Scott Crawford: Scott-Crawford

Sumanyu Gupta: sumanyugupta

<https://github.com/sumanyugupta/CS1632_Deliverable4>

CS 1632 – Deliverable 4: Automated Web Testing

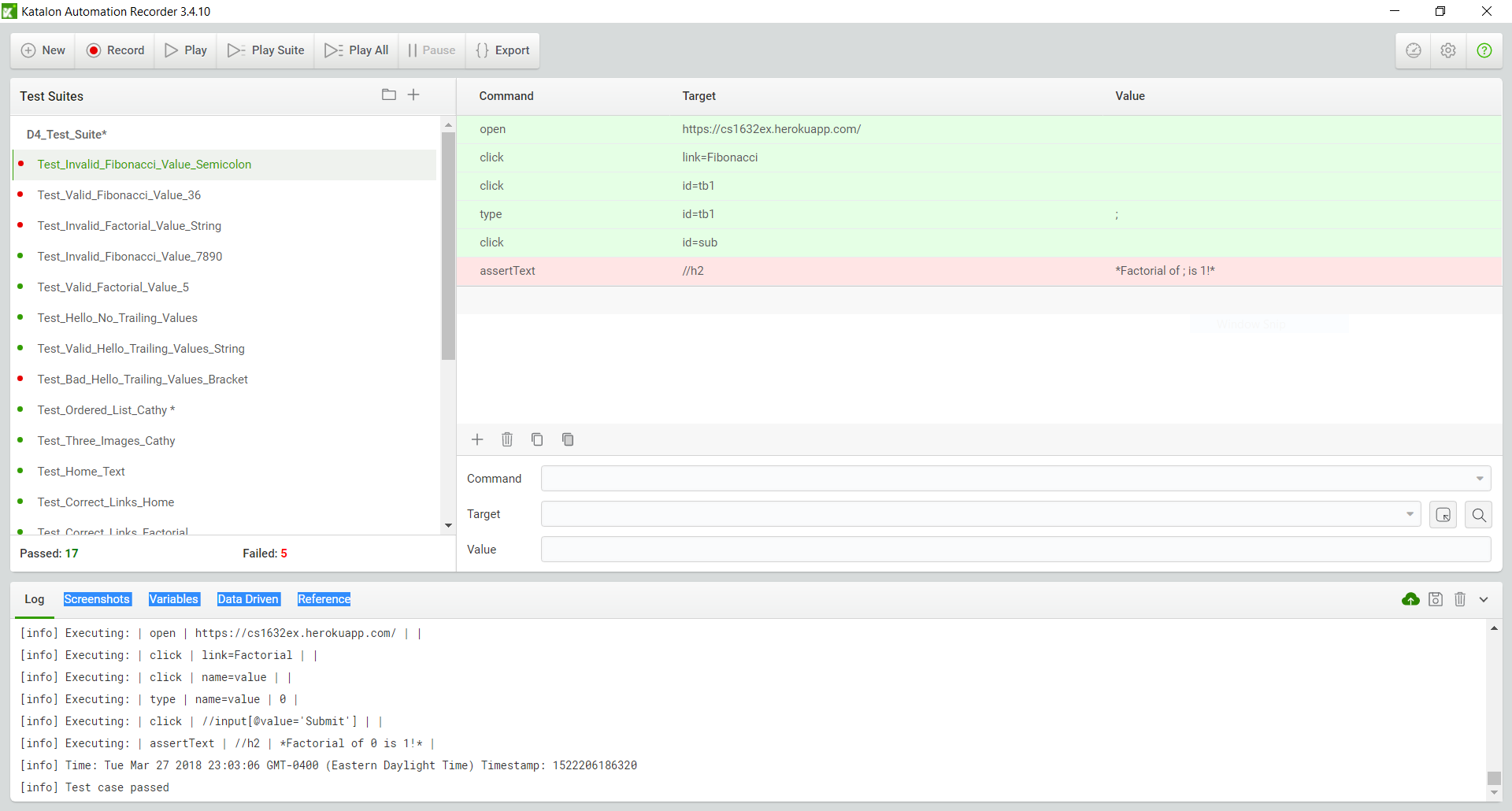
**Summary**

Perhaps the most difficult issues we faced while testing this application came before we actually started writing tests. There was a bit of a learning curve in learning how to use the Katalon Automation Recorder and this caused us to start the project in a slightly delayed fashion. There are an enormous amount of assertions possible in Katalon and deciding which ones to use in our test cases proved to also take up a lot of time. In order to have a broad variety of cases while simultaneously focusing on edge and base cases, more planning was required up-front that involved writing out the order in which we would begin testing. There were also some issues with not being able to assert everything we wanted to in order to make sure the requirements were being met correctly. For requirement 1, we were unsure how to test for there being a new line character or not within our tests. For requirement 2, we were able to identify that the links existed on each page, but not where they linked to. Finally, for requirement 8, it is possible to know that there are 3 images on a page, but not if they are images of Cathy outside of looking at the alt text, which may be a lie.

