Informe sobre el Testing de WIS

Integrantes del grupo C1.026:

* Ignacio Blanquero Blanco ([ignblabla@alum.us.es](mailto:ignblabla@alum.us.es))
* Adrián Cabello Martín ([adrcabmar@alum.us.es](mailto:adrcabmar@alum.us.es))
* María de la Salud Carrera Talaverón ([marcartal1@alum.us.es](mailto:marcartal1@alum.us.es))
* Joaquín González Ganfornina ([joagongan@alum.us.es](mailto:joagongan@alum.us.es))
* Natalia Olmo Villegas ([natolmvil@alum.us.es](mailto:natolmvil@alum.us.es))

Fecha: Sevilla, 16 de Febrero 2024

Tabla de contenido

[Resumen del Informe 3](#_Toc159008623)

[Historial de Versiones 4](#_Toc159008624)

[Introducción 5](#_Toc159008625)

[WIS 6](#_Toc159008626)

[¿Qué sabíamos sobre WIS? 6](#_Toc159008627)

[¿Qué es WIS? 6](#_Toc159008628)

[Testing 7](#_Toc159008629)

[¿Qué sabíamos sobre el Testing? 7](#_Toc159008630)

[¿Qué es el Testing? 7](#_Toc159008631)

[Testing de WIS 8](#_Toc159008632)

[¿Qué sabíamos sobre el testing de WIS? 8](#_Toc159008633)

[¿Cómo es el testing de WIS? 8](#_Toc159008634)

[Conclusiones 9](#_Toc159008635)

[Bibliografía 10](#_Toc159008636)

# Resumen del Informe

En este informe, explicaremos en que se basan los tests de la arquitectura WIS y nuestros conocimientos previos a cursar la asignatura de DP2.

Para ello cada miembro del grupo ha expresado brevemente los conocimientos que tenía sobre este campo y lo hemos contrastado con la información que ahora tenemos a disposición.

Debería tener entre 50 y 250 palabras en la mayoría de los casos. It contains a short statement that addresses the problem or proposal detailed in the attached documents and features background information, a concise analysis and a conclusion. An executive summary is designed to help executives and investors decide whether to go forth with the proposal, making it critically important.

# Historial de Versiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Contenidos** | **Fecha** | **Contribuyente** |
| V0.1 | Informe sobre qué sabíamos sobre testing de WIS antes de esta asignatura. | 16/02/2024 | Joaquín González Ganfornina |
|  |  |  |  |
|  |  |  |  |

# Introducción

En el marco de la asignatura DP2, este informe se enfoca en evaluar el conocimiento previo de los integrantes de nuestro grupo de trabajo sobre el testing de la arquitectura WIS.

Primero, hemos puesto en común todos los conocimientos previos sobre la arquitectura WIS, sobre el testing en un producto software y finalmente los conocimientos del testing de WIS. En este informe se redactan nuestros conocimientos previos a cursar la asignatura y luego verificamos si nuestro conocimiento se acerca a lo que se define en las trasparencias de la asignatura.

En resumen, primero hemos definido los conceptos definidos previamente según nuestro y después hemos comprobado si estábamos en lo correcto o no.

# WIS

## ¿Qué sabíamos sobre WIS?

En rasgos generales sabíamos de su existencia y la habíamos puesto en practica previamente en otros contenidos como DP1, la cual todos los integrantes del grupo han cursado y aprobado previamente, por tanto, el formato que encontramos en el proyecto no nos pilla por sorpresa, aun así, en términos técnicos lo visto en DP2 nos ha aportado mucho ya que aun habiendo utilizado esta arquitectura no sabíamos mucho de esta debido a que la teníamos más interiorizada que aprendida de forma teórica.

En este proyecto concretamente nos podemos dar cuenta de su uso en varios aspectos, como las 3 diferentes capas ya definidas previamente, como la aparición de controladores, servicios y repositorios.

## ¿Qué es WIS?

El tipo de arquitectura WIS es usada en esta asignatura con el objetivo de aprender sobre las capas en las que se dividen los componentes de un sistema y cómo interactúan entre sí.

Normalmente estas capas son una capa de presentación, que muestra la interfaz al usuario, una capa de servicio que recibe las peticiones de la capa de presentación y devuelve los resultados y una base de datos que almacena los datos de la aplicación.

# Testing

## ¿Qué sabíamos sobre el Testing?

Como todos los integrantes del grupo aprobamos la asignatura de DP1, estábamos bastante familiarizados con el concepto del testing, debido a que en dicha asignatura tuvimos que aplicar tests para el proyecto de la misma, además de estudiarlos de forma teórica.

El testing en un proyecto software es un proceso mediante el cuál se verifica si las funciones implementadas en el proyecto funcionan correctamente, con el objetivo de encontrar errores para poder solucionarlos lo antes posible.

Una de las formas más efectivas de realizar tests es mediante pruebas unitarias, debido a que son fáciles y rápidas de ejecutar, aunque se pueden realizar distintos tipos de pruebas.

## ¿Qué es el Testing?

El testing en un proyecto de software se refiere al proceso de evaluación y verificación del sistema para asegurar que cumple con los requisitos especificados y funciona correctamente. Es El objetivo principal es identificar defectos, errores o problemas en el software antes de su implementación o lanzamiento.

El testing abarca diversas etapas y niveles en el desarrollo de software, como pruebas unitarias, pruebas de integración, pruebas de sistema y pruebas de aceptación. Cada una de estas etapas se enfoca en aspectos específicos del software y tiene como objetivo garantizar la calidad y confiabilidad del producto final.

El proceso de testing implica la creación de casos de prueba, la ejecución de esos casos, la identificación y documentación de los resultados, así como la repetición del proceso en diversas fases del desarrollo para asegurar la detección temprana de posibles problemas. Además, el testing busca validar que el software cumple con los requisitos funcionales y no funcionales establecidos, proporcionando así un producto final más robusto y confiable.

# Testing de WIS

## ¿Qué sabíamos sobre el testing de WIS?

Como dije previamente, todos los miembros del grupo aprobamos la asignatura de DP1, por lo que pensábamos que los tests de la arquitectura WIS serían bastante similares a los que vimos en esta asignatura.

## ¿Cómo es el testing de WIS?

En esta asignatura, aplicaremos el enfoque E2E (end to end). Este enfoque se basa en simular la experiencia completa del usuario. Para ello prueba el proceso completo o el sistema completo, desde el inicio hasta el final, esperando como resultado que funcione correctamente.

Además, se usa el plug-in de Eclipse Sonar Lint que aplica algunos tests para comprobar si existen bugs sin la necesidad de ejecutar el código.

# Conclusiones

En conclusión, los conceptos de testing y de la arquitecuta WIS están bastante interiorizados por parte de nuestro grupo debido a que la trabajamos recientemente en la asignatura de DP1. Sin embargo, no conocíamos el enfoque E2E (end to end) ni sabíamos que aplicaremos este enfoque para la arquitectura WIS.

# Bibliografía

En blanco a propósito.