**CS 1632 - DELIVERABLE 3: Automated System Testing of a Web Application**

**Author:** Navdeep Handa

**GitHub Username:** navdeephanda

**Code and Tests:** https://github.com/navdeephanda/D3

**SUMMARY/ISSUES FACED**

Overall, I found automated systems testing to be a very interesting blend of the ideas of unit testing and manual testing. I found myself drawing from both testing methods when creating the systems tests. I took the user focus and idea of testing to the interface that I learned from manual testing and combined it with the idea of making your source code testable, something I learned from unit testing.

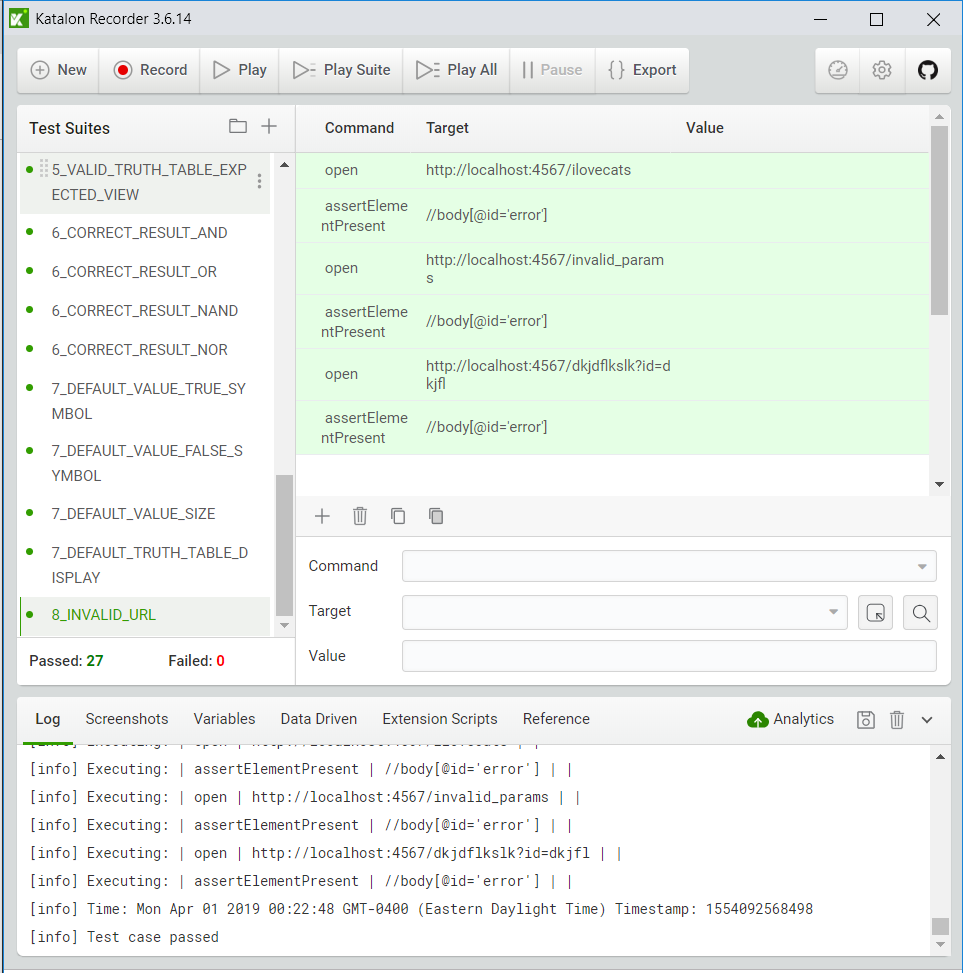
The major issues I faced were of handling ambiguity in the requirements, learning Katalon/Selenese, and making my code receptive to the language of Katalon/Selenium.

