Methodology

First I started with just blindly testing URLs that I knew should work. I created a new function called testManualTest(). All I did was create strings of URL and tried to find any errors at all. I created about a dozen tests. Start with something simple as,http://www.google.com, then I moved on to varying the URL by adding request information and ports http:/www.google.com:32/test?arg=a . I was just looking for any little thing that might bring about a failing test case. What finally gave me first bit of clues was when I tried test://www.google.com/test?action=view . That passed so I knew there was any issue with using http://. I started to see that you could add anything before the :// and you would still get a pass.

For the partition sections I was really confused. So I did a google search and look at the site forum. I ended up creating two functionstestYourFirstPartition() and testYourSecondPartition(). These were simply to assure myself of correct and incorrect URLs that I know would mostly pass or fail with the supplied codebase. Each function had an array with URLS, incorrectURLS[] and correctURLS. Those were asserted to verify I was pretty much on the right track.

Lastly I had a programatic function called testIsValidPrograming(), that would test a few variation of URLs that I had seen so far pass or fair from testManualTest(). I created two arrays with parts of URLs, combined them in a loop and then checked if they were valid, String[] prefix = {"http", "http://", "test","test://"," "};

String[] mainURL = {"www.google.com", "www.google.com:32", "www.google.com/test", " "};

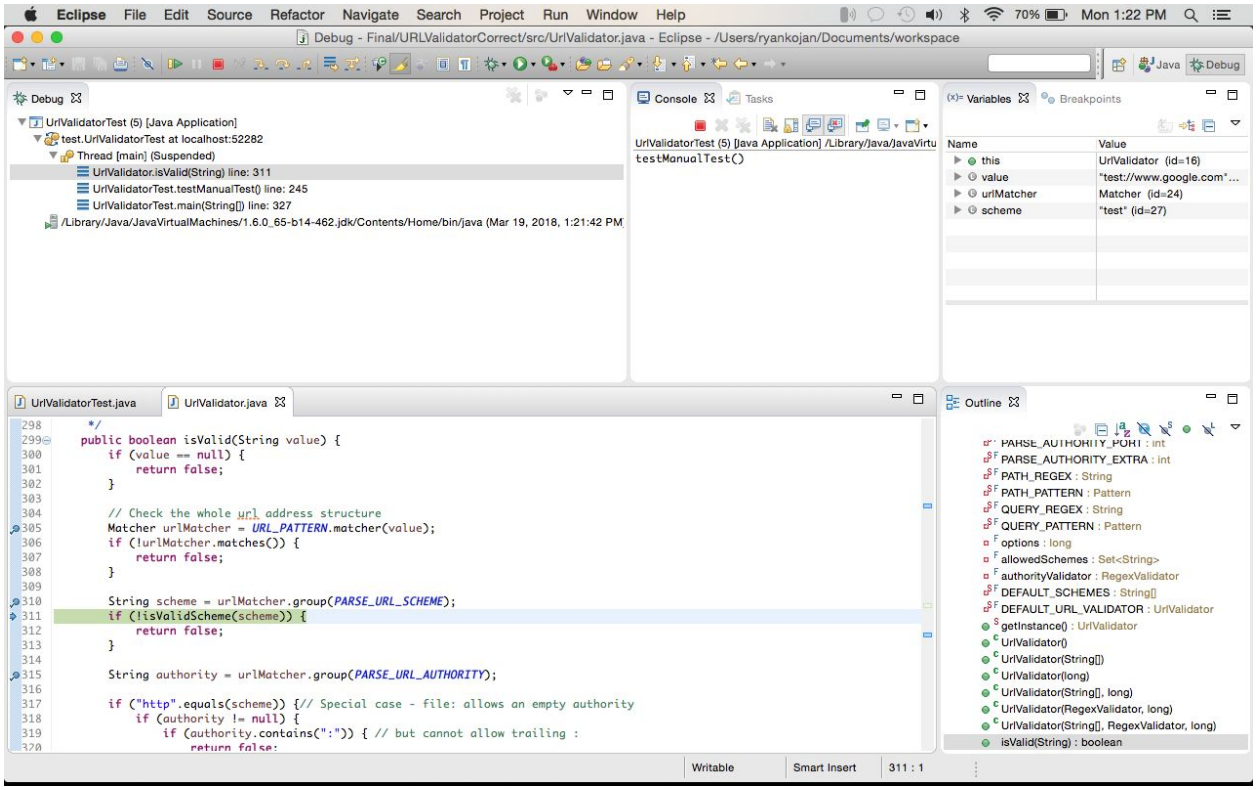
This function helped me determine was really going on in the error version of the codebase, by these test cases that passed: http://www.google.com
http://www.google.com/test
http://
test://www.google.com
test://www.google.com:32
test://www.google.com/test

From what I found here you can have any text before :// and the URL will pass. If you have http:// before and port anywhere in the URL it will fail. This was the information I needed for debugging and bug hunting.

