Shayna Mitchell

SNHU

Week Four

Unit Testing

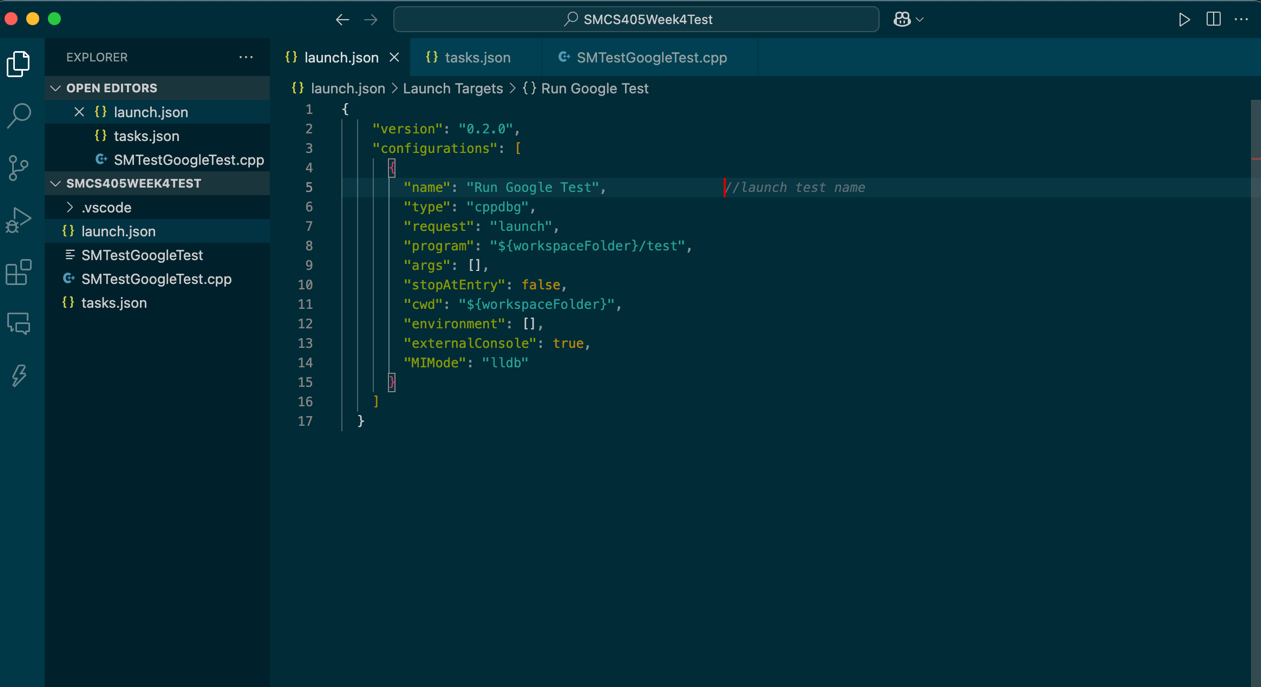
03/29/2025

CS 405 Secure Coding

Unit Testing is a really great way to make sure a program is doing what you want it to do essentially. It is smart to add these tests in as your work on your project to help you determine whether the code is performing as you want it to. Sometimes you might think the program is performing a specific process, when it is actually doing quite the opposite. Placing these checks throughout the code can help catch errors or thrown exceptions before you get the chance to even become confused. You can make sure processes are running as expected throughout the entire project.

Initially I did not quite understand why to add these in or how to ask the program to check things for me. I definitely had some typos and process checks that did not make sense initially. Setting up my testing framework did take longer than I had expected, and I needed to change some local settings as I am using a different computer than usual. I initially needed to make sure my project was calling the correct library paths, as well. Another thing I initially did wrong was I placed size and resize incorrectly in my checks. Once going back and fixing these mistakes I was able to run the checks properly. Lastly, I did not call my helper function correctly in CheckMaxSize and caused the program to crash. This was fixed by not specifying count = 0. After working thorough all of these tests, I now understand much better how to see whether I am missing anything while I code rather than wondering later why my program does not work. Knowing exactly which areas to examine when an exception is thrown is extremely helpful.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.