

Unidad 3. Herramientas para la ejecución de pruebas

Objetivo de la unidad: Conocer, analizar y utilizar herramientas de acceso abierto para la ejecución y simulación del plan de pruebas.

Temas de la unidad

3.1 Herramientas para sistemas web

3.2 Herramientas para sistemas móviles

Introducción

Iniciemos considerando que la prueba del software *es una fase crítica que muchas veces es tediosa para alcanzar la finalización y mejorar la precisión del producto.*

Años atrás los probadores invertían horas probando una funcionalidad en particular y la mayoría de las veces no lograban obtener resultados del 100%, sin embargo actualmente existen herramientas inteligentes de prueba de software para facilitar las pruebas manuales con las que se permite reducir los errores y el tiempo de prueba con resultados más precisos que antes. A su vez existen también otras herramientas que permiten automatizar el proceso de pruebas del software.

Las pruebas de software son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada o stakeholder. Es una actividad más en el proceso de control de calidad del SW.

Según William Howden “una prueba de SW es un conjunto de actividades que pueden y deben planearse por adelantado para realizarse de manera sistemática, por lo que durante el proceso de la prueba del SW, es necesario definir una plantilla, considerando un conjunto de pasos que incluyen métodos de prueba y técnicas de diseño de casos de prueba específicos. Probar es la parte inevitable de cualquier esfuerzo responsable por desarrollar un sistema de SW”.

La prueba de SW es un proceso que usualmente es conocido como validación y verificación (V&V), en el que se abarcan muchas actividades de aseguramiento de calidad de SW.

- Verificación: ¿Construimos el producto correctamente?
- Validación: ¿Construimos el producto correcto?

Herramientas de prueba automatizadas

Las herramientas de prueba de software automatizadas ayudan a ejecutar pruebas funcionales y de regresión en la aplicación. Estas herramientas deben producir resultados consistentes con los datos entrantes proporcionados

Existen muchas herramientas de prueba de software disponibles en el mercado, cada una con sus propias ventajas y características. exploremos algunas de las mejores herramientas para facilitar las pruebas de software

Qué es la automatización de pruebas

Se conoce como **automatización de pruebas** al proceso de realizar pruebas de manera automática, gestionando los datos para poder aumentar la calidad del software desarrollado. Las pruebas del software deben ser automatizadas para acelerar los tiempos, asegurar la calidad del software, disminuir los costos y la carga de trabajo de los equipos para que puedan centrarse en otras tareas que generen valor.

Las pruebas de software realizadas en forma manual y repetitiva **que consumen mayor tiempo y recursos, además de que se exponen al error humano** deben ser realizadas de forma automática utilizando alguna herramienta específica para ello. Las pruebas manuales se orientan principalmente en cuanto a la interfaz de usuario, mientras que las pruebas automatizadas lo hacen en arquitecturas orientadas a servicios.

Criterios de la automatización de pruebas

Para que las pruebas sean automatizadas deben cumplir con una serie de criterios como:

- **Deben ser repetibles.** Una prueba automatizada debe poder realizarse muchas veces, es decir que sea aplicable a un proceso que requiere ser validado de muchas formas.
- **Deben ser objetivas.** La automatización de pruebas debe realizarse bajo criterios objetivos, pues no se trata de una cuestión de opinión. Para criterios subjetivos se cuentan con otro tipo de herramientas o pruebas, como los test de usuario donde aportan su opinión.
- **Deben ser determinantes.** La automatización de pruebas debe entregar un resultado que sea determinante, y que si se repite dicha prueba, el resultado vuelva a ser el mismo (o muy aproximado).

¿Cuándo no es recomendable hacer automatizaciones?

Funcionalidades inestables
Resultados no predecibles
Tiempo y recursos insuficientes

Diferencia entre pruebas manuales y pruebas automatizadas

Con las **pruebas manuales** se necesitan la participación humana en mayor proporción que con las **pruebas automatizadas** en las que disminuye la intervención del hombre. Las pruebas manuales por su parte son importantes, pues **permiten un análisis más profundo y una perspectiva más real**. Son utilizadas comúnmente para evaluar **la usabilidad y la experiencia de usuario**.

Ambos tipos de pruebas son **complementarias y necesarias** en el proceso de desarrollo de software. Aquellos test que tratan con tareas repetitivas y tediosas son los ideales para ser automatizados, pues lo harán de forma mucho más rápida y eficiente. Además, al ser repetibles, se pueden realizar cada vez que sean necesario, como cuando se produce un cambio en el código.