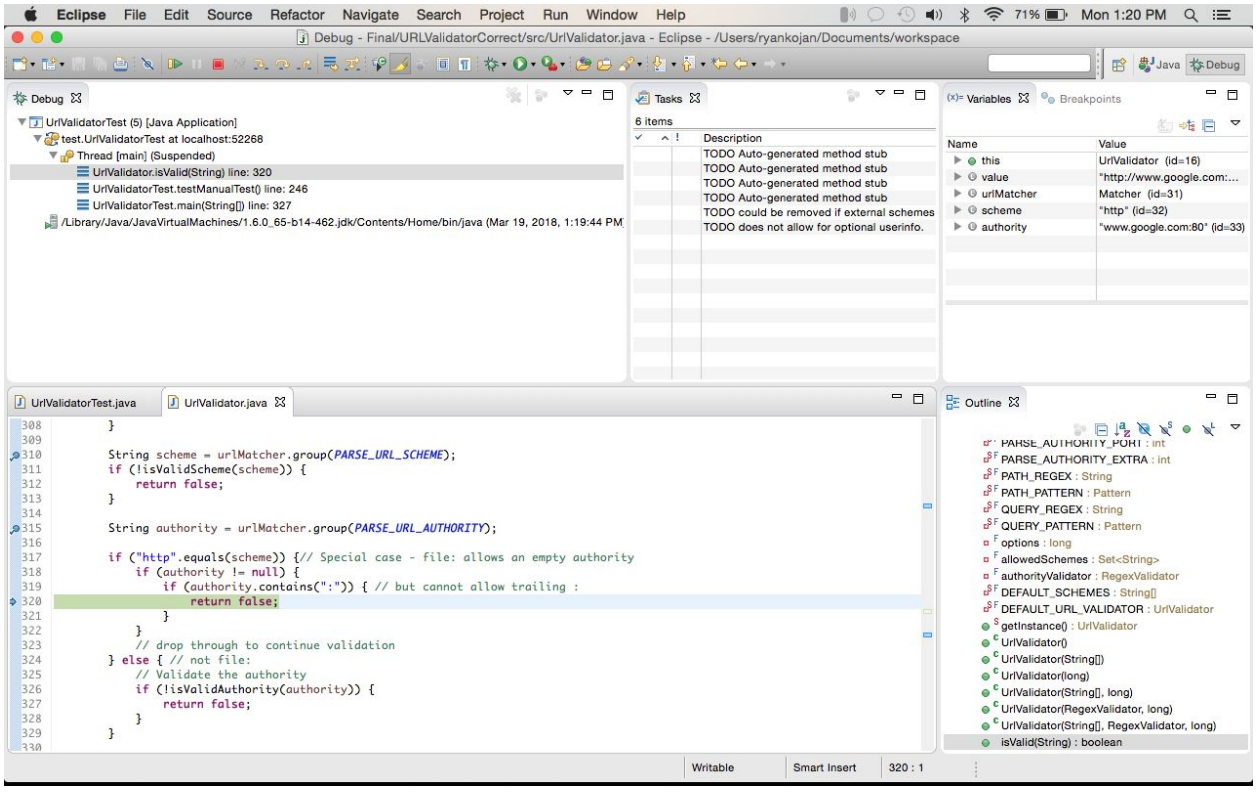
Bug Report

1) The first bug is being able to add any kind of text before :// . I found this spot in the code by using the Eclipse debugger. I set breakpoints in the isValid() function at each

conditional check and I found it be failing here. I'm not 100% what's causing it



2) The second bug is when you have for http://www.google.com:30 that will fail. You should be able to include a port number in a URL call



Debugging

To find the spots of code that needs to be refactored I used the build it Eclipse debugger. I used my first function testManualTest(), but added my breakpoints in the isValid() function at each conditional check. The two lines that 311 and 317. Those lines use a build in library for Java that checks URL's and not sure what that code is doing, but when the URL from my test is being sent to that library there is an issue with what is being sent. I don't the exact cause in isValid() but those are the areas to start refactoring.

Team work

Again for this part I had to work alone because I tried to reach out to the members I talked with at the start of the term, and they would not return my emails. I decided to not wait around for them or try to join a new group. All in all there wasn't that much work that I could do on my own. I spent maybe a full hour from the first test I wrote until the point I started writing this write up. For this to be a real group project there needs to be much more work to make it feasible.