Workgroup C1.067

WIS TESTING REPORT

20/02/2025

Aarón Jesús Acuña Bellido - [aaracubel@alum.us.es](mailto:aaracubel@alum.us.es)

Javier Arellano López – [javarelop@alum.us.es](mailto:javarelop@alum.us.es)

Raúl Calero Capote – [raucalcap@alum.us.es](mailto:raucalcap@alum.us.es)

Tudor Cristian Lacatus Cosma [-ionlac@alum.us.es](mailto:-ionlac@alum.us.es)

Miguel Prado Jiménez – migprajim@alum.us.es

Table of Contents

[INTRODUCTION 4](#_Toc190950749)

[CONTENTS 4](#_Toc190950750)

[CONCLUSIONS 5](#_Toc190950751)

[BIBLIOGRAPHY 5](#_Toc190950752)

Executive Summary:

During our coursework, we explored various types of testing, focusing on their theoretical foundations and practical applications. We emphasized enhancing test coverage, particularly through unit testing with frameworks like JUnit and Mockito, and integration testing to ensure seamless communication between components. We also reviewed frontend testing methodologies and utilized Jest for testing React components. Throughout the process, we identified structural issues and eliminated redundant code. Our collaborative approach to testing helped us ensure thorough system validation and address potential issues effectively.

Revision Table

|  |  |  |
| --- | --- | --- |
| Revision Number | Date | Description |
| 001 | 18/02/2025 | First version of the document |
| 002 | 20/02/2025 | Revised document to improve quality and consistency |
|  |  |  |

# INTRODUCTION

Our learnings from DP1 (DISEÑO Y PRUEBAS I) have equipped us with a thorough understanding of several critical testing concepts and best practices. These insights have continuously informed our approach to testing throughout our projects.

# CONTENTS

During our coursework, we explored various types of testing, gaining a solid understanding of both their theoretical foundations and practical applications. Our goal was to increase test coverage, with particular attention to code coverage and branch coverage, to maximize the overall quality of our project.

* **Unit Testing.** We learned how to design unit tests using frameworks such as JUnit and Mockito. In our project, we applied these concepts to test key entities, services, and controllers, ensuring that core functionalities were robust and reliable.

**Integration Testing.** We studied strategies to verify correct communication between controllers, services, and databases. Although integration testing was not the primary focus in previous courses, we gained exposure to its concepts and later applied them in our project by testing interactions between backend components and data persistence layers.

* **Frontend Testing** We reviewed frontend testing methodologies but did not implement them extensively. However, we understood the importance of validating UI behaviors, user interactions, and component rendering. In our project, we applied this knowledge by using Jest to test React components, ensuring that game elements and user information are displayed correctly.

Structural modeling issues emerged during our testing phase, such as an excess of squares on our game board compared to the intended design. Additionally, we identified redundant code, including unused endpoints and unreachable exceptions, which highlighted areas that required optimization.

We gained an understanding of end-to-end (E2E) and acceptance tests, even though we did not implement them during the project, as our coursework provided us with valuable knowledge about these methodologies. Moreover, we are also familiar with informal testing practices.

Testing was a collective responsibility within our team. Each member was tasked with testing specific entities and reviewing the tests created by others. This collaborative approach not only ensured a thorough validation of the system but also helped pinpoint potential issues that might have been overlooked.

# CONCLUSIONS

This coursework has equipped us with essential testing methodologies and fostered a collaborative mindset. By sharing testing responsibilities, we improved our project's quality and developed skills that will be invaluable in future endeavors in software testing and quality assurance.

# BIBLIOGRAPHY

Intentionally blank.