Beneficios de la automatización de pruebas automatizadas

- a) **Mayor precisión:** Con las pruebas automatizadas se eliminan los típicos errores causados por los desarrolladores cuando se realizan pruebas manuales. Cada vez que se ejecuta una prueba automatizada se hace de forma precisa, ejecutando siempre las mismas operaciones, lo que ofrece un resultado más preciso y fiable.
Durante el ciclo de desarrollo se deben ejecutar las pruebas cada vez que el software es modificado, pudiendo ejecutarse de forma automática cada vez que se realicen cambios. Ejecutar pruebas de forma automática siempre que se realice un cambio asegura la calidad en el desarrollo del software.
- b) **Reducción de tiempo y costo:** Las pruebas automatizadas tienen un tiempo de ejecución menor, pero su costo inicial de implementación es alto. Sin embargo, **el ahorro de tiempo amortigua esta inversión inicial consiguiendo que sean mucho más rentables a nivel económico**. Una empresa conseguirá utilizando pruebas automatizadas reducir notablemente el tiempo necesario para realizarlas, y disminuir los costos asociados.
- c) **Mayor reutilización:** Las pruebas automatizadas **pueden repetirse una y otra vez para testear cómo reacciona el software**. Al realizar las pruebas distintas veces se detectarán errores o procesos no optimizados que pueden ser corregidos. Gracias a poder automatizar las pruebas y repetirlas constantemente sin costos adicionales de desarrollo y de forma más rápida,

se podrán **entregar actualizaciones o nuevas versiones en mucho menos tiempo.**

- d) **Incrementa la productividad de los equipos de trabajo:** Los equipos de trabajo se benefician de la automatización de pruebas al no tener que ocupar su valioso tiempo en realizar pruebas manuales constantemente. **Su motivación e implicación al dedicarse a otras tareas más complejas hará que sean mucho más productivos**, ayudando a incrementar el valor del software desarrollado.

¿Cuándo y porque automatiza las pruebas del SW?

Automatizar las pruebas de software tiene sus ventajas y desventajas, y depende del proyecto, tiempo, costo, la metodología a utilizar, y el contexto de desarrollo, se realiza con el propósito de cumplir los objetivos de la mejor manera posible que permita desarrollar un SW de calidad.

¿Cómo determinamos que un producto de SW es de calidad?

- La **calidad del diseño** de algún producto determinado se refiere a las características que los diseñadores especifican para el producto: el tipo de materiales, tolerancias y especificaciones de desempeño, todo contribuye a la calidad del diseño, la calidad de un producto se incrementa si se fabrica de acuerdo a las especificaciones.
- La **calidad del diseño de una aplicación de SW** se refiere al grado en el que el diseño cumple las funciones y características especificadas en el modelo de requerimientos.
- La **calidad de la conformidad** se centra en el grado en el que la implementación se apega al diseño y en el que el sistema cumple sus metas derequerimientos y desempeño.

Según David Garvin [Gar,84] “La **calidad** es un concepto complejo, de facetas múltiples, que puede definirse desde cinco diferentes puntos de vista:

1. **Desde el punto de vista trascendental**, dice que la calidad es algo que se reconoce de inmediato, pero que no es posible definirlo explícitamente.
2. **Desde el punto de vista del usuario**, se concibe a la calidad en los términos de las metas específicas del usuario final, si el producto las satisface, tiene calidad.
3. **Desde el punto de vista del fabricante:** la calidad se define en términos de las especificaciones originales del producto, si este las cumple tiene calidad.
4. **Desde el punto de vista del producto**, sugiere que la calidad tienen que ver con las funciones y características inherentes de un producto, si sirve para lo que se hizo, tiene calidad.
5. **Desde el punto de vista basado en el valor**, se mide a la calidad de acuerdo con

lo que un cliente está dispuesto a pagar por un producto.”

Atributos de la calidad del estándar ISO 9126

Este estándar señala seis atributos clave para determinar que una aplicación de SW es de calidad:

1. **Funcionalidad:** Grado en el que el SW satisface las necesidades planteadas según las establecen los atributos siguientes: adaptabilidad, exactitud, interoperabilidad, cumplimiento y seguridad.
2. **Confiabilidad:** Cantidad de tiempo el SW se encuentra disponible para su uso, según lo indican los siguientes atributos: madurez, tolerancia a fallas y recuperación.
3. **Usabilidad:** Grado en el que el SW es fácil de usar, según lo indican los siguientes atributos: entendible, aprendible y operable.
4. **Eficiencia:** Grado en el que el SW emplea óptimamente los recursos del sistema, según lo indican los siguientes subatributos: comportamiento del tiempo y de los recursos.
5. **Facilidad de recibir mantenimiento:** Facilidad con la que pueden efectuarse reparaciones al SW, según lo indican los atributos: analizable, cambiable, estable, susceptible de someterse a pruebas.
6. **Portabilidad:** Facilidad con la que el SW puede llevarse de un ambiente a otro según lo indican los siguientes atributos: adaptable, instalable, conformidad y sustituible.

