Técnicas de prueba

DigitalHouse >
Coding School

Certified Tech Developer
The Ultimate Degree

Agenda

1. Categorías de técnicas de prueba
2. Técnicas de caja negra
3. Técnicas basadas en la experiencia

1 Categorías de técnicas de prueba

“

El objetivo de una técnica de prueba es **ayudar a identificar las condiciones, los casos y los datos de prueba.**

”

Elección de una técnica de prueba

La elección de la técnica de prueba a utilizar depende de los siguientes factores:

- Tipo y complejidad del componente o sistema
- Estándares de regulación
- Requisitos del cliente o contractuales
- Clases y niveles de riesgo
- Objetivo de la prueba
- Documentación disponible
- Conocimientos y competencias del probador
- Modelo del ciclo de vida del software
- Tiempo y presupuesto

Clasificación de las técnicas de prueba

Técnicas de caja negra:

Se basan en el comportamiento extraído del análisis de los documentos que son base de prueba (documentos de requisitos formales, casos de uso, historias de usuario, etc). Son aplicables tanto para pruebas funcionales como no funcionales. **Se concentran en las entradas y salidas sin tener en cuenta la estructura interna**.

Técnicas de caja blanca:

Se basan en la estructura extraída de los documentos de arquitectura, diseño detallado, estructura interna o código del sistema. **Se concentran en el procesamiento dentro del objeto de prueba**.

Técnicas basadas en la experiencia:

Aprovechan el conocimiento de desarrolladores, probadores y usuarios para diseñar, implementar y ejecutar las pruebas.

2 | Técnicas de caja negra

Partición de equivalencia

En esta técnica se dividen los datos en particiones conocidas como **clases de equivalencia** donde cada miembro de estas clases o particiones es procesado de la misma manera.

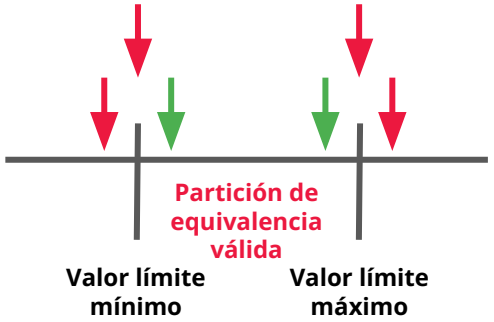
Las características de esta técnica son:

- La “partición de equivalencia válida” contiene valores que son aceptados por el componente o sistema.
- La “partición de equivalencia no válida” contiene valores que son rechazados por el componente o sistema.
- Se pueden dividir las particiones en subparticiones.
- Cada valor pertenece a solo una partición de equivalencia.
- Las particiones de equivalencia no válidas deben probarse en forma individual para evitar el enmascaramiento de fallos.
- La cobertura se mide de la siguiente manera:

$$\text{Cobertura} = \frac{\text{Particiones probadas}}{\text{Particiones identificadas}}$$

Análisis de valores límites

Es una extensión de la técnica de partición de equivalencia que solo se puede usar cuando la partición está ordenada, y consiste en **datos numéricos o secuenciales.**



- Se deben identificar los valores límites mínimo y máximo (o valores inicial y final).
- Se pueden utilizar 2 o 3 valores límites.
- Para 2 valores límites se toma el valor que marca el límite (como valor que corresponde a la partición válida), y el valor anterior o posterior que corresponda a la partición de equivalencia inválida.
- Para 3 valores límites se toma el valor que marca el límite, un valor anterior y otro posterior a ese límite.
- La cobertura se mide de la siguiente manera:

$$\text{Cobertura} = \frac{\text{Valores límites probados}}{\text{Valores límites identificados}}$$

Tabla de decisión

Esta técnica se utiliza para **pruebas combinatorias**, formadas por reglas de negocio complejas que un sistema debe implementar. Las características de esta técnica son:

Condiciones	Reglas							
Condición 1	S	S	S	S	N	N	N	N
Condición 2	S	S	N	N	S	S	N	N
Condición 3	S	N	S	N	S	N	S	N
Acción 1	X	X						
Acción 2				X		X		X
Acción 3			X				X	
Acción 4					X			

