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 19, 2025

Contents

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	2000-02-18	Initial draft

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, Vimeo, and another additional service if we wanted to add another. For this, we followed a microservices architecture and used the Spring-Boot framework to build the three systems. The concrete functions that made the collection and retrieval of the videos possible were the subject of the tests we had to perform.

3. Design and Testing 1

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

mock testing, etc.

First, as for the unit tests, we implemented in the corresponding project of the subject, tests for the service classes, domain model classes, validators, and custom queries interface tests. For these, we implemented two positive cases, where we checked that it followed a normal behaviour, and one negative case, where we tested abnormal behaviours. It was done using the JUnit5 testing framework and the DataJpaTest annotation.

We also studied the application of sociable tests versus solitary tests and test doubles. In this last section, we saw the use of Mockito framework.

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, applying a Mock model view controller to simulate the Http calls on for the whole app. This allowed us to test several elements of the project, for example, the registering form.

Lastly, for several of the react components we as a group implemented, we learned and created, with the use of Jest, some react component tests, checking that they were in the correct states and that the backend invocations were performed correctly.

4. Conclusions

In general, during these subjects, though mostly on the last one, we have acquired the knowledge and tools to approach the related activities of the project proposed in this subject.

References

Intentionally blank.