

TED UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

SENG453/SENG311

Software Quality Assurance

Assignment 3 Report

Integration Testing

Hilal Yurtoğlu

17386054060

Table of Contents

[1.Introduction 3](#_Toc197876072)

[2.Call-Graph 3](#_Toc197876073)

[3.Implementation 4](#_Toc197876074)

[a. Course & CourseList Integration 4](#_Toc197876075)

[b. Course & GPACalculator Integration 4](#_Toc197876076)

[c. CourseList & GPACalculator Integration 5](#_Toc197876077)

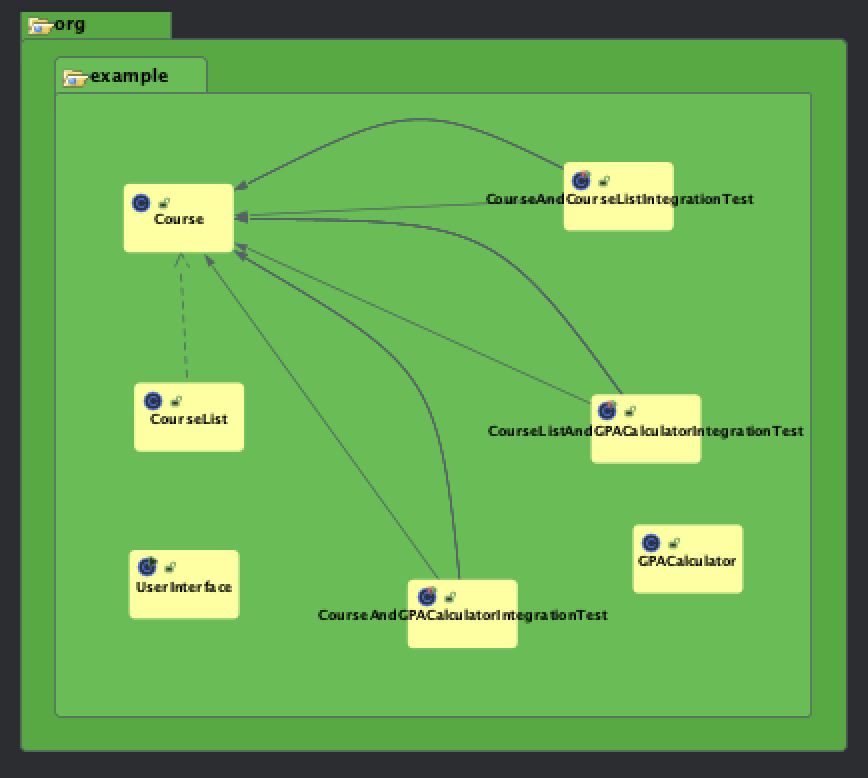
[4.Conclusion 6](#_Toc197876078)

# 1.Introduction

This report documents the integration testing process applied to the GPA Calculator system developed as part of a software engineering course assignment. The system consists of three primary classes: Course, CourseList, and GPACalculator. The aim of these tests is to verify that these components work correctly together under realistic conditions using the JUnit 5 framework and Mockito for mocking dependencies.

# 2.Call-Graph

The integration interactions between the components can be summarized in the following call flow:



This structure shows that the test classes act as clients which initiate calls to both CourseList and GPACalculator. Each course object is mocked to return predefined values, enabling isolated and controlled tests.

