



SECCIÓN VII: INTRODUCCIÓN A LA AUTOMATIZACIÓN DE PRUEBAS CON SELENIUM IDE



CONCEPTOS RELACIONADOS A LA AUTOMATIZACIÓN DE PRUEBAS



La automatización es un término “tendencia” estos días dentro del plan de carrera de muchos testers y consiste en utilizar la tecnología para agilizar la ejecución de casos de prueba.



Esto representa un hito importante en todo equipo de Testing, ya que implica incluir habilidades técnicas en el equipo y alinearse con metodologías de desarrollo ágiles como Scrum, Kanban o DevOps.

CONCEPTOS RELACIONADOS A LA AUTOMATIZACIÓN DE PRUEBAS

- La automatización permite enfocar el esfuerzo de testing en la generación de scripts para la ejecución de pruebas tantas veces como se necesite. Es un esfuerzo a largo plazo, para así dedicar a los testers a actividades más analíticas como el relevamiento de información y diseño de casos de prueba.
- Además, las pruebas automatizadas tienen múltiples beneficios, entre ellos que su resultado no depende de la interpretación, pues siempre será ejecutado de la misma manera.

¿CUÁNDO AUTOMATIZAR?

- En general, la automatización de pruebas permite agilizar actividades de la ejecución de pruebas y siempre que el beneficio de automatizar sea mayor al esfuerzo que implica, entonces será bienvenida.
- Conviene automatizar en los siguientes casos:
 - Ejecución de CPs repetitivos
 - Set de datos variados para el mismo CP
 - Varios ambientes de pruebas
 - Test de regresión varios

¿QUÉ AUTOMATIZAR?

- Generación de datos de prueba
- Re-ejecución de casos de prueba
- Pruebas de regresión
- Pruebas de integración
- Verificación de ambientes
- Medición de tiempos de respuesta

¿QUÉ SE NECESITA SABER PARA AUTOMATIZAR?

- Conocer el proceso manual de ejecución de la prueba a automatizar
 - Es especialmente importante el paso a paso y resultado esperado
- Si se automatizará la prueba de un servicio web
 - Será necesario conocer la URL y formato de datos de entrada del servicio
- Si se automatizará la prueba de una aplicación web
 - Es valioso el conocimiento básico en HTML para extraer información sobre cómo se identifican los elementos de la página (campos de texto, listas, checkboxes, radio button, botones, etc.)
- Conocer alguna herramienta de automatización
 - Algunas requieren conocimientos técnicos en algún lenguaje de programación como Java o Javascript

HERRAMIENTAS DE AUTOMATIZACIÓN DE PRUEBAS

- En el mercado existen diferentes herramientas de automatización, algunas gratuitas y otras licenciadas, cada una de ellas tiene diferentes niveles de complejidad y aplicación.
- Entre las más conocidas se tienen herramientas para automatización de:
 - Sistemas web
 - Aplicaciones móviles
 - Aplicaciones de escritorio
 - Servicios web
- Algunos ejemplos de herramientas
 - Selenium
 - Katalon
 - UFT
 - Appium
 - Cypress

SELENIUM IDE



Es un entorno de pruebas de software para aplicaciones basadas en la web.



Provee una herramienta de grabar/reproducir para crear pruebas sin usar un lenguaje de scripting para pruebas.



Incluye también un lenguaje específico de dominio para pruebas (Selenese) para escribir pruebas y luego tiene un conversor compatible con un amplio número de lenguajes de programación populares.

SELENIUM IDE – CÓMO SE AUTOMATIZA



Instalar la extensión “Selenium IDE”



Abrir la herramienta



Preparar el caso de prueba



Configurar la aplicación



Encender el modo grabación



Ejecutar el caso de prueba manualmente (caso feliz)



Parar la grabación

SELENIUM IDE — COMANDOS

Los tests automatizados bajo Selenium IDE quedan representados como una serie de comandos en el lenguaje Selenese.

Los más utilizados son:

- Open: abre un navegador en la URL indicada en la configuración del test.
- Click: hace clic sobre un elemento del sistema web. Por ejemplo una imagen, hipervínculo o botón.
- Type: escribe en un campo de texto del sistema web. Por ejemplo un formulario.

En general, estos comandos permiten la reproducción de acciones cotidianas para la ejecución de un caso de prueba en un sistema web.

SELENIUM IDE – COMPROBACIONES

Una de las características más versátiles de Selenium IDE es la posibilidad de implementar comprobaciones de resultado esperado y se denominan 'asserts' en la herramienta. Estos asserts permiten incorporar verines de resultado esperado sobre el test automatizado. Por ejemplo, para evaluar un mensaje.

Los más utilizadas son:

- Text: valida un mensaje
- Element present: valida si un elemento (botón, campo, mensaje, imagen) está presente en la pantalla
- Enabled: valida si un elemento está habilitado o no en el sistema

DESPUÉS DE APRENDER, VIENE LA EMOCIÓN DE APLICAR LO APRENDIDO!

MUCHAS GRACIAS POR PARTICIPAR EN ESTA CAPACITACIÓN

