Joseph Manzano

April 23, 2023

CS-320-J7991

Southern New Hampshire University

To describe my unit testing of pros for each of the three features a straightforward answer will be the best way of me describing it. The extent of my approach being a line with the software requirements is absolute. The purpose of my efforts programming solutions was to create a program that will meet expected requirements from the customer. The requirements being as they are define each step I take when developing programs. They are the directions in which the program is developed and the tests are created. Understanding the requirements and executing their creation is the key to success. The JUnit testing is measured in percentages. To check accuracy and overall completeness of the program. Percentages represent the amount of coverage that design covers. High percentages are best and 100% is full completion. The percentages you’re testing creates is your measurement of quality for coverage percentage.

The J unit tests when created were designed to me a satisfactory goal. With expectations being met in mind, the program was developed in a conscientious manner where the goal was the center focus. To ensure code was technically sound, tests were created to ensure accuracy and create confidence when demonstrating the program. Classes were created with functionality in mind that illustrate accuracy and dependability. The efficiency of code created is to minimize the amount of steps and effort for the hardware to process. To choose various lines of code that demonstrate efficiency is a difficult task as the lack of lines of code and functional steps is what creates the efficiency. Using tools and libraries to create a functioning program is the most efficient way of development.

A tool that can be utilized when developing code is to create unit tests. The unit test is software functionality that checks parts of the developer’s software. Unit tests are a fundamental way of correct errors during a development process. Creating unique ideas that are 10 characters. Long was the goal of the software. Meaning, the requirements of 10 characters was necessary in completing this program, so enable to ensure 10 characters with met a test, was written for an object when created.

Unit testing is a form of software that functions by testing a minimal size of code multiple times. Expectations and requirements needed to be met so the unit test will correct the developers creation, a long during development to ensure the program meets all request from the customer.

Due to the program, being smaller and precise in its requests, multiple forms of testing were not needed. Integration testing is used to test. The end. Product went to or more software units are being corrected. As mentioned before, this product was not utilized, but mentioned due to its efficiency and usefulness. Reflecting on the program aids in creating other ways of executing the creation of the program. Acceptance testing evaluates the systems ability to follow functionality and requirements for business purposes. System testing is similar to immigration testing but this form of testing is done to a finalize product to ensure all requirements and unnecessary requirements are met.

The mindset needed for this project as a software tester is a pessimistic point of view. I am looking to be negative and find heirs to correct. When it comes to programming, one mistake can be catastrophic. Many horror stories, I have heard in the past have instilled a fear and sense of severity when developing. I am very appreciative of the challenges and complexity of the code I was testing. Being challenged is a real life expectation, where a consumer or customer is expecting results we must provide. As somebody who has already been in the workforce, I can say without a doubt there are days that are easy and there are days that are filled with stress and pressure. Limiting bias is a difficult task when you are testing your own code. The task is extremely difficult when you want your program to be perfect and you keep finding errors in it. The personal conflict internally is brought to my own attention when all I want is to be finished with a project and mentally all I can think of is finding ways to be finished fast. Discipline is the only way you can defy your feelings of wanting to be done or quit. Without discipline, nothing will ever truly be good in life. Cutting corners when writing or testing code as someone with experience, the feeling or mention of that action holds the same emotional response as someone committing a cardinal sin. While consequence may be minimal, in some cases, other cases may cost others lives. Avoiding technical debt as a practitioner is to be conscious of one’s own feelings and emotions. As badly as I wanted to cut corners or brush something under the rug, I must complete the program with satisfactory results. Mental breaks are an example of taking care of yourself, biologically to not have program functionality problems.