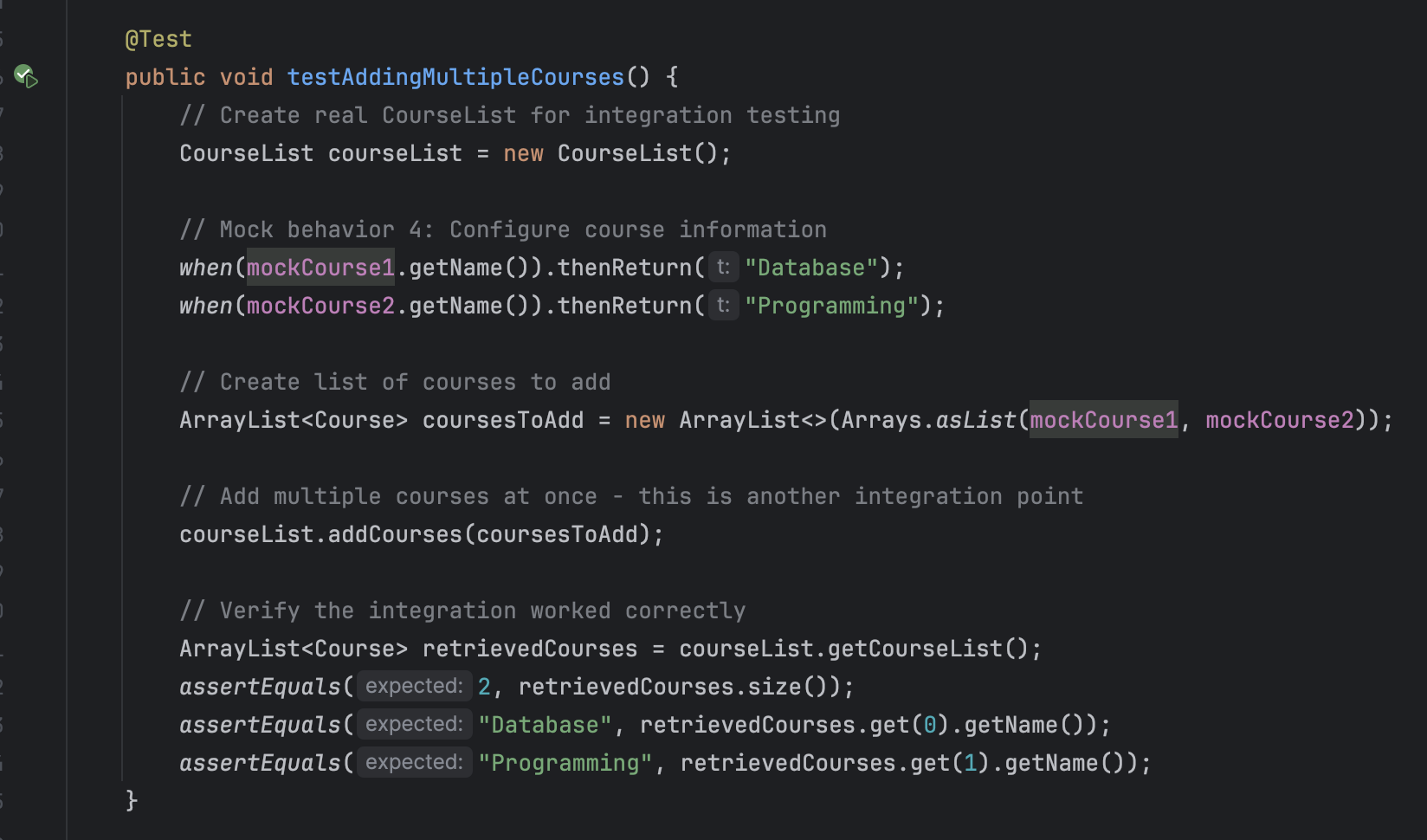
# 3.Implementation

The integration testing was divided into three categories:

## a. **Course & CourseList Integration**

This test validates whether courses (mocked using Mockito) can be added and retrieved accurately from a real CourseList object. It includes:

