

# 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.

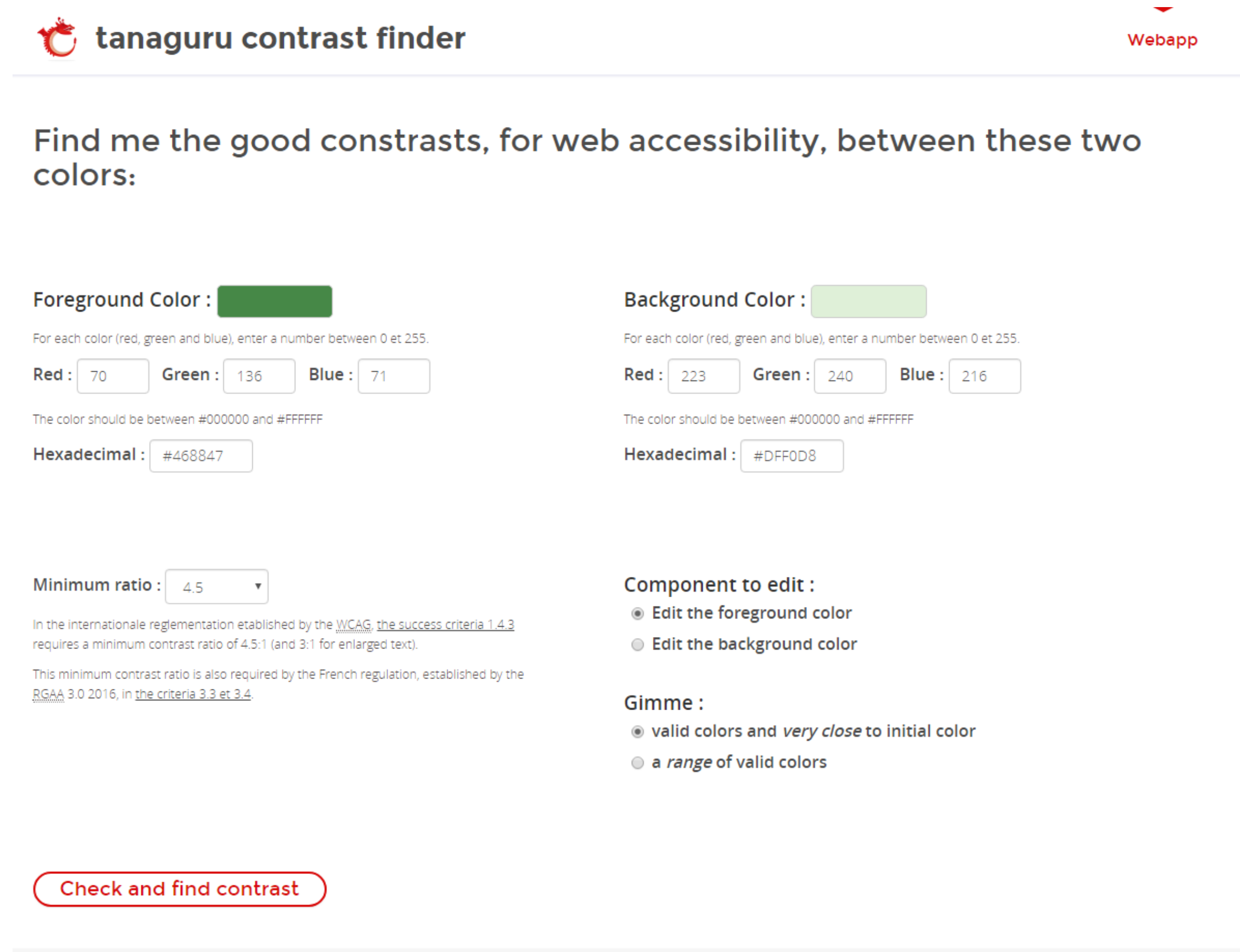


Figure 1. Tanaguru Contrast Finder  
<https://contrast-finder.tanaguru.com/>

## 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

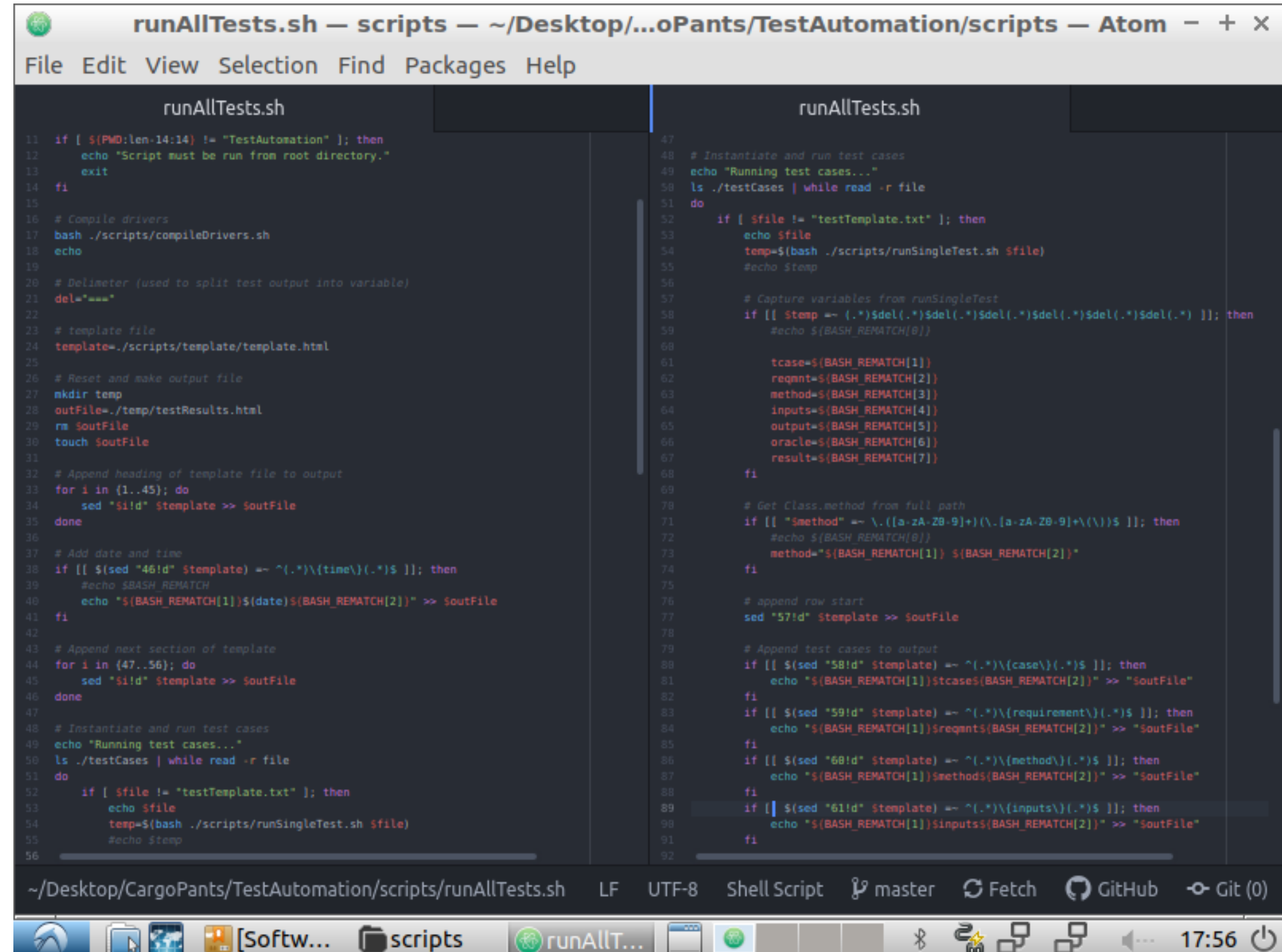


Figure 2. Portions of runAllTests.sh

Case	Requirement	Method	Inputs	Output	Oracle	Result
01	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	220 136 15	#DC8B0F	#DC8B0F	pass
02	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	0 0 0	#000000	#000000	pass
03	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	255 255 255	#FFFFFF	#FFFFFF	pass
04	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	30 15 2	#1E0F02	#1E0F02	pass
05	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	184 93 23	#B83D19	#B83D19	pass
06	The method will take a valid color defined by hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xDC8B0F	rgb(220, 136, 15)	rgb(220, 136, 15)	pass
07	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0x000000	rgb(0, 0, 0)	rgb(0, 0, 0)	pass
08	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xFFFFFF	rgb(255, 255, 255)	rgb(255, 255, 255)	pass
09	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	18 52 86	rgb(18, 52, 86)	rgb(18, 52, 86)	pass
10	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xAABBC	rgb(170, 187, 204)	rgb(170, 187, 204)	pass
11	This method takes two doubles and computes the contrast between them and outputs in the form of a double.	ContrastChecker.computeContrast()	50.0 20.0	2.5	2.5	pass
12	This method takes two doubles and computes the contrast between them and outputs in the form of a double.	ContrastChecker.computeContrast()	-50.0 0.0	-999.0	-999.0	pass

Figure 3. Results Before Fault Insertion

## Test Cases

### Methods Tested:

- ColorConverter.rgb2Hex()
- ColorConverter.hex2Rgb()
- ContrastChecker.computeContrast()
- ColorConverter.offsetRgbColor()
- 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.