The first main requirement ambiguity that I saw was how spaces in input should be interpreted. I thought that it didn’t make sense, from a usability perspective, to have a single space be the true character or the false character, because that would make the truth table really annoying to read. Furthermore, I thought it’d be awkward if the user entered a space for the number accidentally and they were led to an invalid error page. They may not even realize they entered that space in the textbox. In those situations, I interpret a space to be a blank parameter and use the default. In the case where they enter a character/number into any of the input fields and then some whitespace, I used basically the same reasoning to say that it was most likely accidental and interpret it as just the single character. The second requirement ambiguity was whether, if the page where the truth table is rendered is a separate URL/route, that page should be publicly accessible and how it should behave in the absence of parameters if it is. My implementation uses an index page that has an input form, from which parameters are passed to a display route. This means that my display route is publicly accessible and gives away some of the structure of my web application. I didn’t think this was a very big deal (nobody gains anything by hacking this simple little truth table generator). Therefore, I kept this /display route, and I made it so that when a user accesses display directly (i.e. all the parameters which are usually passed to /display are nil), the truth table is set to its defaults. I believe this is valid behavior because the requirements didn’t say I couldn’t do this.

The tool itself used for automated testing, Katalon Web Recorder, caused me probably the most amount of issues. I thought writing the source code for the web app was far easier than utilizing Katalon to test it. I wrote my application one way and found that the way I had done some things made it either impossible/harder to test or would produce a more fragile test. For example, on the root page, I had put the labels for the user input textboxes in plaintext in the body of the html document. Doing this meant I could not verify the presence of each individual label preceding its respective textbox. I ended up changing my implementation to use the <label> tag so that I could select each label on the webpage.

As a side note, regarding the requirements, my approach to testing was slightly different to my approach to the web app. I chose to write the tests very closely to the stated requirements. My application itself closely follows both the written requirements and the screenshots. For example, in my test for Requirement 3, I verify an h1 tag because the requirements specifically state it. However, I use the less strong assertTextPresent to verify the text on that page because the requirement doesn’t specify how that information supposed to be laid out (although the screenshots suggest a way that it should be done). I did it this way because I felt that testing to exactly the way the web application was presented in the screenshots is too granular. I felt the specificity in the requirements was an appropriate granularity to test at.

**EVIDENCE OF TESTS PASSING**



|  |  |
| --- | --- |
| **Requirement** | **Associated Test Case(s)** |
| 1 | 1\_MAIN\_PAGE\_EXPECTED\_VIEW |
| 2 | 2\_MAIN\_PAGE\_LINKS\_TO\_SELF  2\_INVALID\_ADDRESS\_PAGE\_LINKS\_TO\_MAIN\_PAGE  2\_INVALID\_PARAMS\_PAGE\_LINKS\_TO\_MAIN\_PAGE  2\_DISPLAY\_PAGE\_LINKS\_TO\_MAIN\_PAGE |
| 3 | 3\_INVALID\_PARAMS\_PAGE\_EXPECTED\_VIEW |
| 4 | 4\_SAME\_TRUE\_FALSE\_SYMBOL\_USER\_INPUT  4\_SAME\_TRUE\_FALSE\_SYMBOL\_DEFAULT\_TRUE  4\_SAME\_TRUE\_FALSE\_SYMBOL\_DEFAULT\_FALSE  4\_DIFFERENT\_TRUE\_FALSE\_SYMBOL  4\_SIZE\_INTEGER\_TWO  4\_SIZE\_INTEGER\_GREATER\_THAN\_TWO  4\_SIZE\_INTEGER\_LESS\_THAN\_TWO  4\_SIZE\_STRING  4\_SIZE\_DOUBLE  4\_MULTICHARACTER\_TRUE\_FALSE\_SYMBOLS  4\_WHITESPACE\_IN\_INPUTS |
| 5 | 5\_VALID\_TRUTH\_TABLE\_EXPECTED\_VIEW |
| 6 | 6\_CORRECT\_RESULT\_AND  6\_CORRECT\_RESULT\_OR  6\_CORRECT\_RESULT\_NAND  6\_CORRECT\_RESULT\_NOR |
| 7 | 7\_DEFAULT\_VALUE\_TRUE\_SYMBOL  7\_DEFAULT\_VALUE\_FALSE\_SYMBOL  7\_DEFAULT\_VALUE\_SIZE  7\_DEFAULT\_TRUTH\_TABLE\_DISPLAY |
| 8 | 8\_INVALID\_URL |

**TRACEABILITY MATRIX**