

Previous WIS testing knowledge



Acme AirNav Solutions, Inc

Group Number: C1.066

Repository: <https://github.com/mquirosq/DP2-C1.066>

Members:

María Quirós Quiroga, marquiqui@alum.us.es

Guillermo Rodríguez Narbona, guirodnar@alum.us.es

Ignacio Mora Pérez, ignmorper1@alum.us.es

Daniel Herrera Urbano, danherurb@alum.us.es

Alejandro Parody Quirós, aleparqui@alum.us.es

February 20, 2025

Contents

Executive Summary	2
Revision History	3
1 Introduction	4
2 Architecture and Integration of Information Systems	4
3 Design and Testing 1	4
4 Conclusions	5

Executive Summary

The main objective of this report is to present the knowledge about Web Information System testing that we have previously acquired before this subject, thus providing a clear insight on the concepts and content we are already familiar with.

Revision History

Revision	Date	Description
1.0	2025-02-18	Initial draft
1.1	2025-02-20	Minor corrections

1. Introduction

The act of testing code as part of the development of a Web Information System is not something new, as it has been seen during this and the past course. As all members of the team are students from the Software Engineering grade at the University of Seville, we all share the same knowledge about testing a WIS from the previous subjects. This is the reason why, in the following sections, we are going to talk as a group.

2. Architecture and Integration of Information Systems

This subject was the first time we applied some type of testing that was used to check that the behaviour of our code was as expected. This content was seen during a couple of laboratory classes and as part of the second group project.

During the practices, we only saw how to create unit tests. These tests involve writing and running automated tests to verify the correct operation of individual units of code, such as functions, methods, or classes.

The second practical project was to develop a Restful API that provided information about videos collected from Youtube and Vimeo. For this, we followed a microservices architecture and used the Spring-Boot framework to build the three required systems. The concrete functions that made the collection and retrieval of the videos possible were the SUT in this project.

3. Design and Testing 1

This subject was part of the first half of our third academic year, which means that the knowledge is still very recent in our minds. In this subject, we have learned to perform unit testing, mock testing, etc.

First, for the unit tests, we implemented tests for the service classes, domain model classes, validators, and custom queries interface tests. For these, we tried implementing two positive cases, in which we checked that it followed a normal behaviour, and one negative case, where we tested abnormal behaviours. Though in some cases, as it was with the while or for loops, these required more cases, to test all the possible situations that could occur. It was done using the JUnit5 testing framework and the DataJpaTest annotations.

We also studied the application of sociable tests versus solitary tests and test doubles, using the Mockito framework for the latter.

For web layer testing, we studied and applied the Spring WebMvcTest annotations to verify outputs, such as HTTP status or JSON responses, simulating the corresponding

Http requests needed for the tests.

Next, for end-to-end and acceptance tests, we applied a Mock model view controller to simulate HTTP calls for the whole app. This allowed us to test several elements of the project all the way, for example, the registration form.

Lastly, for several of the React components we implemented, we learned and created, using Jest, some React component tests. In this way, we check that they were in the correct states and that they rendered correctly.

4. Conclusions

We, as software engineering students, know the importance of checking that the code we write works as it should, even though it was not after these two subjects that we learned several ways to do it.

Throughout these subjects, though mostly in the last one, we have acquired the knowledge and tools to understand how to perform tests, be it Unit testing, End-to-End, or front-end tests. With both a theoretical and practical foundation obtained from these two subjects, we hope that they will allow us to approach the related activities of the project proposed on this subject.

References

Intentionally blank.