Testing of Humanitarian Free and Open Source Software: Tanaguru Contrast-Finder

Collin Bauer   Dylan Evans   Isabel Lally

CSCI 362: Software Engineering

Abstract

The purpose of this project was to create 25 test cases for five different methods within the HFOSS (Humanitarian Free and Open Source Software) Tanaguru Contrast Finder. Post testing, we injected five faults into the code to have several test cases fail.
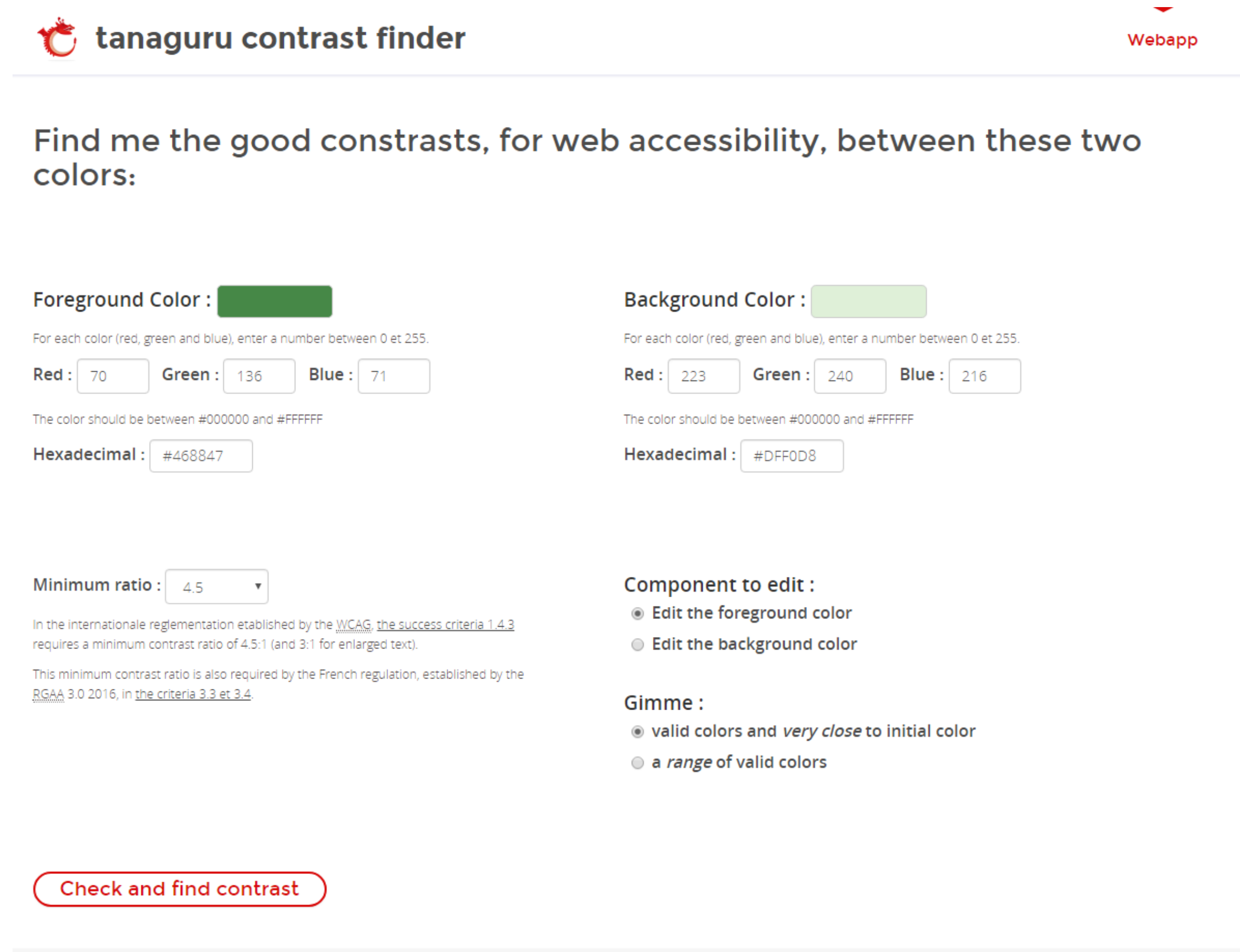
Figure 1. Tanaguru Contrast Finder
https://contrast-finder.tanaguru.com/

Run All Tests

To implement our tests, we created two bash scripts and a Java driver for each method. The main script, runAllTests.sh, takes the following steps:

