

**Team: 404 Error**

**Members: Ricky Ramos, Matthew Schwarz, Matthew Kay**

**10/03/2017**

## **Deliverable #2**

### **Project: OpenMRS**

**Task:** Produce a detailed test plan for five of the eventual twenty-five tests that will be created for our project.

#### **Testing Schedule:**

10/03/2017 - Produce a detailed test plan of five test cases

10/31/2017 - Design and build an automated testing framework to implement our test plan

11/14/2017 - Complete the design and implementation of the testing framework. Create 25 test cases that the framework will automatically use to test OpenMRS

11/21/2017 - Design and inject 5 faults into the code we are testing that will cause at least 5 tests to fail, but not all

**Constraints:** The team ran into no significant issues when building openMRS project and getting the project to run. We originally were working on the Teammates project, but we were having issues getting it to build on some team members computers, and were also having a hard time finding appropriate methods to test, so we decided to switch to the openMRS program.

**Report:** The following are detailed test plans for our current project, OpenMRS. We have specified 5 eventual test cases that we have developed for our software. We have created and tested one of our test cases, the `dateToString()` method in the `FormUtil.java` class. The test ran successfully and matched the expected outcome. In the tables below, we have included the requirement being tested, the class, the component being tested, the test inputs, the expected result, and the test environment. We have not yet produced and run the tests other than for `dateToString()` yet, but that is something that we plan on doing before the next deliverable.

**The Testing Process:** We plan on creating an script that automatically runs the tests created from the openMRS source code. The script will input the test inputs to the test cases, and compare the result to the expected outcome from running the test.

**Test Case 1:**

<b>Requirement Being Tested</b>	Testing that the truncateToSeconds method drops the milliseconds from the input Date object and returns a Date without milliseconds.
<b>Class</b>	DateUtil.java
<b>Component Being Tested</b>	truncateToSeconds(Date date) in the DateUtil.java class
<b>Test Inputs</b>	2017-10-01 16:52:32.128
<b>Expected Result</b>	2017-10-01 16:52:32
<b>Test Environment</b>	OS: Linux Ubuntu 16.04 IDE: Eclipse

**Test Case 2:**

<b>Requirement Being Tested</b>	Testing that the compareNatural method will return a 0 if the two strings are equal
<b>Class</b>	NaturalString.java
<b>Component Being Tested</b>	compareNatural(String s, String t, boolean caseSensitive, Collator collator) in the NaturalString.java class
<b>Test Inputs</b>	compareNatural("test", "test", True, NULL)
<b>Expected Result</b>	0
<b>Test Environment</b>	OS: Linux Ubuntu 16.04 IDE: Eclipse

**Test Case 3:**

<b>Requirement Being Tested</b>	Testing that the compareNatural method will return a positive number if s lexicographically follows t
<b>Class</b>	NaturalString.java
<b>Component Being Tested</b>	compareNatural(String s, String t, boolean caseSensitive, Collator collator) in the NaturalString.java class
<b>Test Inputs</b>	compareNatural("test", "string", True, NULL)
<b>Expected Result</b>	Positive Number
<b>Test Environment</b>	OS: Linux Ubuntu 16.04 IDE: Eclipse

**Test Case 4:**

<b>Requirement being tested</b>	Testing that the compareNatural method will return a negative number if s lexicographically precedes t
<b>Class</b>	NaturalString.java
<b>Component Being Tested</b>	compareNatural(String s, String t, boolean caseSensitive, Collator collator) in the NaturalString.java class
<b>Test Inputs</b>	compareNatural("test", "white", True, NULL)
<b>Expected Result</b>	Negative Number
<b>Test Environment</b>	OS: Linux Ubuntu 16.04 IDE: Eclipse

**Test Case 5:**

<b>Requirement being tested</b>	Testing that the dateToString method returns the correct string of the date in ISO 8601 format by adding a ':' in the time zone section.
<b>Class</b>	FormUtil.java
<b>Component Being Tested</b>	dateToString(Date date) in the FormUtil.java class
<b>Test Inputs</b>	dateToString(new Date()) //the new Date object will create a date with the current time, for testing purposes the date that was created was: 2017-10-03T12:08:12-0400
<b>Expected Result</b>	2017-10-03T12:08:12-04:00
<b>Actual Result</b>	2017-10-03T12:08:12-04:00
<b>Test Environment</b>	OS: Linux Ubuntu 16.04 IDE: Eclipse