- Se deben identificar las condiciones (entradas) y las acciones resultantes (salidas). Estas conforman las filas de la tabla.
- Las columnas de la tabla corresponden a reglas de decisión. Cada columna forma una combinación única de condiciones y la ejecución de acciones asociadas a esa regla.
- Los valores de las condiciones y acciones pueden ser valores booleanos, discretos, numéricos o intervalos de números.
- Ayuda a identificar todas las combinaciones importantes de condiciones y a encontrar cualquier desfase en los requisitos.
- La cobertura se mide de la siguiente manera:

$$\text{Cobertura} = \frac{\text{Número de reglas de decisión probadas}}{\text{Número de reglas de decisión totales}}$$

Transición de estados

Un diagrama de transición de estado muestra los posibles estados del software, así como la forma en que el software entra, sale y realiza las transiciones entre estados. Las características de esta técnica son:

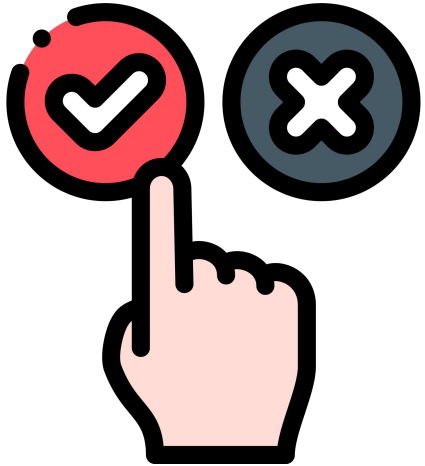
- Una tabla de transición de estado muestra todas las transiciones válidas y las transiciones potencialmente inválidas entre estados, así como los eventos, las condiciones de guarda y las acciones resultantes para las transiciones válidas.
- Los diagramas de transición de estado, normalmente, sólo muestran las transiciones válidas y excluyen las transiciones no válidas.
- La prueba de transición de estado se utiliza para aplicaciones basadas en menús y es extensamente utilizada en la industria del software embebido. La técnica también es adecuada para modelar un escenario de negocio con estados específicos o para probar la navegación en pantalla.
- La cobertura se mide de la siguiente manera:

$$\text{Cobertura} = \frac{\text{número de estados o transiciones identificados probados}}{\text{número total de estados o transiciones identificados en el objeto de prueba}}$$

3 Técnicas basadas en la experiencia

Predicción de errores

Esta técnica se utiliza para anticipar la ocurrencia de equivocaciones, defectos y fallos basados en el conocimiento del probador.

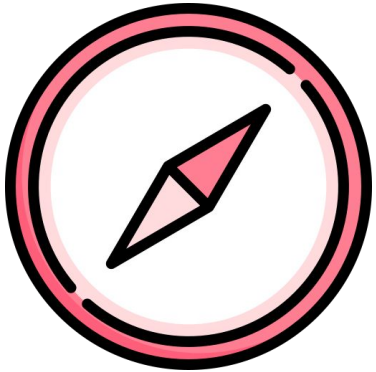


Se crea una lista teniendo en cuenta:

- Cómo ha funcionado la aplicación en el pasado.
- Equivocaciones comunes en los desarrolladores.
- Fallos en aplicaciones relacionadas.

En base a esa lista se diseñan pruebas que expongan esos fallos y defectos.

Prueba exploratoria



En esta técnica se **diseñan, ejecutan, registran** y evalúan de forma dinámica pruebas informales durante la ejecución de la prueba.

Los resultados de estas pruebas se utilizan para aprender más sobre el funcionamiento del componente o sistema.

Generalmente se utilizan para complementar otras técnicas formales o cuando las especificaciones son escasas, inadecuadas o con restricciones de tiempo.

Prueba basada en listas de comprobación

En esta técnica se **diseñan, implementan y ejecutan** casos de prueba que cubren las condiciones que se encuentran en una lista de comprobación definida.

Se crean basadas en la experiencia y conocimiento de lo que el probador cree que es importante para el usuario y se utilizan debido a la falta de casos de prueba detallados.

Durante la ejecución puede haber cierta variabilidad, dependiendo de quién ejecuta la prueba y condiciones del contexto. Esto da lugar a una mayor cobertura.

Se utiliza tanto en pruebas funcionales como no funcionales.


DigitalHouse>
Coding School