

4 Técnicas de diseño de pruebas

Objetivos de aprendizaje para las técnicas de diseño de pruebas

4.1 Identificando las condiciones de prueba y diseñando los casos de prueba

- ✓ Diferenciar entre una especificación del diseño de prueba, una especificación de casos de prueba y una especificación de procedimientos de prueba.
- ✓ Comparar los términos de condición de prueba, caso de prueba y procedimiento de prueba.
- ✓ Escribir casos de prueba:
 - Mostrando una clara trazabilidad a los requisitos;
 - Conteniendo un resultado esperado.
- ✓ Traducir casos de prueba hacia una especificación de procedimientos de prueba bien estructurada en un nivel de detalle relevante al conocimiento de los probadores.
- ✓ Escribir un programa de ejecución de prueba para un conjunto dado de casos de prueba, considerando priorización, y dependencias lógicas y técnicas.

4.2 Categorías de técnicas de diseño de prueba

- ✓ Recordar las razones que tanto los enfoques basados en especificación (caja negra) y los basados en estructura (caja blanca) para el diseño de casos de prueba son útiles, y listar las técnicas comunes para cada uno.
- ✓ Explicar las características y las diferencias entre las pruebas basadas en especificación, las pruebas basadas en estructura y entre las pruebas basadas en experiencia.

4.3 Técnicas basadas en especificación o de caja negra

- ✓ Escribir los casos de prueba de modelos de software dados usando las siguientes técnicas de diseño de prueba:
 - Particiones de equivalencia;
 - Análisis del valor límite;
 - Tablas de decisión;
 - Diagramas de transición de estado.
- ✓ Comprender el propósito principal de cada una de las cuatro técnicas, que nivel y que tipo de pruebas podrían usar la técnica y como la cobertura puede ser medida.
- ✓ Comprender el concepto de pruebas de caso de uso y sus beneficios.

4.4 Técnicas basadas en estructura o de caja blanca

- ✓ Describir el concepto y la importancia de la cobertura de código.
- ✓ Explicar los conceptos de cobertura de sentencias y de decisión, y comprender que estos conceptos también pueden ser usados en otros niveles de prueba que en las pruebas de componente (por ejemplo, procedimientos de negocio a nivel de sistema).
- ✓ Escribir casos de prueba de flujos de control dados usando las siguientes técnicas de diseño de prueba:
 - Pruebas de sentencias;
 - Pruebas de decisión.
- ✓ Evaluar la cobertura de sentencias y de decisión para completación.

4.5 Técnicas basadas en experiencia

- ✓ Recordar las razones para escribir casos de prueba basados la intuición, experiencia y conocimiento sobre defectos comunes.
- ✓ Comparar las técnicas basadas en experiencia con las técnicas de pruebas basadas en especificación.

4.6 Eligiendo las técnicas de prueba

- ✓ Listar los factores que influyen la selección de una técnica apropiada de diseño de prueba para una clase de problema en particular, tales como un tipo de sistema, riesgo, requisitos del cliente, modelos para modelación de caso de uso, modelos de requisito o conocimiento del probador.

4.7 Identificando las condiciones de prueba y diseñando los casos de prueba

4.7.1 Términos

Casos de prueba, especificación de casos de prueba, condición de prueba, datos de prueba, especificación de procedimientos de prueba, script de prueba, trazabilidad.

4.7.2 Introducción

El proceso de identificar las condiciones de prueba y de diseñar pruebas consiste en un número de pasos:

- ✓ Diseñar las pruebas identificando las condiciones de prueba.
- ✓ Especificar los casos de prueba.
- ✓ Especificar los procedimientos de prueba.

El proceso puede ser realizado en diferentes maneras, desde muy informal con poca o nada de documentación, hasta muy formal (como está descrito en esta sección). El nivel de formalidad depende del contexto de la prueba, incluyendo la organización, la madurez de la prueba y los procesos de desarrollo, restricciones de tiempo y de la gente involucrada.

Durante el diseño de prueba, la documentación base de pruebas es analizada para determinar que probar, es decir, para identificar las condiciones de prueba. Una condición de prueba está definida como un elemento o evento que podría ser verificado por uno o más casos de prueba (por ejemplo, una función, transacción, característica de calidad o un elemento estructural).

