**Testing a WIS**

| **Group:** C1.024 | |
| --- | --- |
| **Repository:** https://github.com/davvarmunUS/Acme-ANS-D01 | |
| Student #1  **ID:**  21015312M  **UVUS:**  davvarmun  **Name:**  Vargas Muñiz, David  **Roles:**  Project Manager, Developer | Student #2  **ID:**  77862000P  **UVUS:**  antlopcub  **Name:**  López Cubiles, Antonio José  **Roles:**  Developer |
| Student #3  **ID:**  17478227K  **UVUS:**  enrpermil  **Name:**  Pérez Milla, Enrique  **Roles:**  Developer | Student #4  **ID:**  47561989J  **UVUS:**  jesmarver  **Name:**  Martín Vergara, Jesús  **Roles:**  developer, analyst, operator |
| Student #5  **ID:**  Masked ID number  **UVUS:**  uvus  **Name:**  Surnames, Name  **Roles:**  role1, role2, role3 |  |
| **Date:** place month day, year | |

**1. Resumen Ejecutivo**

Antes de la asignatura, nuestro conocimiento sobre testing en WIS era limitado y se basaba más en la práctica que en un enfoque formal. Sabíamos la importancia de los tests unitarios y habíamos oído hablar de otros tipos de pruebas (integración, aceptación), pero no contábamos con una estrategia clara ni documentación adecuada. Conocíamos algunas herramientas y entendíamos la necesidad de automatizar, aunque no profundizamos en la integración continua ni en buenas prácticas de reporting. En términos generales, nos centramos en verificar la funcionalidad básica (páginas, formularios y consultas a la base de datos) sin un método de pruebas completamente estructurado ni un plan que garantizara la calidad del producto a largo plazo.

**2. Tabla de Versiones**

| Versión | Fecha | Descripción |
| --- | --- | --- |
| 1 | 18/02/2025 | Creación del informe |

**3. Introducción**

Antes de cursar la asignatura de testing en entornos de WIS (Web Information Systems), nuestro grupo contaba con una base muy práctica, pero con un enfoque poco estructurado en lo que a pruebas de software se refiere. La mayor parte de nuestras nociones surgía de la experiencia directa programando proyectos de menor escala o en lo visto en Diseño y Pruebas 1, sin llegar a interiorizar métodos y herramientas de testeo de forma rigurosa. Además, aunque estábamos familiarizados con la terminología fundamental (por ejemplo, la diferencia entre pruebas unitarias y de integración), carecíamos de una estrategia clara que abarcara tanto la planificación como la ejecución y el seguimiento de las pruebas. En pocas palabras, sabíamos que era necesario testear, pero no contábamos con el conocimiento para diseñar una serie de pruebas adecuadas a sistemas web de mayor complejidad.

**4. Contenidos**

**1.Conocimientos previos generales de testing**

* Entendimiento básico de pruebas unitarias, de integración y de aceptación.
* Uso esporádico de tests para verificar funcionalidad, sin un enfoque sistemático.

**2.Estrategia de pruebas y metodología**

* Falta de un plan formal de testing (qué probar, cuándo y cómo).
* Limitaciones en la definición de criterios de aceptación y cobertura de casos de prueba.

**3.Herramientas y entornos de test**

* Conocimiento superficial de frameworks (JUnit, por ejemplo) y de entornos de pruebas.
* Poca experiencia en configuración y automatización de entornos de test en proyectos WIS.

**4.Objetivos de mejora**

* Adoptar metodologías formales de testing (planificación, diseño, ejecución y seguimiento de pruebas).
* Integrar el ciclo de pruebas en la dinámica de desarrollo.
* Asegurar la calidad del producto a largo plazo y no solo “que funcione” en el momento.

**5. Conclusión**

En resumen, nuestro grupo partía de una base de conocimientos de testing muy orientada a la práctica y al ensayo y error, con un entendimiento limitado de las metodologías y herramientas que un proyecto WIS realmente requiere. Aunque contábamos con algo de experiencia en pruebas unitarias y conocíamos de forma superficial otros tipos de test, no disponíamos de un marco sólido que incluyera planificación, documentación y automatización. De cara a proyectos de mayor envergadura, se hacía evidente la necesidad de profundizar en estrategias de pruebas, integración continua y reporting. Por ello, la formación que ofrece la asignatura representa una oportunidad para asentar las bases de un testing más disciplinado y eficaz, que garantice la calidad y la fiabilidad de las aplicaciones web a lo largo de todo su ciclo de vida.

**6. Bibliografía**

Intencionalmente en blaco.