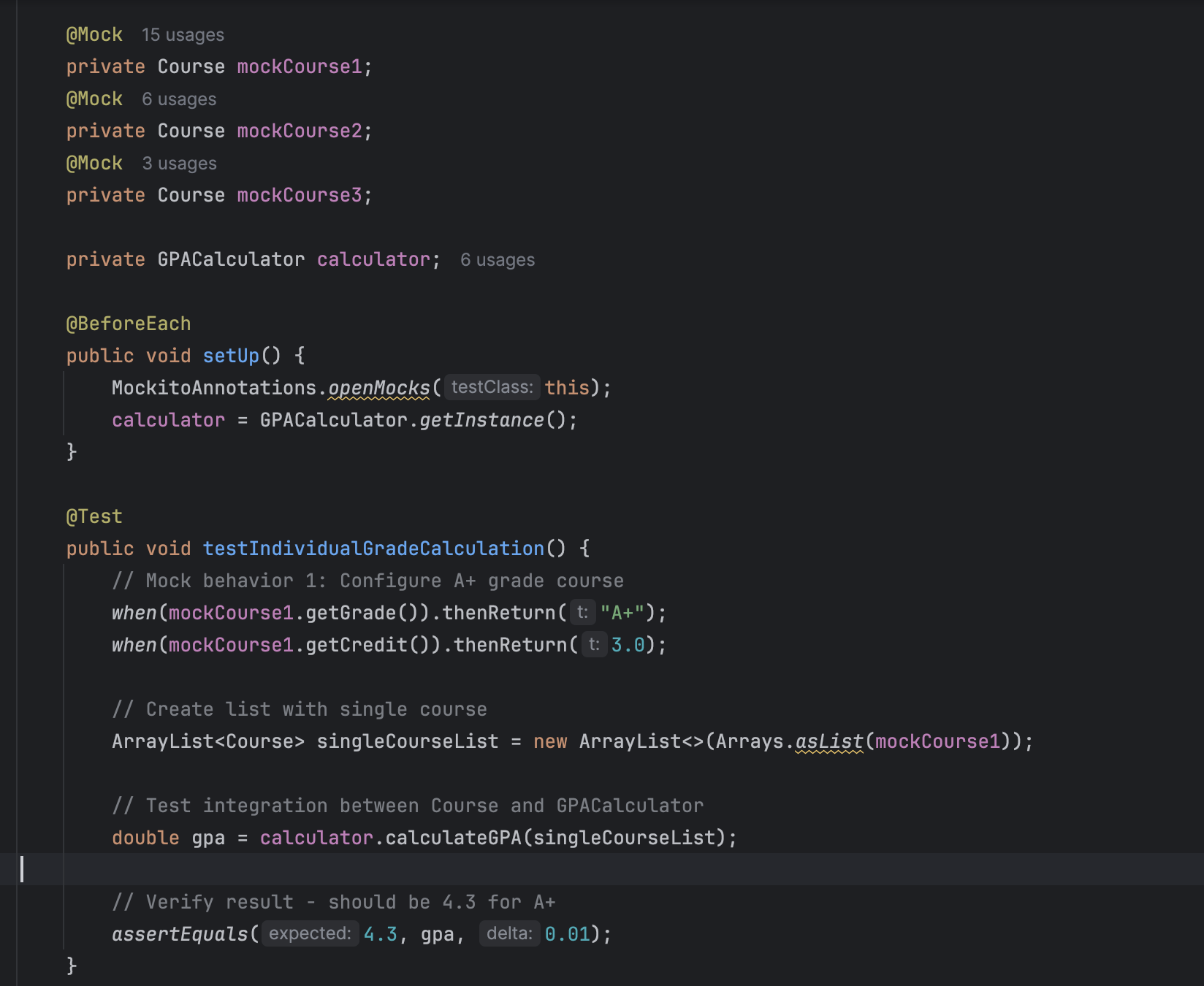
* Adding individual and multiple courses
* Retrieving course properties such as name, credit, and grade
* Ensuring exceptions are thrown when invalid inputs (e.g., null or empty lists) are passed



## b. **Course & GPACalculator Integration**

This test assesses GPA calculation using mocked courses with predefined grades and credits. Various grade cases are tested:

* A+, A, B, etc., with expected GPA values
* Invalid grades (e.g., “Z”) or invalid credits (e.g., 0.0) triggering exceptions

