



4. Pruebas e Implementación.

4.1 Diseño de caso de prueba.

4.2 Pruebas de componentes.

4.3 Pruebas del sistema.

4.4 Documentación de resultados de las pruebas.

4.5 Entrega del sistema y Capacitación a usuarios.

4.6 Entrega de documentación técnica y de usuario del sistema.

Pruebas del sistema

Las pruebas del sistema son un proceso esencial dentro del ciclo de vida del sistema. Estas pruebas se realizan después de las pruebas de unidad y las pruebas de integración, y tienen como objetivo evaluar el sistema completo en lugar de sus componentes individuales, teniendo como objetivo verificar que el sistema en su conjunto cumple con los requisitos especificados y funciona de acuerdo a las expectativas del requerimiento.

Permite identificar defectos o eventualidades que puedan haber pasado desapercibidos en etapas anteriores del proyecto. Esto incluye eventualidades de funcionalidad, rendimiento, seguridad, y cualquier otro aspecto crítico para la operación del sistema.

Al descubrir y corregir eventualidades, antes de la implementación, se reducen significativamente los costos y riesgos asociados con la corrección en un ambiente de producción.



Importancia

- **Garantizar la conformidad:** esenciales para asegurarse de que el sistema cumple con los requisitos y especificaciones.
- **Evaluar el rendimiento:** evaluar el comportamiento del sistema en condiciones de uso real o simuladas. Esto incluye la capacidad de respuesta, la escalabilidad y la capacidad de gestionar cargas de trabajo significativas.
- **Detectar eventualidades de integración:** validar la interoperabilidad entre los diferentes componentes dentro del sistema, relevante en sistemas complejos que involucran múltiples módulos y tecnologías.
- **Validar la seguridad:** identificar vulnerabilidades y debilidades en la integridad, disponibilidad, confidencialidad del sistema.
- **Verificar la usabilidad:** simulación de interacción de manera efectiva y satisfactoria, con base en mejores prácticas de UX/UI.

Nota:

Para llevar a cabo pruebas del sistema efectivas, es necesario seguir un enfoque planificado y metódico. Esto implica la creación de casos de prueba que cubran todos los aspectos importantes del sistema, la ejecución de pruebas en un entorno similar al de producción y el registro de resultados para su posterior análisis. Además, es esencial involucrar a los participantes para garantizar que el sistema cumpla con sus expectativas.



Herramientas y frameworks para realizar pruebas

- **Selenium:** Selenium es una herramienta de automatización de pruebas que se utiliza para realizar pruebas de interfaz de usuario en aplicaciones web. Permite la creación de scripts para interactuar con elementos de la página web y verificar su funcionamiento en diferentes navegadores.
- **JUnit y TestNG:** Estos son frameworks de pruebas unitarias para Java. Se utilizan para escribir y ejecutar pruebas unitarias de forma estructurada y automatizada.
- **Cucumber:** Cucumber es una herramienta de automatización de pruebas que se utiliza en pruebas de aceptación y pruebas de comportamiento. Permite escribir pruebas en lenguaje natural y se utiliza comúnmente en entornos de desarrollo ágil.

Herramientas y frameworks para realizar pruebas

- **Jenkins:** Jenkins es una herramienta de integración continua que puede utilizarse para automatizar la ejecución de pruebas de sistema. Se integra con otros sistemas y herramientas de prueba, lo que facilita la automatización de pruebas en un entorno de desarrollo continuo.
- **Postman:** Postman es una herramienta popular para pruebas de API. Permite crear, enviar y validar solicitudes a servicios web, así como automatizar pruebas de API.
- **Appium:** Appium es una herramienta de automatización de pruebas para aplicaciones móviles. Permite realizar pruebas en aplicaciones móviles en plataformas como Android e iOS.



Herramientas y frameworks para realizar pruebas

- **TestComplete:** TestComplete es una herramienta de automatización de pruebas que se utiliza para pruebas de interfaz de usuario en aplicaciones de escritorio, web y móviles. Es conocida por su capacidad de automatización de pruebas en múltiples plataformas.
- **Robot Framework:** Es un framework de automatización de pruebas que permite escribir pruebas en lenguaje natural y es compatible con pruebas de interfaz de usuario, pruebas de API y pruebas de base de datos.
- **LoadRunner y JMeter:** Estas herramientas se utilizan para pruebas de rendimiento y pruebas de carga. Permiten simular un gran número de usuarios y medir el rendimiento y la escalabilidad de una aplicación.
- **Squish:** Squish es una herramienta de automatización de pruebas que se utiliza para pruebas de GUI en aplicaciones de escritorio, web y móviles. Ofrece soporte para varios lenguajes de programación.

Referencias.

- Abdurrahmanov, Marat. (2023). Developing a unit testing framework for control system sistema implementation of a degassing vacuum pumping station. *Izvestiya vysshikh uchebnykh zavedenii. Gornyi zhurnal*. 113-126. 10.21440/0536-1028-2023-3-113-126.
- Maqbool, Bilal & Rehman, Fateh & Abbas, Muhammad & Rehman, Saad. (2018). Implementation of sistema Testing Practices in Pakistan's sistema Industry. 10.1145/3180374.3181340.
- Agyei, Mensah & Ajoke, Ogunloju & Yaw, Mensah. (2023). Hybrid sistema testing model to improve sistema quality assurance. 104-116. 10.30574/gjeta.2023.17.1.0206.
- Yordanova, Zornitsa. (2019). Educational Innovations and Gamification for Fostering Training and Testing in sistema Implementation Projects. 10.1007/978-3-030-33742-1_23.
- Pargaonkar, Shravan. (2023). A Comprehensive Review of Performance Testing Methodologies and Best Practices: sistema Quality Engineering. *International Journal of Science and Research (IJSR)*. 12, 2008-2014. 10.21275/SR23822111402.
- Masalimov, T. & Sayfullina, S.. (2023). ANALYSIS OF THE ECONOMIC EFFICIENCY OF AUTOMATED TESTING IMPLEMENTATION IN sistema DEVELOPMENT. *Business Strategies*. 11, 68-76. 10.17747/2311-7184-2023-3-68-76.
- Khan, Sarah & Akter, Sanjida & Hasan, Shihab Z & Sharmin, Dilruba. (2023). Various sistema Testing Techniques, their Implementation and Significance on Banking Applications: In Context of Bangladesh. *International Journal for Research in Applied Science and Engineering Technology*. 11, 2072-2089. 10.22214/ijraset.2023.55066.
- Marchetto, Alessandro. (2023). A Rapid Review on Fuzz Security Testing for sistema Protocol Implementations. 10.1007/978-3-031-43240-8_1.