

Pruebas funcionales y Pruebas no funcionales

Calidad de Software - CSY4111



CONTENIDO

01
ESPECIFICACIÓN DE
REQUISITOS DE SOFTWARE

02
PRUEBAS
FUNCIONALES

03
PRUEBAS NO
FUNCIONALES

Refrescando conocimiento

DuocUC[®]

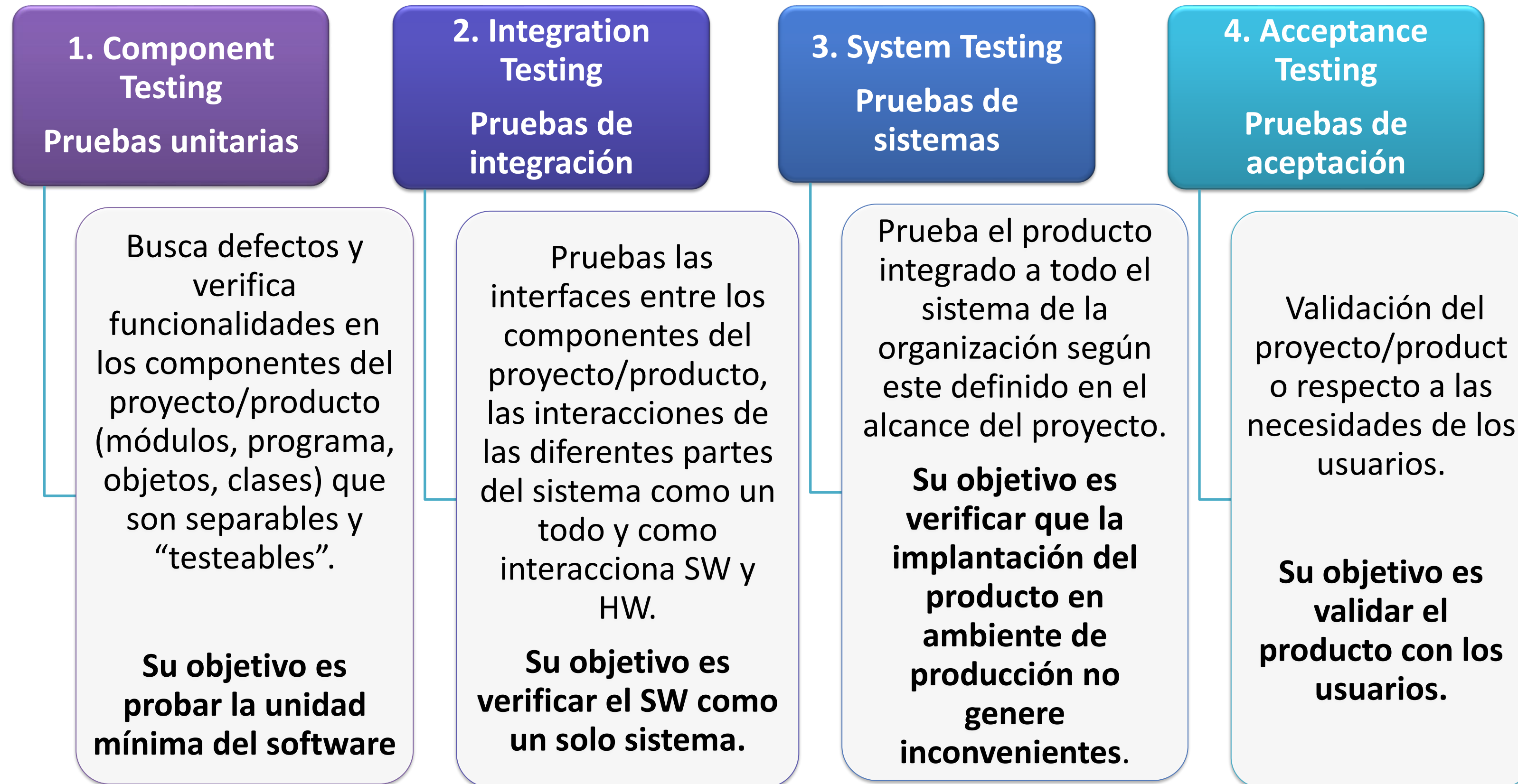


REFRESCANDO CONOCIMIENTOS

El ambiente ideal de las pruebas de Testing es aquel que es totalmente independiente del ambiente de desarrollo del software, de esta manera se logra objetividad en las pruebas.



