David Hammons

Southern New Hampshire University

CS-320 Software Test, Automation QA

Project Two: Testing Reflection

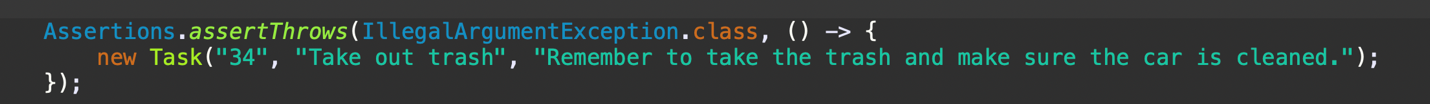
I think that my testing approach was aligned to the requirements as presented to me for the software. We had a list of items that the programs were required to create, such as having a limit of the length of characters or not being able to change the unique ID. My approach when it came to testing was to ensure that those requirements were met within the program. I immediately wrote functions that tested the requirements almost just one by one and checked to ensure that they worked and that if It didn’t meet the requirement expected of software, then it would throw an illegal argument. Once these were met, then I would try to test some other things that I felt could occur in the program. Maybe someone would try to access an index that was out of bounds of the current array? Then it would throw an error and not allow the program to progress forward. Maybe someone didn’t put anything in the field when modifying the program? I also got it to display an error if that occurs and also doesn’t allow the program to move forward.

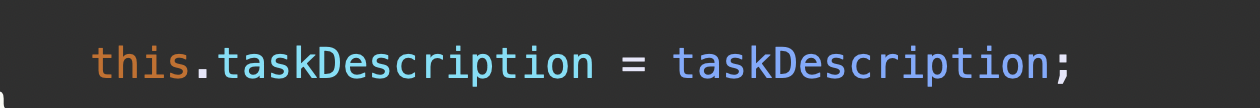
One of the things I did to make sure that my code was sound was test both the service but also test the object itself. And when I would test each function within the object, I would test the 2 main objectives of it. The first one is

A black background with colorful text

Description automatically generated

The main objective of this piece of code is that it makes sure that the requirements are met. If they are met, then the code can proceed to the second objective but if the requirements are not met, then it would throw an Illegal Argument Exception and it wouldn’t allow the code to run any further. I used these two pieces of code to test both the idea that the code could go over the character limit plus also what would happen if instead the code was NULL. 



After the normal code checks and makes sure that none of those things throw the exception then it proceeds to the next and final line of code.

It’s a simple line of code but this is what assigns the object’s “task description” element with the one that was passed into the function. The following line of code is how I test to ensure that the task description has been properly added to the object after it passes all the checks to ensure it isn’t too long or that it isn’t NULL.

When It comes to the techniques that I used when it came to writing and testing my code. I mostly employed the idea of doing a test-later technique. I wrote all of my code first and I did do little small tests within the code to test things here and there but I mostly waited until I got to the end of writing my code before I started writing the tests for my programs. Thinking back on it now I think that this probably wasn’t the most effective way to write the code as there are other, better mothods like using a test first approach. In that approach you would write the tests first and then write code to pass the tests. That way helps to keep you much more focused on the code and what you are needing to write for it to pass but I feel that it kind of keeps you within this framework of just needing to write code to pass the tests. With doing the code first approach I feel it lets you breathe a bit more with writing the code as there really aren’t much constraints except for the requirements but you aren’t coding in a way to ensure the tests pass. Instead you code for what you need and for want you want the program to do and then after that you start work on essentially checking your work by writing the tests for what you wrote and then going back to fix anything that might not be working.

I feel that my mindset when it came to starting this project was just more of a programmer and not of a tester at all. I know that testing is important but I never really employed it in any of my previous code that I wrote and it was only in this project that I really focused on it more. I think the mindset that I really needed to wrap my head around was making my code easy to read. A big part of it was that if we were testing my code and something were to go wrong then I would want to make sure that my code was easy to follow and fix to get it working. I also tried to eliminate bias in my code by taking some time away from my code after writing it. I know it isn’t the best solution but I found that I would be much more susceptible to errors in my code with the testing process if I took like 24 hours after writing my code to leave and then come back later to test. I feel the process helped me disconnect a bit more from the code I had written and allowed it to be easier to test and fix my code.