

El último tema que vamos a estar viendo es automatización. Automatización es un tema muy complejo, pero que me interesa que puedan por lo menos tomar el concepto de qué es la automatización. Que no sea algo misterioso, digamos. Que ustedes puedan ver cómo se automatiza en un ejemplo. Y que puedan entender los conceptos básicos de automatización. Cuándo usar la automatización. Básico. No podemos profundizar mucho. Es un tema muy difícil, pero nada, por lo menos sentar las bases y que entiendan de qué se trata, y entonces, la idea es que ustedes tengan un panorama de todas las herramientas. Si, esto ya es lo último. Automatización es lo más avanzado, pero fíjense que aprendieron cómo hacer casos de prueba, aprendieron cómo hacerlo basado en casos de uso, cómo hacerlo basado en metodología ágil como hacerlo basado en un requerimiento funcional. Aprendieron lo que es la metodología ágil cómo se trabaja en una metodología ágil, aprendimos cómo testear un servicio web de web con postman. Entonces, bueno, los distintos tipos de técnicas que hay para hacer los casos de prueba, de lo que es, los distintos tipos de cobertura que se hacen de caja blanca, de caja negra, que en la partición equivalencia, y bueno, todo esto que vimos como para que ustedes tengan sus herramientas para trabajar. Ahora vamos a meternos en el tema automatización. Esto es así...

En esta unidad, digamos, separaremos nociones básicas de automatización, vamos a tomar simplemente algunos conceptos de automatización. Y entonces, hacemos una definición. Vamos a ver cómo se utiliza, cuándo se utiliza, y vamos a ver un ejemplo. La automatización es una herramienta más dentro del testing cuyo principal objetivo es la reducción de tiempos. No sustituye a otras disciplinas del testing. Entonces, ¿qué hacemos? ¿automatizamos o no automatizamos? bueno, lo que automatizamos son chequeos, comprobaciones que los tester manuales previamente han detectado antes, ciertas pruebas de regresiones, smoke test. No sé si se acuerdan que hablamos en un momento del cuerpo de la regresión. Que las regresiones retestean cosas que ya se testearon. Entonces, si uno considera que va a retestear muchas veces, conviene automatizar porque es mucho más eficiente, es más veloz que testear todo en forma manual.

Pero bueno, la automatización para qué sirve porque elimina el error humano y permite la re ejecución de una prueba en las mismas condiciones, pruebas menos complejas que las manuales garantizan el funcionamiento básico del proceso de negocio de extremo a extremo y hay una reutilización, en la misma prueba con distintos lotes de datos, sobre distintos navegadores, sobre distintos componentes digamos, componentes reutilizables, bueno que se reutilizan, ahora el tema con la automatización es que hay que tratarla con un proyecto de software, porque hay que tratarla como un proyecto de software, porque es un proyecto de software digamos, la automatización requiere de programación, es una programación no tan compleja como como podría ser un desarrollo, un desarrollo de software más importante pero sigue siendo un desarrollo del software, entonces para nosotros para testear un software que hay que desarrollar hacemos otro desarrollo para automatizar los tests para testear a un desarrollo, se entiende, es complicarse la vida un poco también, porque es hacer un programa para tener un programa, pero la verdad que es muy útil cuando hay un retés permanente entonces decimos es un proyecto de software entonces requiere tiempo y recursos, personal especializados no cualquiera automatiza, requiere planificación, requiere diseño, requiere desarrollo por ejemplo y test, o sea, hay que testear el programa que nosotros desarrollamos para testear a otro programa, entonces es fundamental realizar la gestión, mantenimiento y evolución del test, cambios en el sistema afecta la ejecución de las pruebas, entonces qué pasa, si hay cambios quizás hay que mantenerlo también este a este programa automatizado, hay que hacerle mantenimiento, cuando cambie algo del sistema principal hay que mantener el sistema el sistema que lo prueba, se conocen los beneficios de las pruebas