REFRESCANDO CONOCIMIENTOS



ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

01



ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

La ERS es una descripción completa del comportamiento del sistema que se va a desarrollar. Esta incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software.



El objetivo principal de la Especificación de Requisitos del Sistema (ERS) es servir como medio de comunicación entre clientes, usuarios, ingenieros de requisitos y desarrolladores.

ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

Los requerimientos funcionales

Refieren a las acciones y funcionalidades específicas que debe cumplir un sistema o software.

Estos requerimientos describen qué debe hacer el sistema, cómo debe comportarse y qué resultados debe producir.

Por ejemplo, un requerimiento funcional para un sistema de reserva de vuelos podría ser "El sistema debe permitir a los usuarios buscar vuelos disponibles, seleccionar un vuelo y realizar la reserva".



ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

Los requerimientos NO funcionales

Se centran en los **atributos de calidad** del sistema, en lugar de las funcionalidades específicas.

Estos requerimientos describen características como el rendimiento, la usabilidad, la seguridad y la confiabilidad del sistema.



Por ejemplo, un requerimiento no funcional para el mismo sistema de reserva de vuelos podría ser "El sistema debe tener un tiempo de respuesta máximo de 2 segundos para mostrar los resultados de búsqueda".

02
PRUEBAS FUNCIONALES

DuocUC[®]

02

PRUEBAS FUNCIONALES

Duoc UC[®]

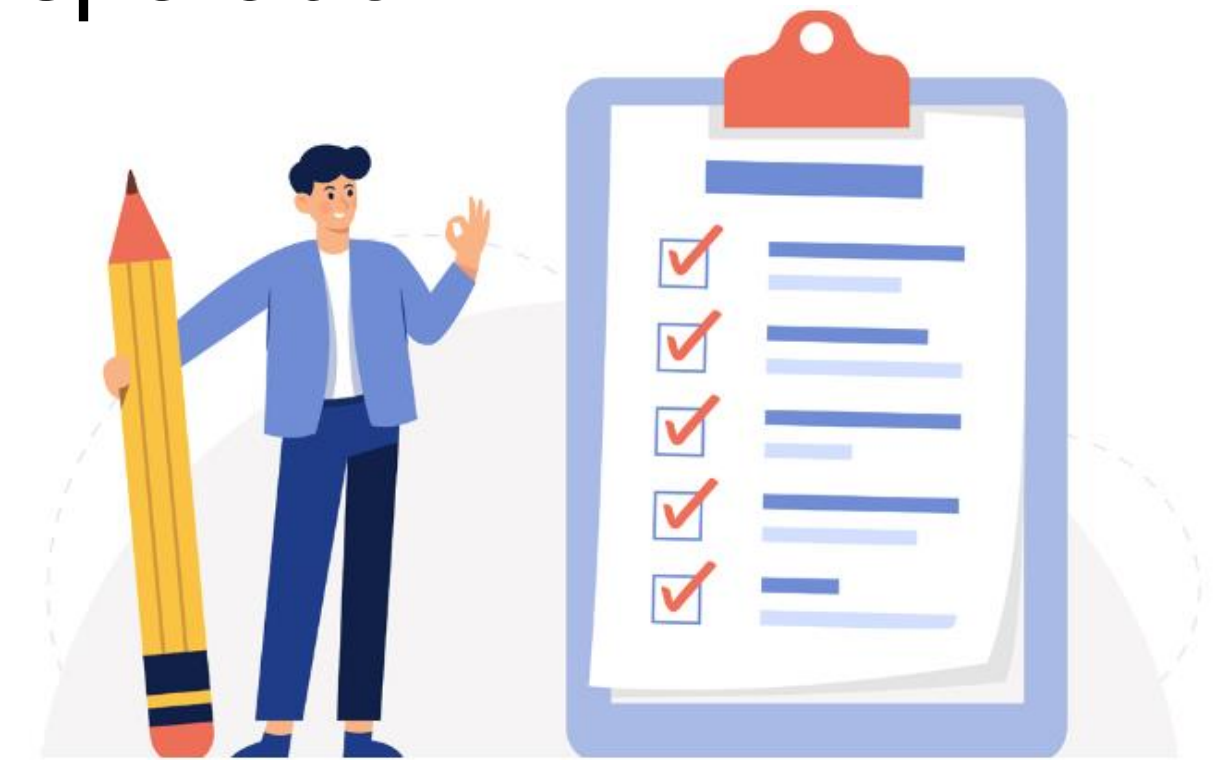


PRUEBAS FUNCIONALES

Las pruebas funcionales se centran en verificar que el software cumpla con los requerimientos funcionales establecidos.

