



Diseño y Pruebas 2
Testing a WIS

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática

D01 - Introduction

Testing a WIS




Grado en Ingeniería Informática – Ingeniería del Software

Diseño y Pruebas 2

Curso 2021 – 2022

Fecha	Versión
25/02/2022	v1

Grupo de prácticas	G6-04	Correo electrónico
Autores	Rol	
Álvarez Campanón, Juan José	Developer	juaalvcam@alum.us.es
Cordero Díaz, Jesús Javier	Developer	jescordia@alum.us.es
Paz Rivera, Roberto	Developer	robpozriv@alum.us.es
Romalde Dorado, Miguel Ángel	Developer	romalde3@gmail.com
Sánchez Mendoza, Nicolás	Developer	nicsanmen@alum.us.es
Suárez David, Rubén	Manager	rubsuadav@alum.us.es


	Diseño y Pruebas 2 Testing a WIS

Enlace al repositorio: <https://github.com/rubsuadav/Acme-Toolkits.git>

Enlace a las tareas asignadas: <https://github.com/users/rubsuadav/projects/1>


Control de Versiones

Fecha	Versión	Descripción
25/02/2022	v1	Creación del documento
28/02/2022	v2	Exponemos nuestras ideas sobre el testeo en un sistema de información web (WIS)
01/03/2022	v3	Revisión del documento

	<p>Diseño y Pruebas 2 Testing a WIS</p>
---	---

Índice


1. Introducción a WIS y al testing	4
2. Tipos de testing	5

	<div>Diseño y Pruebas 2</div> <div>Testing a WIS</div>
---	--

1. INTRODUCCIÓN A WIS Y AL TESTING

Podemos definir un sistema de información web (WIS en inglés) como una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

Por otra parte, el testing consiste en probar las implementaciones que se hacen en un software o en este caso en un WIS (sistema de información web) con el fin de verificar y validar su correcto funcionamiento. Además, se busca que finalmente consiga satisfacer las necesidades y requisitos del cliente para así entregar un producto de calidad.

	<div>Diseño y Pruebas 2</div> <div>Testing a WIS</div>
---	--

2. TIPOS DE TESTING

Primero hay que mencionar que no todos los proyectos o funcionalidades se deben de testear de la misma manera, hay diferentes tipos y formas de testear, además de muchos tipos diferentes de pruebas, con lo cual se debe de analizar qué pruebas son las más indicadas para cada proyecto.


Los tipos de pruebas las podemos clasificar en:

- **Pruebas funcionales:** Son pruebas que se encargan de testear la funcionalidad de cada implementación o de un conjunto de implementaciones y su interacción.
- **Pruebas no funcionales** (o de rendimiento): Son pruebas que tienen en cuenta parámetros como la confiabilidad, usabilidad y rendimiento.

También podríamos mencionar un **tercer tipo de prueba que son las de mantenimiento.**

Dentro de este tipo de pruebas debemos mencionar que pueden ser de dos tipos:

- Pruebas manuales: Son realizadas por las personas, que interactúan con el software y utilizan herramientas apropiadas para ello. Son pruebas que requieren de un profesional encargado de realizarlas.
- Pruebas automatizadas: Al contrario de las manuales, éstas están realizadas por máquinas, que ejecutan un test script que ha sido escrito previamente para esta prueba. Suelen ser más rápidas y confiables.

	<div>Diseño y Pruebas 2</div> <div>Testing a WIS</div>
---	--

Por último, mencionaremos algunos ejemplos de estos tipos de testing que son más relevantes.

- **Pruebas unitarias (unit tests):** Son, como su nombre indica, pequeñas pruebas que se realizan para ejecutar una pequeña parte del código o implementación y se testea que funcione bien por ella misma, sin tener mucho en cuenta su interacción con el resto de componentes del software.
- **Pruebas de integración (integration tests):** Al contrario que las pruebas unitarias, las pruebas de integración prueban que los diferentes módulos y servicios se coordinen y funcionen correctamente entre sí.
- **Pruebas funcionales (functional tests):** Este tipo de pruebas se centran más en el resultado final, es decir, su objetivo es que el resultado obtenido de una acción sea el esperado.
- **Pruebas End-to-End (E2E tests):** Son pruebas que replican el comportamiento de los usuarios que van a utilizar la aplicación o software. Tienen un punto muy a favor y es que son muy versátiles, pudiendo testear tanto parte de frontend como backend, además de que son muy dinámicas, pudiendo testear diferentes tipos de funcionalidades del código. Se pueden hacer manuales o automatizadas.