



WIS TESTING REPORT

Noviembre D01 – Diseño y Pruebas II

25/10/2022

Grupo D03

MARTÍNEZ SUÁREZ, DANIEL JESÚS

danmarsua1@alum.us.es

<https://github.com/danmarsua1/Acme-Courses>

ÍNDICE

RESUMEN	1
HISTORIAL DE VERSIONES	2
INTRODUCCIÓN	3
CONTENIDO	4
CONCLUSIONES	5
BIBLIOGRAFIA	6

RESUMEN

En el grado universitario de Ingeniería Informática del Software, se han adquirido algunas nociones sobre el *testing* de los sistemas de Software, sobre todo en la asignatura de Diseño y Pruebas I. Este documento pretende plasmar los conocimientos que se tienen antes de empezar con el proyecto.

HISTORIAL DE VERSIONES

FECHA	VERSIÓN	DESCRIPCIÓN DE CAMBIOS
21/10/22	V1.0	Redacción del resumen, la introducción y el contenido del reporte
24/10/22	V2.0	Redacción de las conclusiones
25/10/22	V2.1	Revisión del informe

INTRODUCCIÓN

A pesar de que el *testing* forma una parte importante de un Sistema de Información Web para poder detectar errores en el sistema antes de ponerlo en producción y garantizar así la calidad del proyecto, el conocimiento que se tiene por el momento de ello es limitado. De hecho, tan solo se tiene experiencia en *tests* unitarios, de servicio, de controladores y funcionales.

En este reporte se describirán dichos conocimientos, adquiridos en anteriores asignaturas del grado y también gracias a la experiencia laboral.

CONTENIDO

Tanto en el grado universitario como en las experiencias laborales que he tenido, he podido tener un pequeño acercamiento al *testing*.

A nivel universitario, la introducción al *testing* se ha hecho principalmente a través de la asignatura Diseño y Pruebas I. En ella se comprendió la necesidad de hacer testeos en el software, ya que eso permite, por un lado, detectar errores en etapas más tempranas del proyecto, reduciendo así el impacto que podrían suponer a medida que éste avanzara, y, por otro lado, asegurar la calidad del proyecto. Los *tests* que se usaron en esta asignatura fueron *tests* unitarios, de servicio y de controladores. A nivel laboral, se ha trabajado usando pruebas funcionales.

A continuación, se definen brevemente y según la experiencia personal, cada uno de estos *tests*.

- Por un lado, los *tests* unitarios son las pruebas donde se testean componentes individuales de un programa.
- Por otro lado, los *tests* de controladores se llevaron a cabo empleando la herramienta Mockito, que es un motor de *testing* automático orientado a entornos en los que se necesitan llamadas a endpoints, además de facilitar el testing de controladores.
- En el caso de los *tests* de servicios, éstos se realizan para cada método dentro del servicio y funcionan con el principio de la triple A - *Arrange* (Preparar), *Act* (Actuar), *Assert* (Afirmar) -. En la primera parte se definen los términos a tratar; en la segunda se llevan a cabo operaciones sobre ese método que se va a testear; y en la última se comprueba que estas acciones se han llevado a cabo satisfactoriamente.
- Por último, el *testing* funcional consiste en comprobar que las funcionalidades que se requerían en el desarrollo se hicieron siguiendo las especificaciones y los requisitos del cliente, y satisfaciendo sus necesidades iniciales.

CONCLUSIONES

El conocimiento que se tiene antes de empezar este proyecto sobre el *testing* no es muy profundo. Aun así, voy a sacar el máximo provecho de lo que ya sé sobre ellos y la experiencia de pasados proyectos en los que he podido usarlos, tanto en la universidad como laboralmente.

Considero que con ello y lo que voy a aprender en esta asignatura, voy a adquirir un buen conocimiento que me va a servir tanto para este proyecto como en mi futuro.

BIBLIOGRAFIA

“Intentionally blank”