChallenge()

Success/fail: a html page contains a list of challenges meeting the criteria is generated

Expected Success: a list is correctly displayed of relevant challenges

Expected Failure: No new page is displayed

Test Case 3: Display Error

Function Tested: DisplayError(error)

Success/fail: a detailed error message is displayed on screen

Expected Success: an error message that contains info about the error is displayed

Expected Failure: a generic error message of “something went wrong” is displayed

**Search Tests**

Test case 4: build search

Function Tested: buildSearch()

Success/fail: a successful and correct searchParameters has been built,

Expected Success: An accurate searchParameters is built from user generated parameters

Expected Fail: A default searchParameters is built from default values

Test case 5: send query

Function Tested: sendQuery()

Success/fail: a successful query has been passed to DBconection

Expected Success: a query is sent down through DBconnections to the database

Expected Failure: a query is unable to be sent, and the user is notified that it failed to search

**Format Data Tests**

Test Case 6: Format list

Function Tested: formatList(listData)

Success/fail: correct data is provided, and a list is built

Expected Success: a successful list is built and sent to the lister

Expected Failure: a list was unable to be built and an error is sent to lister

Test Case 7

Function Tested: formatChallenge(challengeData)

Success/fail: correct data is provided and a challenge has been built

Expected Success: an accurate challenge object is sent to viewer

Expected Failure: a error is sent to challenge viewer

Test Case 8

Function Tested: buildPage()

Success/fail: a page has been correctly built that can be displayed

Expected Success: a page has been built and passed into the interface to display to user

Expected Failure: an error message is displayed to user