

1. frameworks xUnit

1.1. La necesidad de herramientas

Definición 1.1. (framework) Un entorno de trabajo (inglés framework) o marco de trabajo es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

1.2. Primer intento: SUnit

1. Para escribir una clase de pruebas, debemos hacer que la misma derive de la clase TestCase.
2. Cada método de prueba debe poder tener cualquier nombre, a condición de que empiece con la palabra test.
3. Al escribir cada método, nos podemos ayudar con el comportamiento provisto en TestCase, que incluye algunos métodos de nombres *assert*, *deny*, *should*, *etc.*
4. SUnit se ocupa de que todos los métodos test escritos hasta el momento en una clase derivada de TestCase se puedan ejecutar a la vez.
5. Al correr un conjunto de pruebas en SUnit, una interfaz gráfica nos indica con color verde o rojo qué pruebas pasaron y cuáles no.

Definición 1.2. (Digresión: SUnit es un framework. Case) La diferencia entre biblioteca y framework es muy sutil para muchos profesionales. Sin embargo, una biblioteca brinda servicios a través de clases y funciones listas para usar, mientras que un framework brinda servicios a través de comportamientos por defecto que se pueden cambiar o ampliar a través de clases provistas por el programador.

Dicho de otra manera, una biblioteca está lista para ser usada mediante la invocación de sus servicios, mientras que un framework es un programa que invoca a los comportamientos definidos por el programador.

1.3. Herramientas de cobertura

Herramienta que mostrase si el código era cubierto o no por las pruebas