University of Sevilla

Higher Technical School of Computer Engineering

D01 – WIS Testing Report

Degree in Computer Engineering - Software Engineering

Desing and Testing II

Course 2024 – 2025

Date	Version
18/02/2025	v1.0

Practice Group: C1.040	
Members	**Email**
Isabel Sánchez Castro	isasancas@alum.us.es
Javier Aponte Pozón	javapopoz@alum.us.es
José María Portela Huerta	josporhue@alum.us.es
Paula María Suárez Linares	pausualin@alum.us.es
Julia Virginia Ángeles Burgos	julangbur@alum.us.es

Repository: https://github.com/DP2-2025-C1-040/Acme-ANS

Table of contents

Revision Table

Date	Version	Description	Delivery
18/02/2025	V1.0	First Delivery analysis	D01

Introduction

In this document, we will share our previous knowledge about testing in a Web Information System (WIS), within the context of the course Design and Testing II. We will discuss the different types of tests we are familiar with, focusing especially on unit tests. We will cover the classes that are usually tested in this context. It is essential to note that this information is familiar to the group before starting this course, providing us with a solid foundation for understanding and applying the concepts discussed in this course. Our goal is to provide an overview of the key concepts and best practices we have learned in previous projects, thus contributing to our learning process and promoting discussion within the group and with the faculty.

Contents

Some of the types of tests known to the pre-subject group are:

- **Unit tests:**

  These are those tests that focus on validating exclusively a functionality of the system. They follow an 'Arrange/Fixture', 'Act', 'Assert' structure. The tests can be performed manually or automated. They can be either affirmative, verifying that the functionality is executed correctly when applicable, or negative, confirming that exceptions are generated when applicable.

  Within a Model-View-Controller (MVC) architecture, we understand that these tests are usually performed to validate the operation of the methods in the “Service” and “Controller” classes. In “Service” tests, the operation of the business logic is verified, while in “Controller” tests, the behavior of HTTP requests from the client to the server is evaluated. For “Controller” tests, in order to avoid system dependency, test doubles using 'Stub', 'Mock' or 'Fake' can be employed. 'Stub' returns a default value, 'Mock' verifies that the corresponding method is called without considering the value, and 'Fake' creates a simplified replica of the method to avoid resorting to the real one.

- **Integration testing:**

  These are tests that verify that the different modules and/or services used by the system work in harmony when working together.

- **Acceptance testing:**

  These tests are designed to validate whether a system meets business requirements. They are executed with the software running and focus on emulating user actions to ensure the achievement of objectives.

- **Exploratory testing:**

  Exploratory testing is a methodology in which you learn about the application, design test cases and execute them simultaneously. Each testing session has a defined objective and a time limit for its completion.

- **End-to-End testing:**

  These are tests that replicate the behavior of users with the system. These tests verify that the flows followed by a user give the expected results.

Conclusions

Testing is essential because it allows us to verify the correct functioning of the system we have developed before its full completion. This avoids having to correct errors that could

become more costly as the system grows. In addition, they help us to more easily detect problems within the code. It is also essential to know how to select the right type of test depending on what you want to validate.

Bibliography

Theory slides given in the course Design and Testing II.