Estas pruebas se enfocan en probar las acciones y funcionalidades específicas del sistema para asegurar que se comporten según lo esperado.

Por ejemplo, en el sistema de reserva de vuelos, una prueba funcional podría ser verificar que los usuarios puedan buscar vuelos, seleccionar uno y completar el proceso de reserva sin errores.



Se enfocan en garantizar que el software cumpla con los requerimientos y funcionalidades establecidas

PRUEBAS FUNCIONALES

Estas pruebas verifican que el sistema realice las acciones correctas, produzca los resultados esperados y se comporte de acuerdo con las especificaciones funcionales.

Su objetivo es asegurar que el software funcione correctamente desde el punto de vista de las acciones y tareas que debe realizar.



Se enfocan en garantizar que el software cumpla con los requerimientos y funcionalidades establecidas

PRUEBAS FUNCIONALES

Algunos aspectos clave:

Identificación de escenarios de prueba: Se deben identificar los escenarios de prueba relevantes que cubran todas las funcionalidades del software.

Diseño de casos de prueba: Se deben diseñar casos de prueba que describan las acciones a realizar y los resultados esperados.

Estos casos de prueba deben cubrir todas las condiciones y caminos posibles dentro de la funcionalidad bajo prueba.

Cada escenario debe estar diseñado para probar una funcionalidad específica

PRUEBAS FUNCIONALES

Algunos aspectos clave:

Ejecución de pruebas: Los casos de prueba diseñados deben ejecutarse en el software para comprobar si el sistema cumple con los resultados esperados. Durante la ejecución, se registran los resultados y se documentan los errores o fallos encontrados.

Verificación de resultados: Se comparan los resultados obtenidos durante la ejecución con los resultados esperados. Si hay alguna discrepancia, se considera un defecto y se documenta para su posterior corrección.

Se deben clasificar las incidencias o defectos encontrados durante la ejecución de las pruebas.


PRUEBAS FUNCIONALES

Algunos aspectos clave:

Cobertura de prueba: Es importante asegurarse de que todas las funcionalidades críticas del software estén cubiertas por las pruebas. Esto implica probar diferentes escenarios, combinaciones de datos y condiciones límite para garantizar una cobertura exhaustiva.

Regresión de pruebas: Después de realizar cambios en el software, es necesario ejecutar pruebas funcionales nuevamente para asegurarse de que los cambios no hayan introducido nuevos errores o afectado las funcionalidades existentes.

Las pruebas funcionales pueden ser manuales o automatizadas, dependiendo de la complejidad y la repetitividad de las pruebas.

A black and white photograph of two IT professionals in a server room. A man in a white shirt and tie is standing and working on a server rack, holding a handheld device. A woman is sitting at a desk, typing on a keyboard. The room is filled with server racks and cables. A large black diagonal bar is on the left side of the image.

03 PRUEBAS NO FUNCIONALES

DuocUC[®]

PRUEBAS NO FUNCIONALES

Las pruebas no funcionales se centran en verificar los atributos de calidad y características del sistema establecidos en los requerimientos no funcionales. Estas pruebas se enfocan en aspectos como el rendimiento, la usabilidad, la seguridad y la confiabilidad del sistema.

Por ejemplo, en el sistema de reserva de vuelos, una prueba no funcional podría ser medir el tiempo de respuesta del sistema al buscar vuelos y verificar si cumple con el requerimiento de tener un tiempo de respuesta máximo de 2 segundos.



La calidad cuesta tiempo y dinero, la mala calidad también

PRUEBAS NO FUNCIONALES

Se realizan para evaluar el rendimiento, la seguridad, la usabilidad y otros aspectos importantes del software. Algunos detalles relevantes sobre las pruebas no funcionales son:

Rendimiento: Evalúan la capacidad de respuesta, la velocidad, la escalabilidad y la eficiencia del software bajo diferentes condiciones de carga.

Simulan situaciones de alta demanda y medir el tiempo de respuesta del sistema.



