

Tipos de pruebas en proyectos de software

En el desarrollo de software, existen diferentes tipos de pruebas que se llevan a cabo para garantizar la calidad del producto final. Algunas de las más importantes son las pruebas de rendimiento, accesibilidad, funcionalidad y aplicación de buenas prácticas.



por YEINER DANIEL ANAYA DUARTE





Pruebas de rendimiento

1

Carga

Evaluar el comportamiento del sistema bajo diferentes niveles de carga y estrés.

2

Escalabilidad

Comprobar que el sistema puede manejar un aumento en el número de usuarios o transacciones.

3

Estabilidad

Asegurar que el sistema mantiene un rendimiento estable y confiable a lo largo del tiempo.



Objetivo de las pruebas de rendimiento

Identificar Cuellos de Botella

Detectar puntos débiles en el sistema y mejorar su desempeño.

Garantizar Escalabilidad

Comprobar que el sistema puede crecer sin perder calidad ni rendimiento.

Mejorar la Experiencia

Optimizar la velocidad y fluidez del software para brindar una mejor experiencia de usuario.

Reducir Riesgos

Identificar y mitigar problemas de rendimiento antes de que afecten a los usuarios.

Pruebas de accesibilidad

Usabilidad

Garantizar que el software sea fácil de usar para todo tipo de usuarios, incluyendo aquellos con discapacidades.

Compatibilidad

Asegurar que el sistema funcione correctamente en diferentes dispositivos, navegadores y tecnologías de asistencia.

Percepción

Comprobar que la información se presente de forma clara y accesible para usuarios con distintas capacidades.



Objetivo de las pruebas de accesibilidad

1

Inclusión

Garantizar que el software pueda ser utilizado por personas con diferentes tipos de discapacidad.

2

Cumplimiento

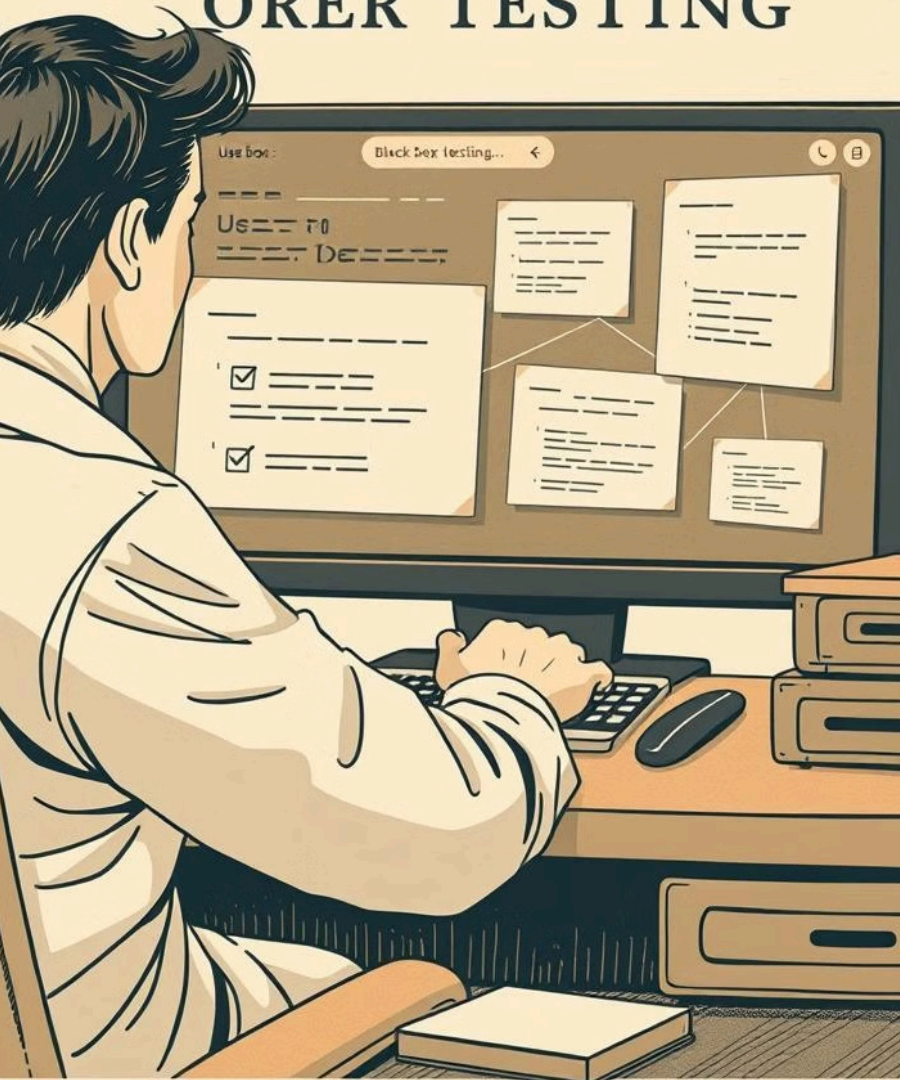
Asegurar que el producto cumpla con los estándares y normativas de accesibilidad aplicables.

3

Experiencia

Mejorar la experiencia de usuario para todos, sin importar sus capacidades o limitaciones.

PHCH-TIOX ORER TESTING



Pruebas de funcionalidad (caja negra)

AB

Entradas

Validar que las entradas del sistema se procesen correctamente.



Salidas

Comprobar que las salidas generadas sean las esperadas.



Flujos

Verificar que los flujos de trabajo y casos de uso funcionen sin problemas.



Objetivo de las pruebas de caja negra

1

Requisitos

Comprobar que el software cumple con los requisitos y funcionalidades especificados.

2

Usabilidad

Evaluar la facilidad de uso y la experiencia del usuario final.

3

Calidad

Asegurar que el software entregado cumpla con los estándares de calidad establecidos.

Pruebas de aplicación de buenas prácticas

Estructura del código

Legibilidad y mantenibilidad

Patrones de diseño

Implementación de soluciones probadas

Naming conventions

Consistencia y estandarización



Objetivo de las pruebas de buenas prácticas

Calidad de código

Garantizar que el código fuente cumpla con altos estándares de legibilidad y mantenibilidad.

Eficiencia

Mejorar el rendimiento y la escalabilidad del software mediante el uso de mejores prácticas.

Colaboración

Facilitar el trabajo en equipo y la evolución del proyecto a largo plazo.

Mejora continua

Mantener un proceso de desarrollo sostenible y adaptable a los cambios.