- Compiles all drivers
- Reads in information from each test case text file
- Sends the information to the correct driver to run
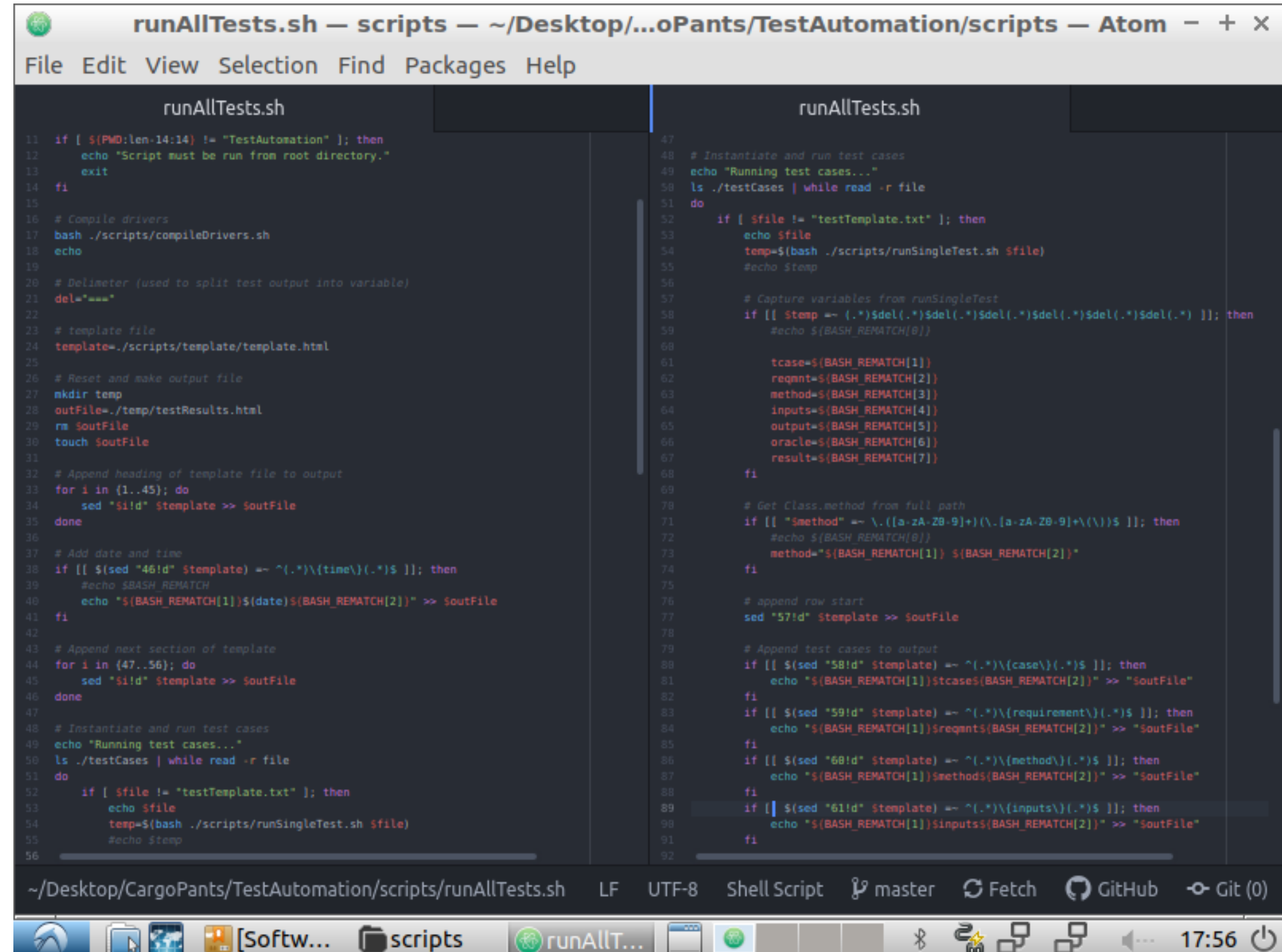- Sends output to an HTML file and opens that file in a browser

Figure 2. Portions of runAllTests.sh

Figure 3. Results Before Fault Insertion

Test Cases

Methods Tested:

- ColorConverter.rgb2Hex( )
- ColorConverter.hex2Rgb( )
- ContrastChecker.computeContrast( )
- ColorConverter.ofsetRgbColor( )
- DistanceCalculator.calculate( )

