Journal 5-2: Software Testing Techniques

Dhiraj Gurung

Southern New Hampshire University

CS320

Toni Farley

February 5th, 2025

**Software Testing Techniques**

While completing the milestone assignments I needed to ensure that my code was functioning properly by running tests on them. One software testing technique that I used was using JUnit tests to test individual components of the software. For example, for the Appointment Class, I made sure to test the setters separately to ensure that they were functioning properly. Then I made sure to create separate tests for each requirement like testing to see if the program could handle null values or past appointment dates. The practical use and implication of testing software at the individual level is that you can detect errors and pinpoint the source of those errors much more easily when you test blocks of code rather than the whole program. In conjunction with unit testing, I was also applying acceptance testing principles. I tested for each of the different requirements given. I made sure to test the null values for the invalid appointment ID, date, and description. I did the same tests to see if the program could handle errors when too many characters were given for the appointment ID and description by using realistic values. The practicality of acceptance testing is to emulate a real-world scenario and not just test for edge cases. By using values a user might enter you can check to see if the program would work in a real-world environment.

There are a lot of different software techniques and I did not use them all. An example of a technique I did not use would be security testing. Although I may have used private methods for the IDs to ensure that the IDs couldn’t be changed by the user, I did not code or test the program with the intention of creating a secure program. Testing and coding for security would help to keep the databases of these IDs secure from harm in case someone with malicious intent wanted to access the appointment data. Additionally, there was no usability testing done to ensure a user-friendly program. Doing so would make it easier for the consumer to create accounts and use the software in their day-to-day life. If this was a program created for the masses by a private company then this may improve the customer’s impression of the company. However, the focus of this program and code was functionality. As a result, when I coded the program I only ensured that the program could be used by me and that it met the given requirements.