Establecer la trazabilidad desde las condiciones de prueba de vuelta hasta las especificaciones y requisitos permite tanto el análisis de impacto, cuando los requisitos cambian, como la cobertura de requisitos a ser determinada por un conjunto de pruebas. Durante el diseño de prueba el enfoque de prueba detallado es implementado basado en, entre otras consideraciones, los riesgos identificados (ver capítulo 5 para más información sobre análisis de riesgo).

Durante la especificación de casos de prueba, los casos de prueba y los datos de prueba son desarrollados y descritos en detalle al usar las técnicas de diseño de prueba. Un caso de prueba consiste en un conjunto de valores de entrada, precondiciones de ejecución, resultados esperados y post-condiciones de ejecución, desarrollados para cubrir cierta(s) condición(es) de prueba. El “Estándar para la Documentación de Prueba de Software” (IEEE 829) describe el contenido de las especificaciones del diseño de prueba.

Los resultados esperados deben ser producidos como parte de la especificación de un caso de prueba y deben incluir entradas, cambios a los datos y estados y cualquier otra consecuencia de la prueba. Si los resultados esperados no han sido definidos entonces un resultado plausible, pero erróneo, puede ser interpretado como el correcto. Los resultados esperados deben ser idealmente definidos previos a la ejecución de prueba.

Los casos de prueba son colocados en un orden ejecutable; esto es la especificación de procedimientos de prueba. El procedimiento de prueba (o script de prueba manual) especifica la secuencia de acción para la ejecución de una prueba. Si las pruebas son ejecutadas usando una herramienta de ejecución de prueba, la secuencia de acciones está especificada en un script de prueba (el cual es un procedimiento de prueba automatizado).

Los diversos procedimientos de prueba y los scripts de prueba automatizados son formados subsecuentemente en una agenda de ejecución de pruebas que define el orden en el que los diversos procedimientos de prueba, y posiblemente los scripts de prueba automatizados, son ejecutados, cuando serán llevados a cabo y por quien. La agenda de ejecución de pruebas tomará en cuenta tales factores como pruebas de regresión, priorización y dependencias técnicas y lógicas.

4.8 Categorías de técnicas de diseño de prueba

4.8.1 Términos

Técnicas de caja negra, técnicas basadas en experiencia, técnicas basadas en especificación, técnicas basadas en estructura, técnicas de caja blanca.

4.8.2 Introducción

El propósito de una técnica de diseño de prueba es identificar las condiciones de prueba y los casos de prueba.

Es una distinción clásica el denotar técnicas de prueba como caja negra o caja blanca. Las técnicas de caja negra (también llamadas técnicas basadas en especificación) son una forma de derivar y seleccionar las condiciones de prueba o los casos de prueba basadas en un análisis de la documentación base de pruebas, ya sea funcional o no funcional, para un componente o sistema sin referencia a su estructura interna. Las técnicas de caja blanca (también llamadas técnicas estructurales o basadas en estructura) están basadas en un análisis de la estructura interna del componente o sistema.

Algunas técnicas caen claramente en una simple categoría, otras tienen elementos de más de una categoría. Este programa se refiere a los enfoques basados en especificación o basados en experiencia como técnicas de caja negra y a los basados en estructura como técnicas de caja blanca.

Los rasgos comunes de las técnicas basadas en especificación:

- ✓ Modelos, ya sean formales o informales, son usados para la especificación del problema a ser resuelto, el software o sus componente.
- ✓ De estos modelos los casos de prueba pueden ser derivados sistemáticamente.

Los rasgos comunes de las técnicas basadas en estructura:

- ✓ Información sobre como el software está construido y usado para derivar los casos de prueba, por ejemplo, el código y el diseño.
- ✓ La medida de cobertura del software puede ser medida por los casos de prueba existentes y posteriores casos de prueba pueden ser derivados sistemáticamente para incrementar la cobertura.

Los rasgos comunes de las técnicas basadas en experiencia:

- ✓ El conocimiento y la experiencia de la gente son usados para derivar los casos de prueba.
 - El conocimiento de los probadores, desarrolladores, usuarios y otras partes interesadas sobre el software, su utilización y su entorno;
 - El conocimiento sobre defectos probables y su distribución.