Nota: Es importante considerar que esos factores al igual que los establecidos por otras instituciones no necesariamente conducen a *una medición directa*, pero *proporcionan una base útil para hacer mediciones indirectas y una lista de comprobación excelente para evaluar la calidad del sistema.*

Entre los **factores** que permiten determinar que **una aplicación de SW es buena calidad** se encuentran: Que la aplicación sea:

- Fácil de usar
- Fácil de aprender
- Eficiente
- Robusta
- Rica en características propias.

Para evaluar estos atributos es importante que consideremos:

1. **Que sea intuitiva** se refiere al grado en el que la interfaz sigue patrones esperados de uso, de tal modo que un novato sin mucha capacitación la pueda utilizar (Usabilidad y facilidad de aprender.)
 - a) ¿Es fácil de comprender?
 - b) ¿Sus operaciones son fáciles de localizar e iniciar?
 - c) ¿La interfaz utiliza una metáfora reconocible?

d) ¿La entrada esta especificada de modo que economiza el uso del teclado o de ratón?

e) **¿La entrada sigue las tres reglas de oro?**

1. Dejar el control al usuario
2. Reducir la carga de memoria del usuario
3. Hacer que la interfaz sea consistente

f) **¿La estética del diseño ayuda a su comprensión y uso?**

2. **Que sea eficaz**, se refiere al grado en el que es posible localizar o iniciar las operaciones y la información:

- a) ¿La distribución y estilo de la interfaz permite que un usuario introduzca con eficiencia las operaciones y la información?
- b) ¿Una secuencia de operaciones o entradas de datos puede realizarse con economía de movimientos?
- c) ¿Los datos de salida o el contenido están presentados de modo que se entienden de inmediato?
- d) ¿Las operaciones jerárquicas están organizadas de manera que minimizan la profundidad con la que debe navegar el usuario para hacer que alguna se ejecute?

3. **Que sea robusta**, Se refiere al grado en el que el SW maneja entradas erróneas de datos o en el que se presenta interacción inapropiada por parte del usuario?

- a) ¿El SW reconoce el error si entran datos en el límite de lo permitido o más allá, y continúa operando sin fallar ni degradarse?
- b) ¿La interfaz reconoce los valores cognitivos o de manipulación y guía en forma explícita al usuario de vuelta al camino correcto?
- c) ¿La interfaz proporciona un diagnóstico y una guía útiles cuando se descubre una condición de error, asociada con la funcionalidad del SW?

4. **Que sea rica en características**, grado en el que la interfaz provee un conjunto abundante de características.

- a) ¿Puede personalizarse la interfaz según las necesidades específicas del usuario?
- b) ¿La interfaz tiene gran capacidad para permitir al usuario identificar una secuencia de operaciones comunes con una sola acción o comando?

Herramientas de acceso abierto que permiten ejecutar y simular los casos de prueba en sistemas móviles.

A medida que a una aplicación se le agregan más características y código, aumentará el esfuerzo que se requiere para probarla hasta que se alcance a un punto en el que se necesite encontrar un proceso más eficiente para crear, ejecutar e informar sobre sus casos de prueba. Las herramientas de automatización de pruebas móviles ayudan a aumentar la productividad de los evaluadores y aumentar la cobertura del código.

Hay muchas herramientas de automatización de pruebas móviles disponibles en el mercado y cada día se lanzan nuevas. A continuación se proporciona una lista de las más utilizadas actualmente:

1. TestComplete
2. Tricentis Tosca
3. Ranorex
4. Katalon Studio
5. Appium Studio
6. TestProject

Fuentes de referencia

1. Automatizar pruebas de SW, ¿Cuándo y por qué?:
<https://cl.abstracta.us/blog/automatizar-pruebas-de-software/>
2. Las 27 mejores herramientas de pruebas de **rendimiento** para usar en 2021;
liga: <https://kinsta.com/es/blog/herramientas-pruebas-rendimiento/#3-apache-jmeter>
3. Las 11 mejores herramientas de automatización de pruebas para interfaces de usuario 2021, y revisa el contenido en la liga:
<https://ciberninjas.com/10-mejores-herramientas-pruebas-ui/>
4. Introducción a la Automatización de pruebas de SW;
https://www.youtube.com/watch?v=R_hh3jAqn8M
5. Automatización de Pruebas de SW | Automatización WEB | Pruebas de Aplicaciones Web
<https://www.youtube.com/watch?v=eFoMFWn5by4>
6. Probar aplicaciones para dispositivos móviles;
<https://www.youtube.com/watch?v=6HMhdiIhvDI0>