With each of the milestones we had to submit, we learned and implemented items from Junit5. In the week 3 milestone, we had to test a simple contact class and a contact service class. It had to be able to add and delete contacts as well as update contact information. The contacts also had unique IDs that were not updateable. With this information, I had to test that each method would work properly. Not only that but I also had to test that each possible issue would be handled correctly, for example, if an item was null. The week 4 milestone was a simple task application. It was very similar to week 3 but it gave me a chance to clean things up. Week 5 milestone built from what I had learned in the previous weeks. I had the same functionality, but it allowed me to research Java’s date class and how it can be used. This way I can implanted a date and test that an appointment was not set in the past.

One thing that was not being tested in input and output from a user standpoint. Though JUint5 allows me to fake input it is still skipping that actual part. User input can become an issue if not everything is covered. I tested everything that covered the requirements, but nothing that could potentially protect against hacking or something along the lines of code injection. This milestone is not really at a point to make that kind of testing plausible do to the fact that we are just testing individual classes and how they run.

This kind of testing can be used over and over. Some of the tests can be used for a variety of different classes instead of rewriting the test all over again. This was proven by how we, over the last few weeks, did 3 completely different applications. They were all unique in what they did, but had similar functionality, therefore used very similar tests that were able to be reused and modified for the new purpose. Some applications may need an extra method or two that would have to be tested, but something like adding an object to a list is easily reused.

**Citations**

García, Boni. (2017). Mastering software testing with JUnit 5. Packt Publishing.

Hambling, B., Morgan, P., Samaroo, A., Thompson, G., & Williams, P. (2019). Software testing: An ISTQB-BCS certified tester foundation guide - 4th edition. BCS, The Chartered Institute for IT.

*Junit - Test Framework*. Online Tutorials, Courses, and eBooks Library. (n.d.-a). https://www.tutorialspoint.com/junit/junit\_test\_framework.htm