In our test suite, we had a total of 5 failing tests and 17 passing tests. The failing tests were considered as defects in the application. A major defect occurred with the Fibonacci page, where any positive integer input greater than 30 and less than 100 always returned the value 1 instead of actually calculating the correct Fibonacci value of the input. This was impactful because 66% of the potentially valid input was no longer useful. Another defect we encountered was that accessing the “hello” page with a trailing character such as a square bracket or percent symbol caused the entire application to crash. More specifically, the application did not have any sort of graceful exit or error handler built in because the user was able to see the Heroku error page. As an application, this is something the user should never be able to view as it diminishes the overall UI of the page. Finally, a similar but different defect yet again involved both the Fibonacci page and Factorial. This time, invalid inputs that were not numeric, such as ‘Laboon’ returned Internal Server Error pages to the user instead of providing the correct message. To reiterate, an error page like the aforementioned severely impacts the look and feel of an application and should always be abstracted away from the user.

Overall, we had fun learning a new tool for automated web testing and found it extremely beneficial to simulate many different test cases with just the simple push of a button.

**Screenshot**

****

**Defects**

SUMMARY: Entering a non-numeric value on the Fibonacci does not display the correct message.  
DESCRIPTION: This defect was discovered in test case Test\_Invalid\_Fibonacci\_Value\_Semicolon. Requirement 5 states that entering an invalid value on the Fibonacci or Factorial page should result in the user being informed that the value is 1. However, when an invalid value such as a string or alphanumeric is entered, the page instead displays an “Internal server error” message to the user. This defect can also be seen on the Factorial page.

REPRODUCTION STEPS: User should first navigate to <https://cs1632ex.herokuapp.com/>. Next, he/she should go to the Fibonacci page by clicking on the 3rd blue link labeled “Fibonacci” on the top of the page. Finally, the user should type “Laboon” in the text box and press the “Submit” button.  
EXPECTED BEHAVIOR: The user should be shown a message with the text “Fibonacci of Laboon is 1!”  
OBSERVED BEHAVIOR: An “Internal server error” page replaces the webpage.

SUMMARY: Adding a square bracket while accessing the “/hello” page does not display the correct message.  
DESCRIPTION: This defect was discovered in test case Test\_Bad\_Hello\_Trailing\_Values\_Bracket. Requirement 7 states that entering any input value should result in the page displaying a hello message from that trailing value. However, accessing “./hello/]” causes an application error instead of displaying the required message.

REPRODUCTION STEPS: User should first navigate to <https://cs1632ex.herokuapp.com/>. Next, he/she should go to the Hello page by clicking on the 4th blue link labeled “Hello” on the top of the page. Finally, the user should mouse click after the word “hello” in the address bar of the webpage so that the URL is not highlighted and there is a blinking cursor displayed, type “/]”, and press the “Enter” button.  
EXPECTED BEHAVIOR: The page should display a message with the text “Hello 1632, from ]!”  
OBSERVED BEHAVIOR: The page results in an application error and there is no hello message displayed. Instead, “An error occurred in the application and your page could not be served. If you are the application owner, check your logs for details” is shown to the user.

SUMMARY: A valid input value of greater than 30 on the /fib page results in the incorrect Fibonacci value being shown to the user.  
DESCRIPTION: This defect was discovered in the test case Test\_Valid\_Fibonacci\_Value\_36. Requirement 4 states that a valid input for the Fibonacci page is considered a positive integer between 1 and 100 and pressing submit should show the user the Fibonacci of that input. However, entering a positive integer greater than 30 and less than or equal to 100 always results in the incorrect Fibonacci value of 1 being shown to the user.

REPRODUCTION STEPS: User should first navigate to <https://cs1632ex.herokuapp.com/>. Next, he/she should go to the Fibonacci page by clicking on the 3rd blue link labeled “Fibonacci” on the top of the page. Finally, the user should type “36” in the text box and press the “Submit” button.  
EXPECTED BEHAVIOR: The page should display a message with the text “Fibonacci of 36 is 14930352!”  
OBSERVED BEHAVIOR: The page incorrectly displays a message with the text “Fibonacci of 36 is 1!”

**Traceability Matrix**

Requirement 1: Test\_Home\_Text

Requirement 2: Test\_Correct\_Links\_Home, Test\_Correct\_Links\_Factorial, Test\_Correct\_Links\_Fibonacci, Test\_Correct\_Links\_Hello, Test\_Correct\_Links\_Cathedral

Requirement 3: Test\_Valid\_Factorial\_Value\_5, Test\_Valid\_Factorial\_Value\_1, Test\_Valid\_Factorial\_Value\_100

Requirement 4: Test\_Valid\_Fibonacci\_Value\_36, Test\_Valid\_Fibonacci\_Value\_1, Test\_Valid\_Fibonacci\_Value\_100

Requirement 5: Test\_Invalid\_Fibonacci\_Value\_Semicolon, Test\_Invalid\_Factorial\_Value\_String, Test\_Invalid\_Fibonacci\_Value\_7890, Test\_Valid\_Factorial\_Value\_0, Test\_Valid\_Fibonacci\_Value\_0

Requirement 6: Test\_Hello\_No\_Trailing\_Values

Requirement 7: Test\_Valid\_Hello\_Trailing\_Values\_String, Test\_Bad\_Hello\_Trailing\_Values\_Bracket

Requirement 8: Test\_Ordered\_List\_Cathy, Test\_Three\_Images\_Cathy