

Orientación Objetos 2 - 2013 - Parcialito de Frameworks -

07/06/2013

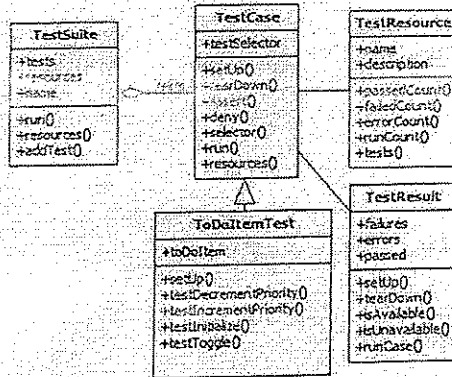
Ejercicio 1

1. ¿Para qué sirve un hotspot en un framework? De algún ejemplo de hotspot de SUnit.
2. Explique brevemente el funcionamiento del mecanismo de inversión de control en un framework. Indique un ejemplo en el framework de GUIs de VisualWorks.
3. ¿Cuál es la diferencia entre un framework de Caja Blanca y uno de Caja Negra?

Ejercicio 2

Sobre la base del framework SUnit se ha desarrollado la Aplicación ToDoItemTest. La siguiente figura muestra el diagrama de clases de ToDoItemTest y más abajo se muestra parte del código. Sus tareas consisten en:

1. identificar el o los hotspots de SUnit utilizados.
2. identificar el o los métodos donde se produce la inversión de control
3. identificar el o los métodos hook utilizados



```

Class: TestSuite
SuperClass: Object
InstanceVariables: 'tests resources name'

```

```

----- Instance -----
#run
| result |
result := TestResult new.
self resources do: [:res | res isAvailable
ifFalse: [:res signalInitializationError]].
[self run: result] sunitEnsure: [self resources
do: [:each | each reset]].

```

```

^result
=====
Class: TestCase
superclass: Object
instanceVariableNames: 'testSelector'
----- Class
#run: aSymbol
^self selector: aSymbol run

#selector: aSymbol
^self new setTestSelector: aSymbol

----- Instance -----
#run
| result |
result := TestResult new.
self run: result.
^result

#run: aResult
aResult runCase: self

#runCase
[self setUp.
self performTest] sunitEnsure: [self tearDown]
#setUp
^self

#tearDown
^self

#performTest
self perform: testSelector sunitAsSymbol.
^self
=====
Class: ToDoItemTest
superclass: TestCase
instanceVariableNames: 'ToDoItem'

```

```

----- Instance -----
#setUp
ToDoItem := ToDoItem new.

#testDecrementPriority
self assert: ToDoItem priority = 1.
ToDoItem decrementPriority.
self assert: ToDoItem priority = 1.
10 timesRepeat: [ToDoItem incrementPriority].
self assert: ToDoItem priority = 10.

```

```

ToDoItem decrementPriority.
self assert: ToDoItem priority = 9.

#testIncrementPriority
self assert: ToDoItem priority = 1.
ToDoItem incrementPriority.
self assert: ToDoItem priority = 2.
10 timesRepeat: [ToDoItem incrementPriority].
self assert: ToDoItem priority = 10.

#testInitialize
self assert: ToDoItem text = ''.
self assert: ToDoItem priority = 1.
self assert: ToDoItem isCompleted = false.

#testToggle
ToDoItem toggle.
self assert: ToDoItem isCompleted = true.
ToDoItem toggle.
self assert: ToDoItem isCompleted = false.

=====
Class: TestResult
Superclass: Object
instanceVariableNames: 'failures errors passed'
----- Instance -----
runCase: aTestCase
[aTestCase runCase.
self addPass: aTestCase]
sunitOn: self class failure , self class error
do: [:ex | ex sunitAnnounce: aTestCase
toResult: self].

addPass: aTestCase
^self passed add: aTestCase

```