**La importancia del testing de software y de la automatización de pruebas**

El testing de software **es una de las actividades más importantes y fundamentales** en el desarrollo de un proyecto, ya que posibilita los procesos, métodos de trabajo y herramientas necesarias para garantizar la calidad de cualquier desarrollo. Sin embargo, hoy en día continuamos encontrándonos con problemas a la hora de comprender el testing de software como una inversión, incluso en los proyectos más tradicionales donde asumíamos que las actividades de pruebas estaban normalizadas.

Con el fin de poder detectar a tiempo cualquier error y garantizar que el producto cumple con todas las premisas establecidas, el testing debe existir en todas las fases de un proyecto: desde la toma de requerimientos en cliente, pasando por las reuniones de seguimiento, hasta la entrega del producto final. Es más, un proyecto carente de este proceso en todas sus fases acaba generando un mayor coste económico y un mayor esfuerzo durante la fase de pruebas.

En la mayoría de los casos, durante la planificación de un proyecto, el equipo de QA tiene unos días establecidos para pruebas, normalmente coincidiendo días antes de un despliegue. En caso de demoras durante el desarrollo, la fase de testing suele ser la más perjudicada, llegando incluso en algunos casos a ser eliminada por falta de tiempo. Se considera en muchas ocasiones que el testing de software es una fase costosa y a la vez prescindible, cuando realmente lo que consigue es asegurar la calidad del producto, por lo que estaríamos hablando de una inversión asegurada. Y es que, la detección de errores en fase final, posterior corrección de éstos y el testing de la corrección genera mayor esfuerzo en horas y mayor inversión económica, que el haber planificado la existencia del equipo de QA en todo el ciclo de vida del desarrollo desde el principio, pudiendo así prevenir los errores de antemano. La clave está en prevenir antes, no detectar al final.

Un buen entendimiento funcional y técnico del negocio por parte del equipo de QA garantiza, a su vez, qué tipo de pruebas serán necesarias aplicar en cada fase de testing y cómo aplicarlas. Hablamos en este caso de las **pruebas automatizadas.**Está más que comprobado que actualmente se consume el doble tiempo en probar una funcionalidad que en desarrollarla. Asimismo, las pruebas exploratorias tan solo consiguen detectar un porcentaje de errores, influido también por el poco tiempo que se dispone para realizarlas y el número de personas disponibles para llevarlo a cabo.

Las **pruebas automatizadas** consiguen la eliminación de errores surgidos a raíz de las pruebas realizadas de forma manual. Además, para cada actualización de software es preciso la realización de tests de regresión que garanticen que el producto continúa funcionando correctamente y que ninguna de las modificaciones ha producido uno o más errores colaterales. Los tests de regresión producen un esfuerzo extra en horas, así como un mayor gasto económico, con lo que se hace necesaria la automatización, ahorrando de esta manera tiempo de proyecto y mejorando la **fiabilidad**. Para ello, hay que analizar qué casos de prueba son proclives a ser automatizados, siempre priorizando aquellos más críticos y los que sean propensos a generar errores incontrolados.

Por ejemplo, en una aplicación siempre va a generar más errores la gestión de un formulario extenso y envío posterior de datos, o bien una pantalla de búsqueda de información con su filtro correspondiente, que el acceso a una o más pantallas desde un menú. Por ende, todas aquellas pantallas que interactúen con *backend* o servicios serán las más prioritarias a ser automatizadas. Por otro lado, a través de un ciclo de **integración continua** conseguiremos minimizar el tiempo y esfuerzo en las pruebas de regresión, acelerar los desarrollos y actualizaciones de producto, reducir el TTM *(Time to Market),* a la vez que se consigue un incremento de la cobertura de pruebas, minimizando los errores en la ejecución.

En definitiva, es necesario ser consciente de la **importancia del testing de software** y de la inclusión del equipo de calidad en todas las fases de desarrollo de un producto, sin que ello sea considerado un perjuicio económico en la planificación de un proyecto. Sin duda alguna, es mucho mejor la detección de errores en su justo momento generando así tiempo material para subsanarlos, que detectarlos antes de la entrega del producto o de un importante paso a producción.

**Cristina López-Goicochea Juárez**