PRUEBAS NO FUNCIONALES

Se realizan para evaluar el rendimiento, la seguridad, la usabilidad y otros aspectos importantes del software. Algunos detalles relevantes sobre las pruebas no funcionales son:



Seguridad: Evalúan la capacidad del software para proteger los datos y resistir intentos de acceso no autorizados o ataques maliciosos.

Se pueden realizar pruebas de penetración o pen testing, pruebas de vulnerabilidad para identificar posibles brechas de seguridad y asegurar la integridad del sistema.

PRUEBAS NO FUNCIONALES

Usabilidad: Estas pruebas se centran en evaluar la facilidad de uso y la experiencia del usuario al interactuar con el software. Se pueden utilizar técnicas como, evaluaciones de accesibilidad y estudios de interacción para identificar áreas de mejora y garantizar una experiencia positiva para los usuarios.

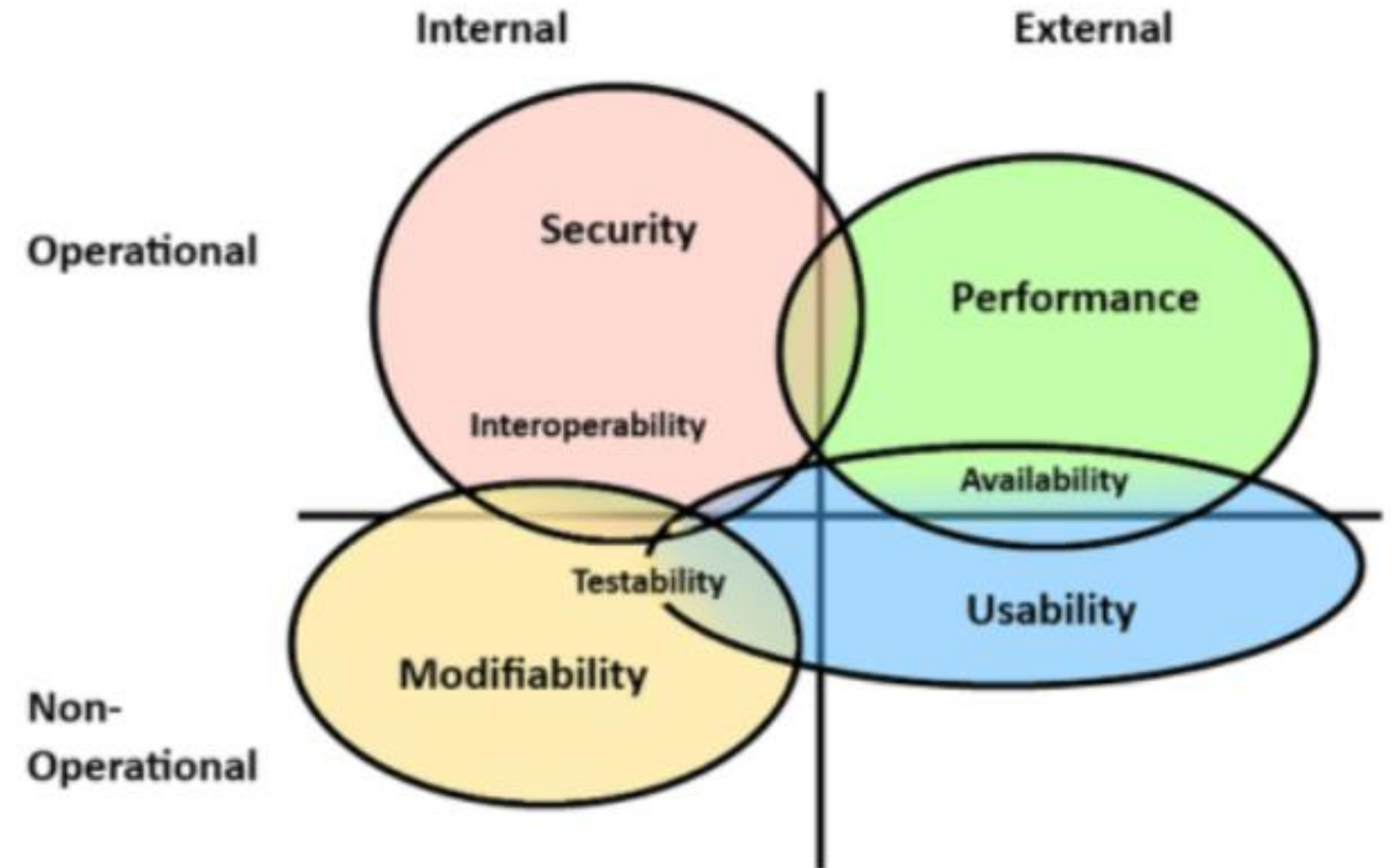
Fiabilidad: Las pruebas de fiabilidad se centran en evaluar la capacidad del software para funcionar de manera confiable y sin fallos durante un período prolongado de tiempo. Se pueden realizar pruebas de estabilidad, pruebas de recuperación ante fallos y pruebas de resistencia para evaluar la resistencia del sistema ante condiciones adversas.

