Module 4 Unit Testing

A screenshot of a computer program

AI-generated content may be incorrect.

In this project, I created a series of unit tests using the Google Test framework to check the behavior of a std::vector collection. First, I set up a test class that included a helper function to add random entries to the collection. The tests included both positive and negative scenarios. Positive tests verified that the collection behaves as expected when adding elements, resizing, clearing, and reserving space. For example, I tested that the collection’s size increases when values are added and that it can be cleared properly. Negative tests were designed to handle error situations, such as checking that accessing an out-of-bounds index throws an exception.

I also created custom tests, like checking that the collection can handle a large number of entries and ensuring that adding a negative number of entries results in an error. The tests were structured with clear names to describe the purpose of each one. After writing the tests, I ran them in my development environment, captured the results, and fixed any issues that arose during testing. Overall, this process helped ensure that the collection functions correctly and handles errors in a safe way.