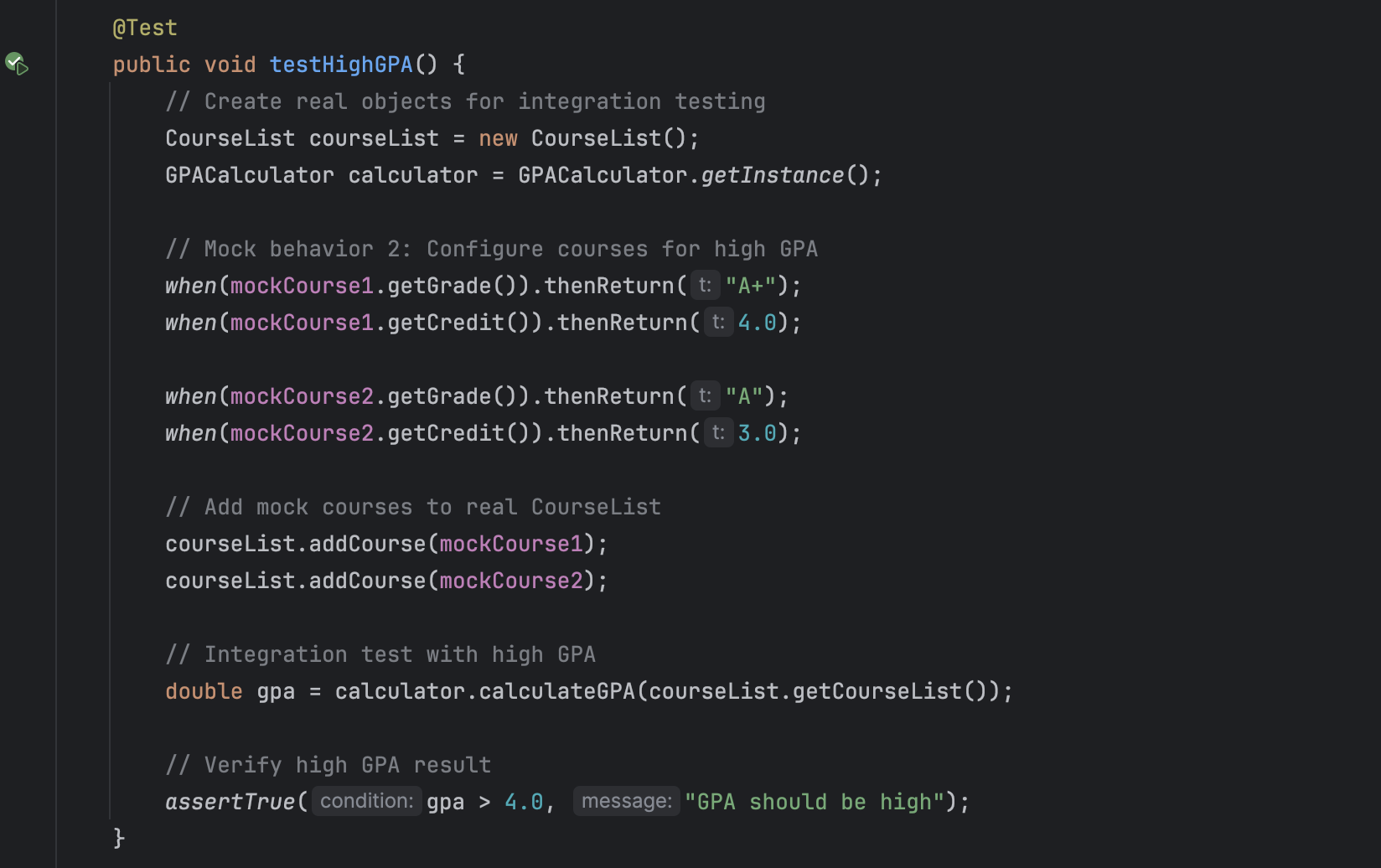


## c. **CourseList & GPACalculator Integration**

Here, a real CourseList object is populated with mocked Course objects, and GPA is calculated using GPACalculator. Scenarios tested:

* GPA for mixed, high, and low grade combinations
* Handling of an empty course list leading to division-by-zero exception

All tests use assertEquals, assertSame, assertTrue, or assertThrows for validation.



# 4.Conclusion

The integration tests effectively verify the correct interaction between the Course, CourseList, and GPACalculator classes. By using real and mocked components, the tests simulate real-world usage while maintaining isolation for focused testing.

The system behaves as expected in normal and edge cases, confirming the robustness of the current design. Exception scenarios are appropriately handled, making the application reliable under abnormal inputs.

Future work may include extending tests to UI components or database interactions if the system is expanded beyond its current in-memory structure.