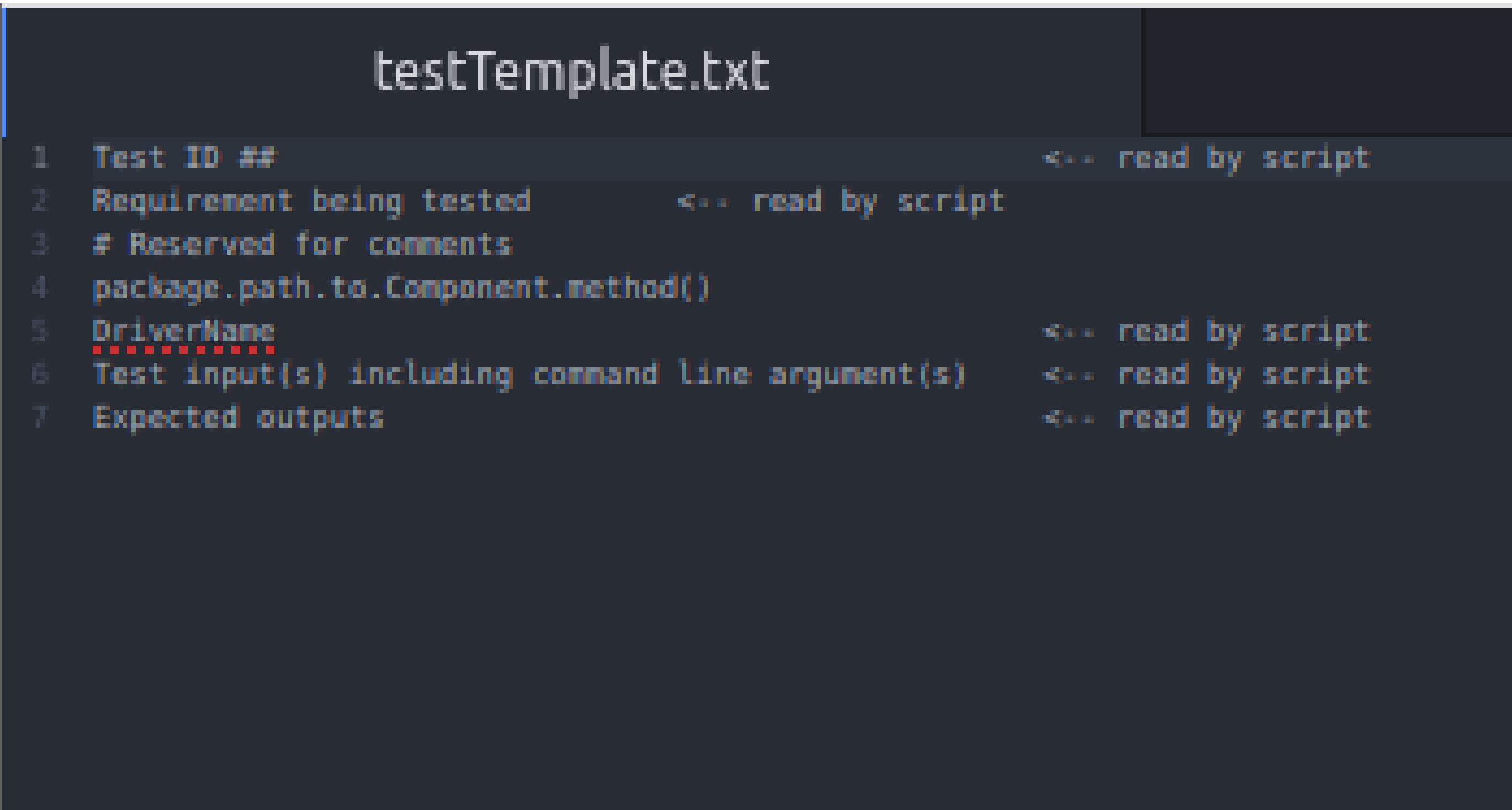


Figure 4. Test Case Template

## 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.

Case	Requirement	Method	Inputs	Output	Oracle	Result
01	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	220 136 15	#8B0FDC	#DC8B0F	fail
02	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	0 0 0	#000000	#000000	pass
03	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	255 255 255	#FFFFFF	#FFFFFF	pass
04	The method will take a valid color defined by rgb and change it to hex.	ColorConverter.rgb2Hex()	30 15 2	#0F021E	#1E0F02	fail
05	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	184 93 23	#3D19B8	#B83D19	fail
06	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xDC8B0F	rgb(15, 220, 136)	rgb(220, 136, 15)	fail
07	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0x000000	rgb(0, 0, 0)	rgb(0, 0, 0)	pass
08	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xFFFFFF	rgb(255, 255, 255)	rgb(255, 255, 255)	pass
09	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	18 52 86	rgb(86, 18, 52)	rgb(18, 52, 86)	fail
10	The method will take a valid color defined by a hex value and convert to rgb.	ColorConverter.hex2Rgb()	0xAABBC	rgb(204, 170, 187)	rgb(170, 187, 204)	fail
11	This method takes two doubles and computes the distance between them and outputs in the form of a double.	ContrastChecker.computeContrast()	50.0 20.0	1035.25	2.5	fail
12	This method takes two doubles and computes the distance between them and outputs in the form of a double.	ContrastChecker.computeContrast()	-50.0 0.0	-24.75	-999.0	fail

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

Tanaguru

Dr. Jim Bowring

