

Desarrollo de Web Api's con .NET

Pruebas automatizadas



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

**Centro de
e-Learning**

Un poco de repaso...

- ¿Qué es una prueba automatizada? ¿Para qué sirve?
- ¿Qué son el SUT, DOC y Test Case?
- ¿Qué tienen de diferencia/similar una prueba de integración y una prueba unitaria?
- ¿A qué nos referimos cuando hablamos de cobertura de código?

¿Qué vamos a ver hoy?

- Pruebas de caja negra y caja blanca
- Clases de equivalencia

Pruebas de caja negra

Verifican si se obtiene la salida deseada a una entrada dada.

Es el método en el cual el elemento es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno.

Realizadas predominantemente para pruebas funcionales y no funcionales de unidad, componentes, integración y sistema.

Pruebas de caja blanca

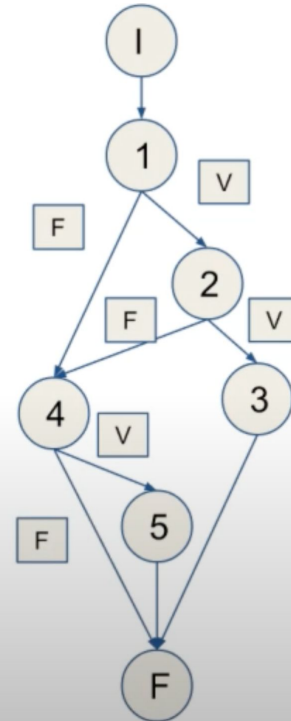
Verifican cómo el sistema se ejecuta internamente, cómo está construido y sus soluciones estructurales garantizando que todos los **caminos** independientes dentro de un componente se han ejercitado al menos una vez.

Mira el código y la estructura del producto que se va a probar y usa ese conocimiento para la realización de las pruebas.

Son muy costosas y frágiles dependiendo de la implementación.

Pruebas de caja blanca

```
public int aMethod(  
    boolean a, boolean b,  
    boolean c) {  
    int ret = 0;  
    1      2  
    if (a && b) {  
        3 ret = 1;  
    4    } else if (c) {  
        5 ret = 2;  
    }  
    F return ret;  
}
```



Clases de equivalencia

La división en particiones de clases de equivalencia es una técnica de pruebas de **caja negra** en la que las entradas del programa se clasifican en clases.

Considera como perteneciente al dominio de entrada a todas las entradas que puedan tener efecto en la salida de la prueba, positivo o negativo.

Cada clase se llama clase de equivalencia porque todos elementos en la clase prueban la misma cosa.

Si uno de los elementos de una clase provoca un error, todos los de esa clase probablemente lo harán.

La idea principal de esta técnica es identificar los casos de prueba usando un elemento de cada clase de equivalencia.

Definiendo nuestras clases de equivalencia

- Rango
- Valor específico
- Perteneciente a un conjunto
- Valor lógico

Equivalencia por **rango**

Si una entrada está condicionada a un rango de valores (por ejemplo, edades comprendidas entre 18 y 65 años ambos inclusive).

Se define una clase de equivalencia válida para todos los valores pertenecientes al rango y dos no válidas, una para los valores menores al límite inferior del rango y otra para los mayores al límite superior.

Equivalencia por **valor específico**

Similar a la clase de equivalencia por **rango**, solo que el rango quedaría restringido a un único valor (la edad es de 18 años).

Se siguen creando igualmente 1 clase de equivalencia válida y dos no válidas.

Equivalencia perteneciente a un conjunto

Cuando un valor es válido si pertenece a un conjunto (días de la semana en minúscula: «lunes», «martes»...) se define una clase de equivalencia válida para cada uno de los valores del conjunto, además de una no válida, para aquellos valores que no pertenecen a él.

Equivalencia por **valor lógico**

Se corresponde a la evaluación de una condición (por ejemplo, la edad debe ser mayor o igual a 18).

Se establece una clase de equivalencia válida (se cumple la condición) y otra no válida (no se cumple).

¡¡A trabajar!!

