Project Two Summary and Reflection Report

Bijal Chauhan

CS-320 Software Test Automation & QA

Prof. Neil Kalinowski

Summary:

The Junit test has been used to check the functionality of the features to determine any issues in compilation that can cause program to run. The use of my testing is to ensure that all the software requirements has been fulfilled. The testing includes unit test to determine how each classes run. I have also used integrated testing to determine how the classes run when they have parent and sub-classes. Running different tests helps identify the errors that was missed by one test or other. Running the Junit test by applying common variables and outcomes would help determine if the program was functioning accurately. Quality tests returned functions that enabled the developer to determine the sources and causes of error in the program that could help to have a fully functional program. Simple codes help to create a clean effective system.

Reflection:

Techniques that have been used includes the Junit testing technique was applied in the code by creating a new Junit test case in the Java IDE after adding the Junit Library in the project. The test case was then run, and the test would return a true result if the objects were equal, and if the passed object is null or the getClass method is not equal, the test would return false result. Another test used is structural unit testing to ensure the reliability of the software code. Exploratory testing was also used to inspect the code to determine if it follows standards of coding and the outcome is fully functional code.

Techniques that have not been used includes static testing where the code is tested without executing it, dynamic testing where the code is tested while executed, and manual testing where the code is manually tested.

While reviewing the code, it is very important to test the basic things like creating operators, constructors, loop, method, arrays, objects, and the ability to move further with these basic applications. When testing code any concerns would arrive would be the lack of these basic things or how to run the test.

Being a software engineer, one must be committed to quality of work because it shows how one can have an ability to fulfill the deliverables. While writing a code, there is a need to have team collaboration to reduce the technical debt. When developing new features, one must ensure that it functions well and do not create an error in the functionality of the system.

Mindset:

I have a mindset of programming quality during the developmental process. During programming phase, I ensured that all the attributes of my code were logical and fully functional. It is very important to have qualitative work to reflect one’s skills are up to the quality requirements. It is important to understand the complexity of the code and what I have learned from it because it showcases the skills, I can come up during the software development process.

As a programmer, testing own code can be easy to bias which can result into releasing the software without passing the testing standard. We should always try to test the code as if we are testing for someone else that way, we cannot bias the software. Being a programmer, we should ensure that the quality of the work we provide to client directly affects the image we can make in the industry that can help build the good future. Qualitative work can also help us to boost the self-esteem within that eventually help to do more good work in future.

Cutting corners is also one of the important aspects that should be avoided in programming field. When you cut the corners, it means that you are trying to cheat yourself and harm your standing in the field. Always try to avoid any errors in the software so that you do not run into any problems in long run. Ensure all the work done meets the quality requirements that can benefit us from future liability because we will be responsible for any issues that may arise, and we will be responsible to fix those issues. We can create a dept for ourselves if we do not deliver the client’s software requirements. As a result, we will be responsible to fix any issues arise in future.