Trevin Wadlington

CS 320

8/23/25

Summary and Reflection Report

In my project I used a straightforward approach to develop my classes. My main objective was to fulfill the functional requirements of the assignment. My unit testing the different tasks of each assignment. The contact service class had to include a few operations for contacts such as creation, updating and deleting. To ensure proper usage the tests check for the properties of input. I used JUnit tests in this one. For example, I tested the duplicate contact IDs because the requirements specified uniqueness.

The Task Service dealt with our descriptions. My tests confirmed that descriptions were within the required parameters and validated input appropriately, including error handling for invalid values.

The Appointment Service class focused on appointments and actions involved. Testing had to be inserted to make sure valid appointments were being made. Things included in this are time and duplicate handling. The requirements were the main gps for my development path. Included in the rubric required things to be unique. The duplicate test handling was implemented because of the requirement.

My percentage wasn’t as high as I would have liked. This is an area I hope to improve in as I progress in this discipline. I know that I developed tests that were somewhat efficient based on my percentage. The core foundation was there enough to have confidence in the functionality. I implemented good tests to check for. The tests were efficient enough to verify the main behaviors. An improvement in coverage is an area of growth for me.

Writing the JUnit test improved my understanding of how to apply them. I learned a lot taking on the task of meeting the requirements. I spent a lot of time within the module resources and online learning to be able to complete this assignment. Learning these commands helped me insure I was using standard commands. I would test multiple attributes to correctly cover all boundaries. This helped me become more comfortable with structuring test cases in an efficient way.

The main testing technique I used in my project was Unit Testing, which focused on testing individual actions and ensuring they behaved as expected with valid and invalid inputs. I did not use a lot of regression testing. My mindset was building functional and technical work. I implemented things one at a time and used simple methods to reach my goals. I assumed anything could break the system because it is true. A task such as adding contact should not affect unrelated things. To limit bias, I wrote tests as if I were an external reviewer. As a developer, I recognize that bias can creep in when testing my own work, I may unconsciously avoid writing tests for scenarios. For example, without careful thought, I might not have tested creating an appointment on today’s date, assuming only past dates were invalid.

Quality was an important focus for me. It is important to ensure you have quality coding. I found that the things I had to go back, and edit were much easier to find with the proper principles. Applying the proper testing principles made issues much easier to identify and fix. I used test driven development to avoid issues later.

Overall, this project showed me how important testing is. Quality coding practices and approaching testing cautiously. I want to continue to improve my skills and incorporating more techniques into my future projects.