PREGUNTAS DE REPASO

# ¿Qué es un caso de prueba?

Un caso de prueba o test case es un conjunto de condiciones o variables bajo las cuales un analista determinará si una aplicación, un sistema de software (software system), o una característica de éstos es parcial o completamente satisfactoria. Es decir, un caso de prueba es un conjunto de pasos y resultados esperados que se crean a partir de los requisitos del software que se va a probar.

Debe crearse al menos un caso de prueba por cada requisito definido y tener en cuenta, todos los elementos de diseño, el uso de todo tipo de datos de entrada/salida y cada comportamiento esperado.

# ¿Qué dos técnicas o enfoques se utilizan para llevar a cabo el diseño de casos de prueba?

* Caja blanca: son pruebas estructurales en las cuales se conocen el código y siguiendo su estructura lógica, se pueden diseñar pruebas destinadas a comprobar que el código hace correctamente lo que el diseño de bajo nivel indica y otras que demuestren que no se comporta adecuadamente ante determinadas situaciones. Ejemplo: pruebas unitarias.
* Caja negra: son pruebas funcionales. Se parte de los requisitos funcionales, a muy alto nivel, para diseñar pruebas que se aplican sobre el sistema sin necesidad de conocer como está construido por dentro (Caja negra). Las pruebas se aplican sobre el sistema empleando un determinado conjunto de datos de entrada y observando las salidas que se producen para determinar si la función se está desempeñando correctamente por el sistema bajo prueba. Las herramientas básicas son observar la funcionalidad y contrastar con la especificación.

# ¿Por qué otros nombres se conocen las pruebas de caja blanca? ¿Y las de caja negra?

A las pruebas de caja blanca también se las conoce como pruebas de caja de cristal o pruebas estructurales.

A las pruebas de caja negra también se las conoce como pruebas funcionales o pruebas de comportamiento.

# En la estrategia de prueba del software se diferencia entre 4 tipos de prueba diferentes ¿Cuáles son?

* Pruebas unitarias: es una forma de comprobar el correcto funcionamiento de una unidad de código. Por ejemplo, en diseño estructurado o en diseño funcional una función o un procedimiento, en diseño orientado a objetos una clase.
* Pruebas de integración: se toman los módulos probados anteriormente en las pruebas unitarias y se construye la estructura del programa según las especificaciones del diseño. Se prueba que todos los elementos unitarios que componen el software, funcionan juntos correctamente probándolos en grupo.
* Pruebas de validación (o aceptación): este tipo de pruebas se llevan a cabo en el entorno real de trabajo con intervención del usuario final.
* Pruebas de sistema: en ellas se verifica si encaja de forma adecuada cada elemento alcanzado la funcionalidad y rendimiento total.

# Sobre qué se realizan pruebas en las conocidas pruebas unitarias. ¿Sabrías mencionar alguna herramienta específica para este tipo de pruebas?

Las pruebas unitarias, se prueban los distintos módulos de forma independiente. El propósito es eliminar errores en la lógica interna.

Herramienta: Junit.

# ¿Qué dos enfoques fundamentales existen para las pruebas de integración? Describe brevemente cada uno de ellos.

* Enfoque no incremental o big bang: Después de probar cada módulo por separado, se combinan todos y se ejecuta el programa al completo. De esta forma se encuentran gran cantidad de errores y la corrección es dificultosa.
* Enfoque incremental: se va probando el programa completo según se va construyendo, en este caso los errores son más fáciles de localizar. Se dan las estrategias:
  + Ascendente: la construcción y prueba del programa empieza desde los módulos de los niveles más bajos de la estructura del programa.
  + Descendente: la integración comienza en el módulo principal moviéndose hacia abajo por la jerarquía de control.

# ¿Qué técnicas se utilizan en las pruebas de validación?

* Prueba Alfa: se lleva a cabo por el cliente o usuario final en el lugar de desarrollo. Éste utiliza el software de forma natural bajo la observación del desarrollador que irá registrando los errores y problemas de uso.
* Prueba Beta: se lleva a cabo por el cliente o usuario final en su lugar de trabajo sin estar presente el desarrollador. En este caso es el usuario quien registra todos los problemas que encuentra reales y/o imaginarios en informa siguiendo los intervalos definidos en el plan de prueba. Posteriormente, el desarrollador lleva a cabo las modificaciones adecuadas y prepara una nueva versión del producto.

# La misión de las pruebas de sistema es ejercitar el software ¿Cómo se clasifican?

* Prueba de recuperación: aquí se fuerza el fallo del software para verificar si la recuperación se lleva a cabo de la forma esperada.
* Pruebas de seguridad: aquí se pretende verificar si el sistema está protegido contra accesos ilegales.
* Prueba de resistencia (Stress): enfrenta el sistema a situaciones que demandan gran cantidad de recursos, por ejemplo, diseñando casos de prueba que requieran el máximo de memoria, incrementando la frecuencia de datos de entrada que den problemas en un sistema operativo virtual, etc.

# ¿En qué tres categorías se organiza la documentación para las pruebas?

* Plan de pruebas: describe el alcance, enfoque, recursos y calendario de las actividades de pruebas. Identifica los elementos a probar, las características que se van a probar, las tareas que se van a realizar, el personal responsable de cada tarea y los riesgos asociados al plan.
* Especificaciones de prueba:
  + Especificación del diseño de la prueba: trata de identificar, a partir de los requisitos software, cada caso de prueba y los criterios de pasa/no-pasa.
  + Especificación de los casos de prueba: documenta los valores de entrada de cada caso de prueba y la salida prevista para cada uno de ellos.
  + Especificación de los procedimientos de prueba: describe los pasos necesarios para llevar a cabo cada caso de prueba.
* Informes de pruebas:
  + Informe sobre los módulos que se están probando.
  + Registro de los sucesos ocurridos durante la ejecución de las pruebas.
  + Informe de incidentes: cualquier imprevisto ocurrido en la ejecución de pruebas que precisa revisión.
  + Informe que resuma las actividades de prueba.