Drayton Redick

Nicholas Foster

Lawson Willard

Chapter 3

As it turns out, switching to the full version of Tanaguru caused even more problems than before. Namely, the entire repository is so large that it couldn't even fit on everybody's computer. Thus, we decided it would be best to switch back to just the Contrast Finder section of Tanaguru. This left us with the problem that the program is still designed to be run within Tanaguru, and so we created our own driver to run the program for testing purposes.

We decided to write the script in bash, and after a lot of reviewing bash we ended up with a satisfactory script. Firstly, the script sets up a table to make the html output very clear and readable. Next, it compiles the executables for the test case. It then reads through the test case files we provided and assigns its entries into an array. The script checks what component the file is testing for, runs the tests, and finally prints them into an html file. Lastly, the script opens firefox and displays the results using this html file.

The Testing Process

Start with testing input parameters for the command line interface. We will need to test if the system can parse the correct options and if it can parse a URL correctly. If it is given a invalid input it needs to throw an error describing why the input was invalid. From here we will test different kinds of audits on different services. We will test whether audits are compatible on different services. When testing the audits we will also be testing certain compatibility and conversion services inside the program. Along with this we will test for correct output assuming the correct input was given. Each section will cover roughly 5 test cases with the compatibility and conversions category taking 10 test cases. We decided to have 10 test cases here because this is the most important part of the Tanaguru program.

Requirements Traceability

-The system will need to be able to parse input from a command line. It will need to test if there are the correct number of inputs and if they are valid.

-The system will need to figure out which audits it needs to perform on the website based on the input from the command line. This part must be able to be customized by the users. If an audit crashes it must give an appropriate message as to why it crashed. Audits will also be tested to see if they are compatible on different services.

-When the system is done testing a website it will need to report how many tests passed, how many failed and how many were not used in testing the website. It must do this for every URL that is passed into it.

-When testing audits we will test a lot of the compatibility and converting systems that have been created. This makes up the most of their engine and thus needs the most testing. A lot of this includes sorting websites and converting them into parameters to be input into the different tests. We will test the conversion and parameterization of these objects.

Test recording procedures

-We will create system in which test cases are labelled, displayed and marked whether they passed or failed. On each of the passes or fails it will record the input parameters and remember its output. This output will be compared to the expected output for determining whether the system failed or passed.

Constraints

-Limiting test cases to the most important and useful aspects of the project.

System tests

We are choosing to implement our easiest tests first so that we may also work on a good automated testing framework.

Test Case 1:

-Test command line input options

-This will test the options when determining what type of audit to create

Test Case 2:

-Test command line URL with no URL

Test Case 3:

-Test command line with an invalid URL

Test Case 4:

-Test with a valid URL

Test Case 5:

-Test with no input

Test Cases Input (6)

-URL (3)

-Options(3)

Test Cases Audit (5)

-Page audit on scenario

-Scenario audit on page

-File audit on page

-Site audit on page

-...

Test Cases Conversion & Compatibility (10)

...

Test Cases Output (4)

-Site audit

-Page audit

-File audit

-Scenario audit