INVESTIGACION DE PRUEBAS

BTJL

2693508

PROYECTO

A.D.S.O

CEET

SENA

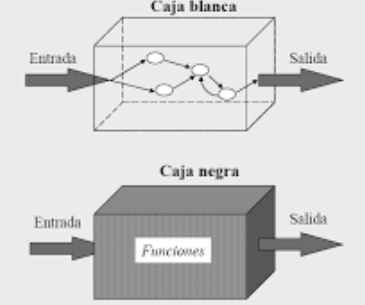
BOGOTA, COLOMBIA

FEBRERO DEL 2024

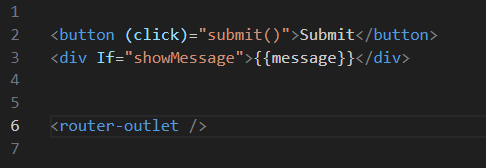
1. Pruebas caja negra

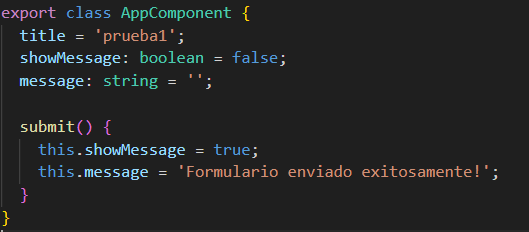
La prueba de caja negra, test funcional o prueba comportamental, es un tipo de prueba de software directa, cuya finalidad es analizar la compatibilidad entre las interfaces de cada uno de los componentes del software. No tiene en consideración la lógica interna del sistema.

Son una técnica de prueba importante que se enfoca en evaluar la funcionalidad del software desde una perspectiva externa, centrándose en las entradas y salidas sin necesidad de conocer su implementación interna, Detecta errores en la lógica de programación, la interfaz de usuario y la funcionalidad general pero Puede no revelar problemas de rendimiento o errores relacionados con la lógica interna del programa.



Ejemplos



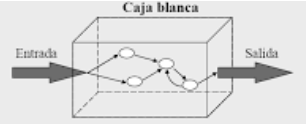


1. Pruebas caja blanca

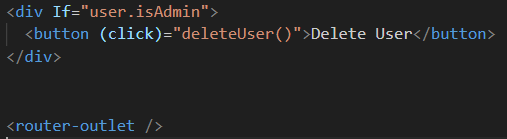
La prueba de caja blanca es una forma de prueba de aplicaciones que proporciona al evaluador un conocimiento completo de la aplicación que se está probando, incluido el acceso al código fuente y a los documentos de diseño.

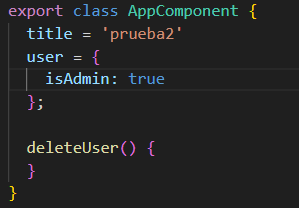
Es un Acceso y estudio del código para identificar caminos de ejecución consiste en:

1. Acceso al código fuente del programa.
2. Identificación de caminos de ejecución y estructuras lógicas.
3. Diseño de casos de prueba basados en la estructura interna del código.
4. Ejecución de pruebas para cada componente del código.
5. Evaluación de la cobertura del código para garantizar que se han probado todas las instrucciones, ramas y condiciones.



