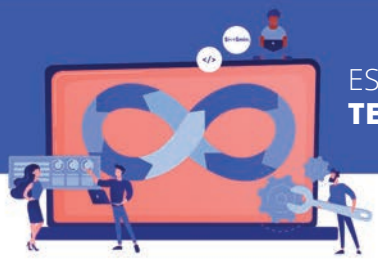

Ariel La Paz, Facultad de Economía y Negocios, Universidad de Chile

"Muchas aplicaciones fallan por la falta de planificación, monitoreo, documentación y pruebas"

El testing de software es una fase clave en el proceso de desarrollo de software. La falta de aplicación de pruebas podría ocultar errores de construcción y/o diseño, que redundarían en fallas técnicas o productos que no cumplen con las expectativas del cliente, en cuanto a la funcionalidad de las herramientas.

¿Cómo definiría el concepto de testing de software?

La prueba de software es el procedimiento mediante el cual se evalúa la calidad técnica, o interna, y funcional de una aplicación. Algunos criterios de evaluación son estándares y estudian aspectos técnicos de la aplicación en particular; mientras que otras variables miden la capacidad de integrarse con otras aplicaciones, o la adaptabilidad hacia el usuario y los procesos de negocio en que se usará. Para las evaluaciones, se utilizan comúnmente las normas ISO 9126 y 25000.

Las pruebas pueden aplicarse a aplicaciones adquiridas como software empaquetado, y a aquellas desarrolladas internamente en una compañía o mediante outsourcing, siendo especialmente crítico realizarlas en desarrollos nuevos.

¿Qué objetivos persigue el testing?

El primer y mayor propósito del testing de software es lograr que las aplicaciones lleguen al punto de robustez técnica, de acuerdo al uso y propósito que se requiere. Esto, por ejemplo, se traduce en sistemas que no tienen caídas inesperadas, o que no colapsan con una cantidad de transacciones, usuarios o transferencia de datos. Por otra parte, las pruebas funcionales deben lograr evaluar que el software hace lo que se espera en términos de alineamiento y apoyo a tareas y procesos de negocio.

La falta de aplicación de pruebas podría ocultar errores de construcción y/o diseño del software, que redundarían en fallas técnicas y/o en la construcción de productos que no cumplen con las expectativas del cliente, en cuanto a la funcionalidad de las herramientas.

¿Por es clave en el proceso de desarrollo de una solución?

Los desarrollos de software requieren de orden y sistematicidad no solo en la construcción del código y algoritmos, sino que en aspectos de la gestión del proyecto de desarrollo. Muchas aplicaciones fallan por la falta de planificación, monitoreo, documentación y pruebas, las que tienden a reducirse ante presiones presupuestarias y financieras. Ciertamente la realización de estas tareas consume recursos financieros y humanos. Deben considerarse como un "seguro", que, en el corto plazo, facilita la evaluación de cambios que surgen durante la construcción del software y, en el largo plazo, simplifica la mantención, escalabilidad, updates y upgrades, o recuperaciones ante fallas y catástrofes.

¿Cómo se estructura un buen proceso de testing?

De modo general, el testing se realiza en cinco pasos, que deben planificarse desde el propio inicio del proyecto. En el paso uno, se deben establecer los objetivos técnicos y funcionales del software sobre los que se ejecutará la evaluación. Segundo, se deben definir los tipos y momentos de la evaluación, durante la construcción, entrega y puesta en marcha. Un tercer paso diseña en detalle cada tipo de prueba, asignando responsables y plazos. En el paso cuatro, se ejecutan y documentan los resultados de las pruebas, para, en el paso final, concluir

{Síguenos en @RevGerencia y Revista Gerencia}

"Se ha identificado una tendencia a la falta de evaluación funcional adecuada y enfocada en el alineamiento y ajuste del software a modelos y procesos de negocio, y que tiene por resultado un desarrollo, implementaciones y uso deficiente de software"

la evaluación con recomendaciones o indicaciones de mejora. Las conclusiones deben indicar las pruebas realizadas, los objetivos logrados, así como las áreas de riesgo y limitaciones.

¿Cuáles son las pruebas más comunes?

Técnicamente, se evalúa la eficiencia, compatibilidad, fiabilidad, seguridad, portabilidad y mantenibilidad de una aplicación. Funcionalmente, evaluamos la adecuación de la aplicación a los procesos y tareas de los usuarios, así como la usabilidad que estos lograrán (por ejemplo, facilidad de uso, estética, y protección frente a errores, etc.).

Como se indicó, las pruebas se realizan en distintos momentos del desarrollo, incluyendo etapas tempranas, para validar los requerimientos de los usuarios; y en etapas previas al diseño, hasta etapas previas y posteriores a la implementación con datos de prueba, marcha blanca en ambientes controlados, y pruebas en uso, durante la explotación del sistema.

¿Qué tendencias son claves hoy en testing de software?

Aunque se incluye el testing funcional en las propias normas ISO, para la evaluación de la calidad de software y datos, la tendencia principal es realizar el testing técnico, desarrollado por personal informático. Se ha identificado una tendencia a la falta de evaluación funcional adecuada y enfocada en el alineamiento y ajuste del software a modelos y procesos de negocio, y que tiene por resultado un desarrollo, implementaciones y uso deficiente de software. Esto es especialmente crítico en software que abarca a varias unidades de una organización o que tiene un alto número de usuarios, herramientas y funciones.

¿Cómo el testing beneficia a los usuarios del software y de qué modo pueden participar?

Lo más importante es la comunicación fluida y un buen plan que destaque los tiempos y propósitos de incluir a los usuarios en el testing de software. En general, no se entiende como responsabilidad del usuario participar en el desarrollo de software. De hecho, la mayoría de las veces no se les compensa ni reconoce por esta labor, o ellos mismos no ven la importancia de su participación en estos procesos.

El jefe del proyecto de desarrollo debe ser capaz de comunicar efectiva y oportunamente a los usuarios por qué y en qué se les necesita. Un buen planteamiento ante el usuario también favorece la gestión del cambio, la aceptación de las herramientas de software y la capacitación. G

Ariel La Paz es Profesor Asociado del Departamento de Control de Gestión y Sistemas de Información de la Facultad de Economía y Negocios de la Universidad de Chile.