Deliverable #3

The Boyoz
Nathan Bell
Logan Sitar
Paul Joseph

Experience

Being the first time that any of us have worked on any test automation, I think we came away with a lot of knowledge. We gained experience in Linux commands, reading and understanding open-source code, creating a script to automate our test cases, printing results to an html file and then removing any necessary files after execution. We also learned that the windows and Linux shells differ in how they save files. The hardest part about this task was understanding the open-source methods and what their functions were.

Description of Framework

If you take a look at the testing script you can see that what we did was pretty simple. First off we run a separate setup script that formats the html results file that we'll be adding to later. Then we compile all java classes in our executable folder which has both our drivers and tested classes inside. After that we loop through all of the test case files. For each test case file we have a while loop that populates variables with the data from each line. That data is then read to know what driver to run the given input in. The output from that driver is passed into another variable which is then checked against the expected output from the test case and is declared either a pass or a fail. All of the data is then passed into the results file as a row of a table. When the script is done looping through every test case it opens the results file in firefox before clearing the executables folder of all of the compiled class files.

How-To Documentation

In order to run this automated test case, clone the "The-Boyoz" repository from github, find the cloned folder in your directories and cd into TestAutomations. Once your directory is set to TestAutomations, run ./scripts/runAllTests.sh from the command line. An html file should open in your browser with the test case results.

Example Test Cases

Example Test Case Outline

```
Id:
Requirement:
Class:
Method:
Input:
Output:
```

Test Case 1:

```
1
Converts a color object to a HSL value
ColorConverter.java
rgb2Hsl
000000
hsl(0, 0%, 0%)
```

Test Case 2:

```
2
Converts a color object to a HSL value
ColorConverter.java
rgb2Hsl
ffffff
hsl(0, 0%, 100%)
```

Test Case 3:

```
3
Converts a color object to a HSL value
ColorConverter.java
rgb2Hsl
1234fc
hsl(231, 97%, 52%)
```

Test Case 4:

```
4
Converts a color object to a HSL value
ColorConverter.java
rgb2Hsl
abc123
hsl(68, 69%, 44%)
```

Test Case 5:

```
5
Converts a color object to a HSL value
ColorConverter.java
rgb2Hsl
8f32a6
hsl(288, 53%, 42%)
```