Ejemplos

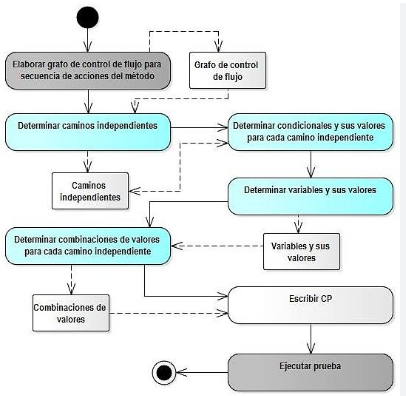




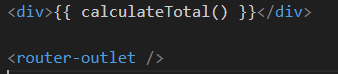
1. Pruebas unitarias

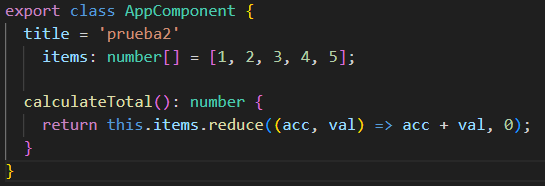
Una prueba unitaria es un bloque de código que verifica la precisión de un bloque más pequeño y aislado de código de aplicación, normalmente una función o un método. La prueba unitaria está diseñada para verificar que el bloque de código se ejecuta según lo esperado, de acuerdo con la lógica teórica del desarrollador. Hay muchos tipos diferentes de pruebas de software, cada una con objetivos y estrategias específicos.

Son una técnica fundamental en el desarrollo de software, ya que ayudan a garantizar la calidad y la integridad de las unidades de código individuales, lo que a su vez contribuye a la fiabilidad y Mantenibilidad del software en general.



Ejemplos

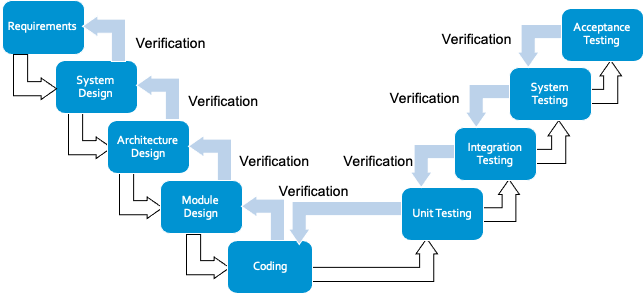




1. ¿Qué son las pruebas de verificación de software?

Se comprueba que el software se ha realizado correctamente

Son un conjunto de actividades y técnicas que se utilizan para evaluar si el software cumple con los requisitos especificados y si es apto para su uso previsto. Estas pruebas se realizan después del desarrollo del software, y antes de su lanzamiento al público.



Ejemplo

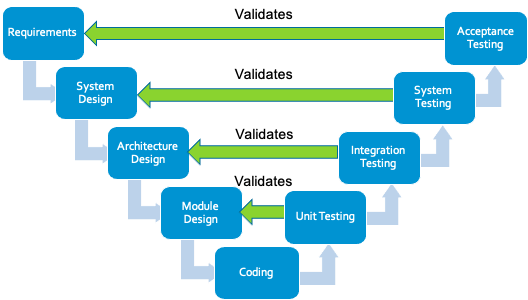
Un sistema de gestión de bibliotecas. Durante la fase de desarrollo, el equipo lleva a cabo pruebas de verificación para confirmar que la funcionalidad de búsqueda de libros funciona según lo esperado, devuelve resultados precisos y muestra la información relevante de manera correcta en la interfaz de usuario. Estas pruebas se centran en asegurarse de que el código implementado cumple con los requisitos funcionales y no funcionales especificados para la función de búsqueda.

1. ¿Qué son las pruebas de validación de software?

Se comprueba si el producto se ajusta a los requisitos del cliente.

Las pruebas de validación de software son un conjunto de actividades y técnicas que se utilizan para evaluar si el software cumple con los requisitos especificados y si es apto para su uso previsto. Estas pruebas se realizan después del desarrollo del software, y antes de su lanzamiento al público. Evalúan si el producto es fiable y seguro, y si se comporta de manera adecuada en las condiciones previstas.

Si el software no cumple con los requisitos, se deben corregir los errores y volver a realizar las pruebas de validación hasta que el software siga los estándares de calidad adecuados.



Ejemplo

Un sistema de gestión de bibliotecas. Una vez que se completa el desarrollo y se entrega el producto al cliente, se llevan a cabo pruebas de validación para confirmar que el sistema cumple con las expectativas del cliente. Esto puede incluir pruebas de usabilidad para asegurarse de que la interfaz de usuario sea intuitiva y fácil de usar, pruebas de rendimiento para verificar que el sistema puede manejar la carga esperada de usuarios y datos, y pruebas de aceptación por parte del cliente para confirmar que el sistema cumple con los requisitos comerciales y funcionales acordados.