Test Strategy

This document will detail the test strategy for developing the My Webserver application for use in the IoT. For writing this document, and therefore planning how to go about testing this designated abandonware, I have consulted Wikipedia, guru99 LIST MORE STUFF HERE AS IT COMES UP

In this document I will specify my planned test methodology for this project, using the headings that I have found in my research (see sources above).

Test Levels

This section is concerned with to what levels will be carried out for this project. I will try to describe my thinking for why I make these choices.

All TDD needs to have at least unit testing, so that is a given. However, I believe there can be value to trying to do some integrated testing to test things like the persistence of the access log.

Roles and Responsibilities

There are two stakeholders in this project. Myself as the developer, and the course management as the ultimate arbiters as to whether the project is fulfilling requirements. So I will:

- Make the test strategy, and amend it when there is need to do so.
- Explore the server program and its tests to understand it and continue building on it.
- Further develop the server's code
- Create further tests as the code grows

The course management will:

- Evaluate the test strategy
- Evaluate the testing documentation
- Evaluate the final software

Environment Requirements

To run this web server the environment must be capable of running Java and have it installed.

Testing tools

Will investigate further for use of tools for this project.

Risks and Mitigation

One major risk with this project is my own inexperience with using testing tools in any rigorous manner. I am very liable for making mistakes or missing something with my tests. To mitigate this, I will try to find best practices when I can.

The other major risk is that I am very short on time with the initial deadline for this being due in just a few days, which limits my ability to properly research good testing methodology, making my work liable to weird, ad-hoc solutions. The mitigation would have been better planning, but at this point I can only play the hand I dealt myself.

Test Schedule

First, the program must pass all unit tests. Then, I shall try to run an integration test to see how the program runs in a controlled environment. As handling heavy loads was one of the requirements, I shall also try to find and use a tool for load testing.

Test groups

I am still exploring the server's code, so expect this to grow as I continue reading.

Initial candidates for test groups:

- File readers
- FILL IN LATER

Test Priorities

First, study and run existing unit tests to get a feel for the current state of the software

As of now, all tests pass. But some tests are empty and should get an automatic fail as things stand now. Furthermore, the old version of mockito gives error messages (or, at least, my research indicates it could be the old version causing it). I would like to try to update the server to use newer versions of Java and Mockito etc. I would also like to implement maven for easier handling of dependencies like Mockito and more.

Amend and add to unit tests until full test coverage has been attained and all unit tests pass

Create test for UC1, including alternate test scenarios, and make them all pass

Create test for UC2, and make them all pass

Create test for UC3 and alternate scenarios, and make them all pass

Finally, try to load test the application.

All of these tests must pass according to the requirements. What exactly constitutes an acceptable responsiveness under load needs to be specified.

Test Status Collections and Reporting

Ideally, I would like daily status collection from tests to see how the project is faring day to day. Due to my failure to allocate sufficient time to this project, this will be deficient, but I plan to work harder on this now.

Test Summary

A test summary should be submitted weekly. Due to the accelerated timeplan I must implement to finish this project between now and the final deadline date, I will likely need to submit more tests than usual.