



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Ingeniería en Tecnologías de la Información



CONTROL DE CALIDAD DEL SOFTWARE

Técnicas del diseño de pruebas

ALUMNO:

Abraham Flores Basilio

ÍNDICE

Introducción.....	3
Técnica del diseño de pruebas.....	3
Pruebas de Caja Blanca o Estructurales.....	4
Pruebas de Caja Negra o Funcionales.....	5
Bibliografía.....	5

Introducción

En lo que concierne a las técnicas de pruebas existen dos grandes grupos o categorías, pruebas de caja negra y caja blanca.

Las técnicas enmarcadas dentro de la categoría de caja negra son pruebas basadas en requisitos, y como su nombre lo indica, son tratadas como una caja negra ya que se desconoce el trabajo interno del proceso que se está probando.

Su funcionamiento se basa en darle un estímulo al sistema y esperar que el resultado de ese estímulo sea aceptado de acuerdo con ciertos criterios, dependiendo de la prueba, de lo contrario la prueba será rechazada.

A diferencia de las pruebas de caja negras, en las de caja blanca se tienen tanto entradas como salidas, pero en este caso si se tiene en cuenta el desarrollo del proceso interno para determinar la aceptación de la prueba.

Todo desarrollo de software es necesario realizarle pruebas de calidad que aseguren el correcto funcionamiento de la aplicación en todos sus aspectos, de ahí, la importancia que adquiere el testing de software dentro del ciclo de vida del desarrollo.

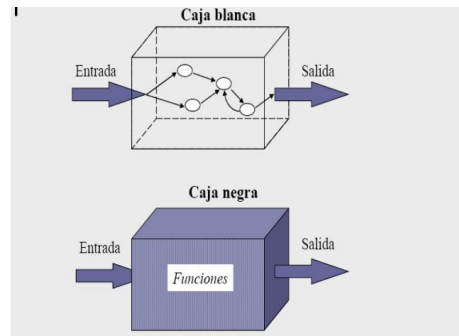
El Software Testing o Prueba de Software es una metodología de desarrollo de software que se basa en la investigación de un modelo para entregar un informe sobre la calidad de este a las partes interesadas. De esa manera, se garantiza que el software cumpla con los requisitos, que no contiene errores graves y que todo funciona como se esperaba.

Técnica del diseño de pruebas

Las técnicas de evaluación dinámica o prueba proporcionan distintos criterios para generar casos de prueba que provoquen fallos en los programas. Estas técnicas se agrupan en:

- Técnicas de caja blanca o estructurales, que se basan en un minucioso examen de los detalles procedimentales del código a evaluar, por lo que es necesario conocer la lógica del programa.
- Técnicas de caja negra o funcionales, que realizan pruebas sobre la interfaz del programa a probar, entendiendo por interfaz las entradas y salidas de dicho programa. No es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar.

Las pruebas de caja blanca necesitan conocer los detalles procedimentales del código, mientras que las de caja negra únicamente necesitan saber el objetivo o funcionalidad que el código ha de proporcionar.



Pruebas de Caja Blanca o Estructurales

A este tipo de técnicas se le conoce también como Técnicas de Caja Transparente o de Cristal. Este método se centra en cómo diseñar los casos de prueba atendiendo al comportamiento interno y la estructura del programa. Se examina así la lógica interna del programa sin considerar los aspectos de rendimiento.

El objetivo de la técnica es diseñar casos de prueba para que se ejecuten, al menos una vez, todas las sentencias del programa, y todas las condiciones tanto en su vertiente verdadera como falsa.

Puede ser impracticable realizar una prueba exhaustiva de todos los caminos de un programa. Por ello se han definido distintos criterios de cobertura lógica, que permiten decidir qué sentencias o caminos se deben examinar con los casos de prueba. Estos criterios son:

- Cobertura de Sentencias: Se escriben casos de prueba suficientes para que cada sentencia en el programa se ejecute, al menos, una vez.
- Cobertura de Decisión: Se escriben casos de prueba suficientes para que cada decisión en el programa se ejecute una vez con resultado verdadero y otra con el falso.
- Cobertura de Condiciones: Se escriben casos de prueba suficientes para que cada condición en una decisión tenga una vez resultado verdadero y otra falso.
- Cobertura Decisión/Condición: Se escriben casos de prueba suficientes para que cada condición en una decisión tome todas las posibles salidas, al menos una vez, y cada decisión tome todas las posibles salidas, al menos una vez.
- Cobertura de Condición Múltiple: Se escriben casos de prueba suficientes para que todas las combinaciones posibles de resultados de cada condición se invoquen al menos una vez.
- Cobertura de Caminos: Se escriben casos de prueba suficientes para que se ejecuten todos los caminos de un programa. Entendiendo camino como una secuencia de sentencias encadenadas desde la entrada del programa hasta su salida.

Pruebas de Caja Negra o Funcionales

También conocidas como Pruebas de Comportamiento, estas pruebas se basan en la especificación del programa o componente a ser probado para elaborar los casos de prueba. El componente se ve como una Caja Negra cuyo comportamiento sólo puede ser determinado estudiando sus entradas y las salidas obtenidas a partir de ellas. No obstante, como el estudio de todas las posibles entradas y salidas de un programa sería impracticable se selecciona un conjunto de ellas sobre las que se realizan las pruebas. Para seleccionar el conjunto de entradas y salidas sobre las que trabajar, hay que tener en cuenta que en todo programa existe un conjunto de entradas que causan un comportamiento erróneo en nuestro sistema, y como consecuencia producen una serie de salidas que revelan la presencia de defectos. Entonces, dado que la prueba exhaustiva es imposible, el objetivo final es pues, encontrar una serie de datos de entrada cuya probabilidad de pertenecer al conjunto de entradas que causan dicho comportamiento erróneo sea lo más alto posible.

Al igual que ocurría con las técnicas de Caja Blanca, para confeccionar los casos de prueba de Caja Negra existen distintos criterios. Algunos de ellos son:

- Particiones de Equivalencia: Particiones de Equivalencia La partición de equivalencia es un método de prueba de Caja Negra que divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. La partición equivalente se dirige a una definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar.
- Análisis de Valores Límite: Particiones de Equivalencia La partición de equivalencia es un método de prueba de Caja Negra que divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. La partición equivalente se dirige a una definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar.
- Métodos Basados en Grafos.
- Pruebas de Comparación.
- Análisis Causa-Efecto.

Bibliografía

Vargas, G. G. (2021, 1 marzo). Técnicas de diseño de pruebas - Categorías de las técnicas de diseño de prueba (K2 - entender, explicar, razonar). AFB. <http://scrum-qa.blogspot.com/2013/04/tecnicas-de-diseno-de-pruebas.html>

Software Testing: Tipos y Técnicas. (2015, 13 diciembre). AFB. <https://www.tecnologias-informacion.com/software-testing.html>

Binder, R., Testing Object-Oriented Systems, American Programmer, 7(4), 22-29, 1994.