Chapter 2

Experiences

We struggled with deliverable 2 for several reasons. We still feel a little unsure about the Test Template as different people on our teams had varied interpretations of what that meant. We were initially searching for a class that had at least five testable methods, which we never found, but we were informed that the methods didn't have to all come from the same file. We finally chose a file to at least base the first five test cases on, but we didn't think through needing a method that takes input. To find methods to test, we chose ColorCombinaisonImpl.java and decided on the method public boolean equals(Object obj).

Report

Once we chose ColorCombinaisionImpl and equals(obj), it became clear on what we need to test. This method checks that the color combination that is being suggested is valid. It will fail if the object is null, isn't a ColorCombinaisionImpl object, or the colors differ between objects. The method will otherwise pass.

Our five tests on this method should test the three ways to fail and cases to pass. Case 1 includes the comparison object being tested against itself, which should pass. Case 2 will be the correct object type with the same values as the comparison object, which we expect to pass. Case 3 will be the correct object, but with color values that differ from the comparison object, which will fail. Case 4 will pass null into the method, which will fail. Case 5 will fail, since the input won't even be the correct object type.

Test Plan

The Testing Process

We will have five test cases per method and we will have five methods to test.

Requirements Traceability

Test Case1: The object constructed from itself will equal an input of itself, returning true.

Test Case2: The input of the ColorCombinaisonImpl(with same parameters as self) will return true.

TestCase3: The input of the ColorCombinaisonImpl(with different parameters as self) will return false.

TestCase4: The input null will return false.

TestCase5: Input of an object other than ColorCombinaisonImpl (String in this test case) will return false.

Tested Items

ColorCombinaisonImpl.java

Method: equals(Object obj)

Testing Schedule

10/01/19 - Design 5 test cases on one method

10/18/19 - Create drivers for first 5 test cases and design 10 more test cases

10/25/19 - Create drivers for test cases 6-15 and design 5 more test cases

10/29/19 - Design and build test case automation

11/01/19 - Create drivers for test cases 16 - 20 and design 5 more test cases

11/08/19 - Create drivers for test cases 20 - 25

11/12/19 - Design 20 additional test cases

11/19/19 - Design and inject 5 faults into code so that at least 5 test cases fail

Run all tests as they are created and run previous tests.

Test Recording Procedures

The output of the tests will be written to a file that we will use diff against testCaseOracle files to determine if the tests pass or fail.

Constraints

Time is the biggest constraint working against us. Between classes and work, finding time to work together can be difficult.

Test Cases

In these test cases, the comparison object will be constructed as ColorCombinaisonImpl(Color(0,0,255), Color(255,0,0), 50).

Test Template:

1. Test ID

2. Requirement being tested

3. Component being tested

4. Method being tested

5. Test input(s) including command line argument(s)

6. Expected outputs

Case 1:

1. 01

2. The object itself, equals itself.

3. ColorCombinaisonImpl

4. equals(obj)

5. this

6. true

Case 2:

1. 02

2. The object must be of type ColorCombinaisonImpl and valid colors.

3. ColorCombinaisonImpl

4. equals(obj)

5. ColorCombinaisonImpl(Color(0,0,255), Color(255,0,0), 50)

6. true

Case 3:

1. 03

2. The object must be of type ColorCombinaisonImpl and valid colors.

3. ColorCombinaisonImpl

4. equals(obj)

5. ColorCombinaisonImpl(Color(255,0,0), Color(0,0,255), 50)

6. false

Case 4:

1. 04

2. Null is not a valid object.

3. ColorCombinaisonImpl

4. equals(obj)

5. null

6. false

Case 5:

1. 05

2. Only the ColorCombinaisonImpl object is valid

3. ColorCombinaisonImpl

4. equals(obj)

5. String "test"

6. false