For each test case, we wrote a text file that follows the format of testTemplate.txt such that the information needed may be accessed by the script.
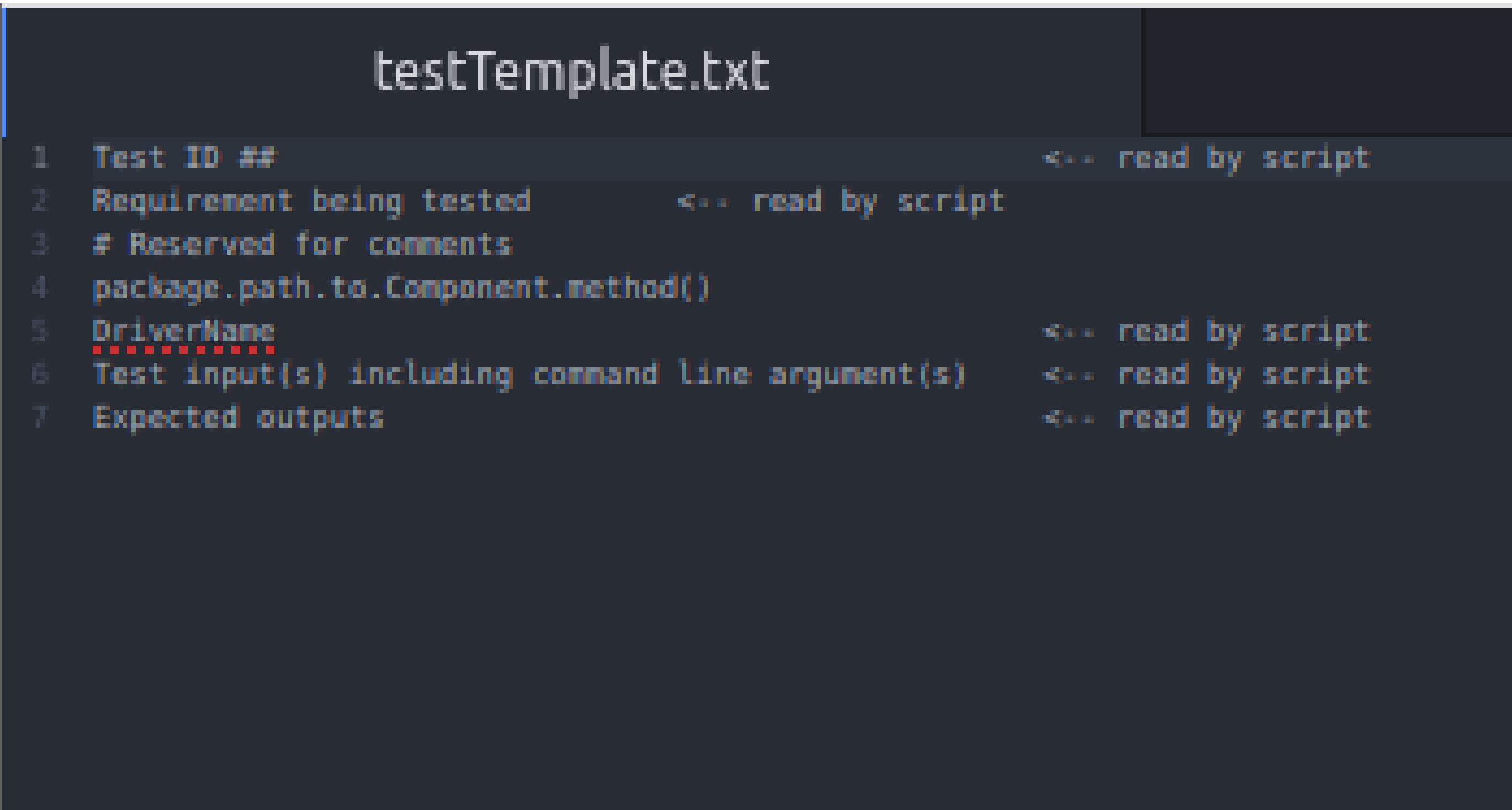

Figure 4. Test Case Template

Faults

We inserted five faults into the Contrast Finder to create failed tests. From our changes in the code, the only tests that still passed were in hex2Rgb() and rgb2Hex() in the cases where order would not matter. We did not insert and error into calculate(), since an error already existed.

Figure 5. Results Post Fault Insertion

Conclusion

Initially, we assumed that all test cases would pass before our fault injections. We were surprised when some test cases for the method DistanceCalculator.calculate() failed due to a formula being implemented incorrectly. (We reported this error on Tanaguru's GitHub). However, from how simple it was to insert faults, it is understandable how this could happen. In this specific case, testing was simple for this method and the fix isn't difficult to implement. At the beginning of our project, 25 test cases sounded like a lot, but have since figured out that 25 test cases is a very small amount that gives little information.

Acknowledgments

COLLEGE of CHARLESTON
DEPARTMENT of COMPUTER SCIENCE
1770