
Resumen de los tipos de testing estudiados

Sanity Testing

- Su objetivo es determinar que los cambios o la funcionalidad propuesta funcionan como se esperaba, que errores reportados se hayan solucionado y que no se introduzcan más errores.
- Se realiza después de que el build haya superado el smoke testing y haya sido aceptado por el equipo de control de calidad para realizar más pruebas.
- Si la prueba de sanity falla, la compilación se rechaza
- El objetivo de las pruebas es verificar la "racionalidad" del sistema para proceder con pruebas más rigurosas.
- No lleva planeamiento, no lleva documentación y se utiliza cuando el tiempo es corto.

Integration Testing

- Probar la integración de sistemas y paquetes; probar interfaces con organizaciones externas. Pruebas realizadas para exponer defectos en las interfaces y en las interacciones entre componentes o sistemas integrados
 - El propósito de este nivel de prueba es exponer defectos en la interacción entre estos módulos de software cuando están integrados (módulo email con módulo contraseña por ejemplo).
 - Antes del testing de integración, existe el unit testing, donde cada uno de los módulos son testeados aislados de otros, y este es realizado normalmente por los desarrolladores.
-

User Interface Testing

- La calidad de la interfaz de usuario decide la primera (y última) impresión del usuario de un sitio web o aplicación. El diseño y la funcionalidad de la interfaz de usuario hacen o deshacen el software.
- UI Testing también se conoce como GUI, que significa graphical user interface.
- Este test incluye verificaciones funcionales, de diseño visual, de rendimiento, de usabilidad y de cumplimiento
- Se verifica tipografía, colores, paddings, borders, margins, gramática, diseño actual contra diseño propuesto, que los inputs muestran los outputs correctos, el layout, imágenes e iconos, legibilidad de textos...

Regression Testing

- Realizado para verificar que un cambio, actualización o mejora de código no ha afectado negativamente a las funciones existentes del producto.
 - Son responsables de la estabilidad general y la funcionalidad de todos los featured existentes.
 - Usualmente, una aplicación pasa por múltiples pruebas antes de que los cambios se integren en la rama de desarrollo principal. La prueba de regresión es el paso final, ya que verifica los comportamientos del producto como uno solo.
 - Existen 4 tipos: Volver a probar todo, selección de pruebas de regresión, priorización de casos de prueba, técnica híbrida.
-

UAT Testing

- Pruebas de aceptación del usuario , pruebas beta o de usuario final, es realizada por el usuario final o el cliente para verificar/aceptar el sistema de software antes de mover la aplicación de software al entorno de producción
- Pruebas de aceptación realizadas por futuros usuarios en un entorno operativo (simulado) centrándose en los requisitos y necesidades del usuario.
- El objetivo principal de esta prueba es validar el software frente a los requisitos comerciales.
- Esta es la prueba final realizada una vez que se completan las pruebas funcionales, de sistema y de regresión, es el último paso antes de que el producto vaya a producción.

Re-Testing o Confirmation Testing

- Con este tipo de test probamos que un bug haya sido arreglado, validamos si este realmente ha sido solucionado por los developers
 - Debe ser ejecutado exactamente igual que la primera vez que encontramos el error, mismo ambiente, misma data, mismas entradas, todo debe ser replicado con exactitud.
 - Confirma que el problema está solucionado y funciona como se esperaba
 - Requiere menos tiempo para la verificación porque está limitado al problema específico.
 - No requiere ninguna configuración de entorno nuevo. La nueva prueba se realiza con los mismos datos y el mismo entorno con una nueva compilación.
-