La calidad cuesta tiempo y dinero, la mala calidad también

PRUEBAS NO FUNCIONALES

Mantenibilidad: Estas pruebas se enfocan en evaluar la facilidad con la que el software puede ser mantenido y modificado en el futuro.

Se pueden realizar pruebas de legibilidad del código, pruebas de cobertura del código y otras técnicas para evaluar la calidad del código y la facilidad de mantenimiento.



PRUEBAS NO FUNCIONALES

Estas pruebas ayudan a asegurar que el software sea seguro, eficiente, fácil de usar y confiable para los usuarios finales.

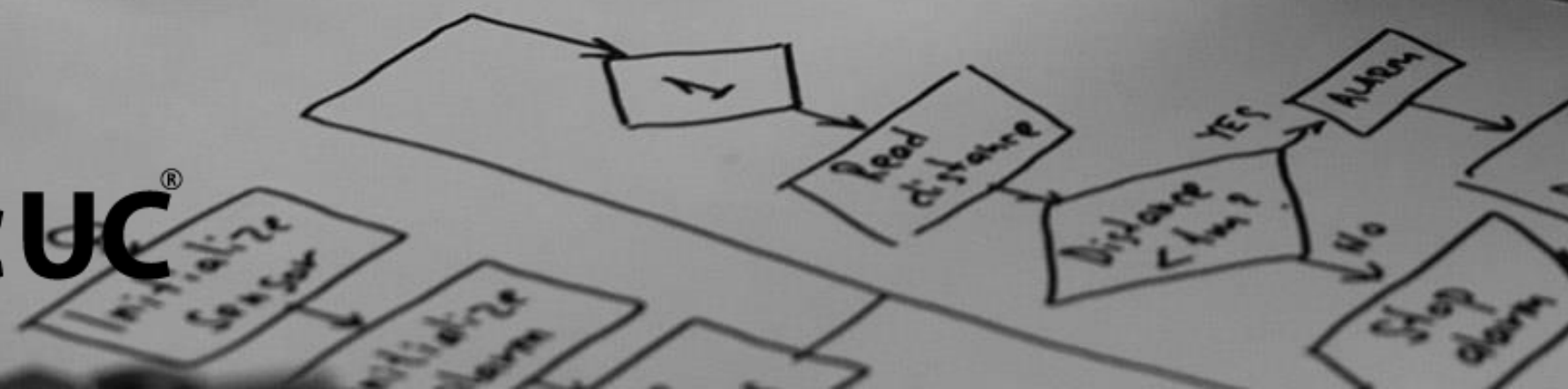


PRUEBAS NO FUNCIONALES

- Mejora la experiencia de usuario.
- Aumenta los ingresos en línea.
- Utiliza métricas importantes para poner a punto tu sistema.
- Determina los cuellos de botella como la configuración de base de datos.
- Ayuda a establecer si una nueva versión está preparada para la producción o no.
- Permite a los interesados conocer el rendimiento real en comparación con sus expectativas.

Conclusiones de la clase

DuocUC[®]



Conclusiones

- ✓ Las pruebas funcionales pueden ser manuales o automatizadas, dependiendo de la complejidad y la repetitividad de las pruebas.
- ✓ Las pruebas de software no funcionales son las que se hacen desde una perspectiva totalmente diferente a las pruebas de los procesos de negocio.

Bibliografía

- ✓ Carla Gomez. (12 de Septiembre 2020). Tipos de pruebas - Pruebas Funcionales. www.diariodeqa.com
Recuperado de <https://www.diariodeqa.com/post/tipos-de-pruebas-funcionales>

¿Qué entendemos por calidad y Calidad de software?

Calidad de Software - CSY4111