automatizadas sin ver sus costos, normalmente dice que bueno automatizar las pruebas, si bueno pero tiene costo, no es tan clara las características de la inversión asociada entonces acá tengo actividades iniciales hacen visibles los resultados a medio y largo plazo y la ejecución de pruebas manuales es dinámica y rápida en tiempo de respuesta, hacer una prueba manual es más rápido que ponerse a automatizar y acá tenemos un ejemplo dónde tenemos desarrolladores, el desarrollador programa, espera por testing, se testea, se reportan los defectos, se comunican los defectos, se reproducen los defectos y se arreglan los defectos estos son todos los pasos normalmente en un circuito manual, ahora en un circuito automatizado que pasa se desarrolla, se ejecuta el testing automatizado, se reproduce el defecto y se arregla como que tenemos menos pasos, está bien, y es si bien yo les decía recién es más rápido hacer un test manual que uno automatizado, por qué, porque el manual me siento y pruebo, el automatizado tengo que ponerme a hacer un desarrollo para automatizar, el tema es que una vez que ya tengo el desarrollo de hecho las pruebas son mucho más rápidas y si hay que re testear ahorro tiempo, por eso hay que ver bien qué es lo que conviene automatizar. Tenemos la productividad de la automatización está en la re ejecución y bueno la cuestión es que va a haber siempre una convivencia, un equilibrio entre pruebas manuales y pruebas automáticas, no vamos a poder automatizar todo, la ventaja son qué esta inversión de automatizar producen mejores y más satisfactorios resultados, proporcionan información real y objetiva de los criterios de calidad, reduce los costos de prueba en el medio y largo plazo no en el corto plazo, permite un incremento en la cobertura de las pruebas, reduce el tiempo de ejecución de los ciclos de prueba, imagínense que es tocar un botón y que se testee esté todo sólo, entonces se reduce el tiempo, los ciclos de pruebas son más cortos, permite un desarrollo interactivo y disminuyen los errores humanos y las desventajas que bueno no es posible automatizar el 100% de las pruebas porque no es rentable automatizar el 100% de las pruebas de las pruebas, es complicado volver al estado original en que se realizaron las pruebas, una vez que se automatiza cambia el estado de los datos, de la base de datos por ejemplo, del torno, bueno a veces es complicado volver al estado original, pero hay que generar un buen dato de prueba y contar con una buena herramienta de automatización, o sea, pero el plato de prueba hay que generarlo porque hay que alimentar de alguna forma al sistema que va a probar, o sea, un sistema probador necesita un lote de pruebas sobre el cual probar, entonces se lo tenemos que dar nosotros y bueno hay costos de licencias, por supuesto, también si vamos a utilizar una herramienta, se requiere tiempo y esfuerzo tal como si fuera un proyecto de desarrollo de software lo que decíamos antes, requiere planificación y diseño, conocer que queremos probar, cuáles son las entradas y la definición del plan de pruebas, requiere una implementación, planificar que merece la pena probar, que merece la pena automatizar y la ejecución que es definir el set de datos a utilizar y ejecutar desatendidamente en un conjunto de pruebas, porque esto por supuesto nosotros le damos un set de datos y es algo que corre solo y desatendido, entonces definir el set de datos a ejecutar efectivamente es un tema que hay que ver cómo armar este set de datos y bueno por último, que se detectan fallos pero bueno hay que evaluarlo, por qué a que se debe el fallo, nosotros sabemos que falló, el test automatizado nos va a decir acá fallo, pero bueno estas nosotros despues tenemos que ver por qué fallo, hacer un análisis sobre eso y qué pruebas automatizamos, bueno hay unas unas pruebas que se llaman “record & play” que son pruebas que digamos, es un tipo de automatización que automatiza los clicks que se hacen en pantalla, pero no son las que más retornos de inversión aportan porque la interfaz de usuario es propensa a cambiar, entonces si yo hago una automatización que me hace un clic en un botón, hace clic en una celda, me carga unos datos, hace clic en otra celda, descarga datos, hace clic en el botón de guardar, eso funciona todo bárbaro hasta que me cambian la interfaz, si me cambio la interfaz gráfica ya todo esos clics y todo eso que programé no me sirve más, entonces ese tipo de automatización por ahí no es tan bueno, la automatización de pruebas de interfaz de usuario tiene un costo de mantenimiento mayor que el resto, o sea, todo lo que es este tipo de pruebas de interfaz

de usuario tiene un costo mayor, una automatización de los tests API de lo que serían los servicios web, es más sencillo eso, porque eso se automatiza y ya está, digamos, eso no va a tener cambios o cambios tan profundos, a este automatizar en el grado adecuado en los niveles adecuados para nuestra aplicación, si, hay que pensar bien qué vamos a automatizar, entonces acá tenemos un gráfico donde se nos está mostrando qué digamos, como debería ser esto, para estar equilibrado, tendríamos que tener muchos test auto-unitarios, automáticos, bastante test a nivel API, de integración, de componentes y de servicios y tenemos que tener menos test de interfaz gráfica automatizado, la interfaz gráfica, la mayoría de los tests automatizados tenían que ser unitarios, algunos de API y no tantos de los que son de interfaz gráfica y también bueno vamos a decir que muchas pruebas manuales y de automatización de pruebas con la interfaz de usuario genera un gráfico de este tipo, es un gráfico del tipo “cono de helado”, o sea, tenemos muchos tests manuales y casi casi ningún test unitario, casi ningún test, se acuerdan que es unitario unitario el que estaba bien bien pegadito al trabajo del desarrollador, este es un patrón inverso al que vimos recién y este no es un buen patrón, este es lo que no habría que hacer, bueno nada les conté digamos una idea general de lo que es automatización y les conté, digamos, esto de que no siempre se puede automatizar todo, que la automatización también es un es un desarrollo de software, entonces como la automatización es un desarrollo de software bueno no todos los tests se automatizan porque la mayoría de los tester no desarrollan, no son programadores, pero hay algunos que son más técnicos que son desarrolladores que testean, que desarrollan pero son más técnicos y pueden hacer este tipo de trabajo y esos son muy bien pagados, están muy bien pagados en el mercado, son personas que ganan muy bien, entonces es interesante, es un es un tema interesante de comprenderse, esto simplemente es lo básico lo que estamos viendo y ahora vamos a ver un ejemplo, vamos a ver un ejemplo, para que terminen de entender esto, si bien es algo técnico el ejemplo les va a servir para terminar de cerrar la idea acerca de la automatización.