Andre Freitas

10/19/2024

CS320

Project 2 Summary and Reflections

In all three features, my tests were done to align with software requirements to the fullest. I made sure that each requirement that was requested was included in the software and tested against. For example in the Contact Java I made sure to have objects for the contact ID, first name, last name, phone number, and address and that none of these could be blank. In the ContactService I made sure that the contact ID could not be modified, that the names, numbers, and addresses could be modified, added and deleted, and that the array would update when they were modified. My JUnit tests went perfectly. I know they were effective because each JUnit had a sequence of code it matched up with so there were no extraneous or missing JUnits. The JUnits each had a green checkmark next to them, showing that the expected result from the test, whether I was testing if everything worked or if the specific portion of the code failed. When there were errors I would go back and fix them and I made sure to make the code as efficient as possible. Rather than putting separate lines for asserting or throwing exceptions, I included them in the same code that would test adding a contact, such an on line 32 on the ContactServiceTest.java.

For this project I used unit testing with the JUnit tests. Just as the name would imply, this tests specific units of code that are executed to check if they fail based on the structures set by the code and what is fed into them by the JUnit test. This is done to make sure that the code, or the specific section being tested, functions as intended. It’s a small part of coding that lets a developer tackle one section and issue at a time without needing the full code to test it out. (Gaba, 2024)

A type of testing I did not use was integration testing. Integration testing tests the interaction between different modules of a system or project and how it can transport data in between. All of my tests were JUnit tests, each being individualized tests that did not rely on another piece of the code to see if they functioned correctly. The closest I came to that was setting private objects but that was still in the same piece of code. Although integration testing would be required in a full project, it was not needed in this one. (GeeksforGeeks, 2024)

Regression testing is another type of testing that I did not use but it very important for software development. Regression testing is the testing of code that has had a change added to it. This is done to make sure that the changes that were made don’t break the code and make it unable to be used. Finding out that the necessary change for one part of a project is making a different part not function has to be done before rolling out any new update or function or else it could be devastating to the user base and result in a bad reputation for the software. (GeeksforGeeks, 2024)

The mindset I had for my projects was a practical one. I did not fuss about the full project’s scope as I was tasked with handling pieces of the code. This is how I approach most software projects. Rather than tackling everything at once or getting bogged down, I do a single part and move on to the next, do the next, and so on. I test each piece to make sure there is nothing that is missing or throwing an error before finishing. But I also know that I have to go in order as well, so I went down the list of things to do rather than find what might personally be easiest for me as I did not want to have any part of the code not function because I missed something requested of me previously.

I did not have a bias issue myself. I tend to approach things with the mindset that if I know something, someone else might know more. When I find or am told that things could be more efficient, I take the advice very easily. I can see how testing could be affected by bias. Someone saying that you could efficiently maximize your code by cutting out lines or adding throws or exceptions somewhere other than where they currently are could be irritating as no one likes to feel criticized. But bias really does not have a place in coding. If the code does not work, it does not work and needs to be fixed. Requirements need to be met, and they cannot be ignored due to ego or bias.

Discipline is important in any field and when your field can affect hundreds to thousands of consumers or coworkers it is doubly important. Maintaining your discipline in regard to clear, readable, functional code is paramount for a developer. Should another developer need to edit code, having the above standards makes their job much easier. Besides which, cutting corners and taking inefficient shortcuts is simply sloppy work. My own worth ethic would not allow me to have low standards with what I make, and if not that then just knowing that there are others who need to be able to see my work makes me want to do better. There is also a horror story that my friends know about regarding a developer trying to make a game with horrifically inefficient code that makes me want to do better if only to not be at his level.

References:

Gaba, I. (2024, July 9). *What is junit: An overview of the Best Java Testing Framework*. Simplilearn.com. https://www.simplilearn.com/tutorials/java-tutorial/what-is-junit

GeeksforGeeks. (2024a, June 13). *Regression testing - software engineering*. https://www.geeksforgeeks.org/software-engineering-regression-testing/

GeeksforGeeks. (2024b, September 26). *Integration testing - software engineering*. https://www.geeksforgeeks.org/software-engineering-integration-testing/