



Tecnólogo en implementación y gestión de bases de datos

# Green Bar Patterns

## Green Bar Patterns

### 1 Fake Its

Implementar constantes y poco a poco se van modificando por variables, con esta serie de pasos se logra poner todo el test en verde.

Ejemplo:

#### Figura 1

Implementación de constantes

```
expect(suma(1,2)).toEqual(3);
function suma(a,b) {return 3}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

Después de tener en verde el test, se pasa a refactorizar el código, en este caso en particular se procede a remover la duplicación.

#### Figura 2

Remover duplicación

```
expect(suma(1,2)).toEqual(3);
function suma(a,b) {return 1+2}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

Ahora se puede observar mejor la duplicación, y se procede a modificar.

#### Figura 3

Limpiar código

```
expect(suma(1,2)).toEqual(3);
function suma(a,b) {return a+b}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

## 2. Obvious Implementation

Esta técnica consiste en pensar por qué queremos probar y pensar en el caso exitoso e implementarlo. El objetivo principal de la implementación rápida es generar un test de forma rápida, se logra un salto de que el test de rojo y por el contrario obtenemos de forma rápida un verde de exitoso el paso por el test. Se debe implementar operaciones simples.

Ejemplo:

### Figura 4

Implementación rápida

```
expect(suma(1,2)).toEqual(3);
function suma(a,b) {return a+b}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

## 3. Triangular

Como dice Kent Beck en su libro *TDD By Example*, triangular es la técnica más conservadora para conseguir la implementación que se busca y no le falta razón. Después de obtener el test en verde, se va a construir otro que coloque el test en rojo

Ejemplo:

### Figura 5

*Triangular test*

```
expect(suma(1,2)).toEqual(3);
expect(suma(3,4)).toEqual(7);
function suma(a,b) {return 3}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

Ahora se dispone de dos ramas:

- Construir la implantación para obtener los dos test en verde.
- Continuar retornando constantes con una implementación más sencilla.

#### Figura 6

Refactorización

```
function suma(a,b) {if(a==1) return 3 else return 7}
```

Nota. Reproducida de Ejemplo de desarrollo aplicando TDD (parte 1). (Galindo, 2018).

Por último, se tiene implementación completa, se procede a limpiar el código eliminando esos test que sobran y no aportan valor al código.