Applying TDD to Android Development

School: test-driven development TDD

Material didáctico para **Scrum masters,**

seminario, 6h: dirigido a coders Android

www.ataraxia_archive.net

Preliminar knowledges

Test-Driven Development (TDD) is a cornerstone practice for agile development teams. TDD provides the benefit of verifying that your code is correct up front while forcing you to use your own APIs. This leads to better design and fewer bugs in the software. In addition, because unit tests can be run in an automated fashion through continuous integration, regression defects are caught immediately. With Android development now being performed in the corporate space by large teams, agile practices are a necessity. Yet, Android development poses some significant challenges when it comes to following a TDD approach.

ΟTags

Agile, java, mobile, test automation, test-driven development, android

II.Documentación referencia

Como hacemos Pilas de Producto.

2016-01-16 14:12 Tradución (http://www.proyectalis.com/) del libro "Scrum and XP from the Trenches" de H.K.. http://www.infoq.com/minibooks/scrum-xp-from-the-trenches

Read Online

> La pila de producto es el corazón de Scrum. Es donde empieza todo. La Pila de Producto es, básicamente, una lista priorizada de requisitos, o historias, o funcionalidades, o lo que sea. Cosas que el cliente quiere, descritas usando la terminología del cliente. Llamamos a esto *historias*, o a veces simplemente *elementos de la Pila*.

. Developing Android unit and instrumentation tests - Tutorial.

2016-01-07 15:16 Lars Vogel http://www.vogella.com/tutorials/AndroidTesting/article.html

Mock object.

2016-01-07 15:18 From Wikipedia, the free encyclopedia:

> In object-oriented programming, **mock objects** are simulated objects that mimic the behavior of real objects in controlled ways. A programmer typically creates a mock object to test the behavior of some other object, in much the same way that a car designer uses a crash test dummy to simulate the dynamic behavior of a human in vehicle impacts.

. *The Robolectric framework & TDD.*

> Robolectric is a framework for unit testing with Android that provides the best of both worlds. The framework intercepts calls to Android classes through "shadow classes," providing testable behavior for a significant portion of the Android SDK.

Program

.*Bloque apertura* *(9:00 - 10:45)*

- Ubicación de TDD en el marco Agile Scrum y la *Pila de producto*.
- Android: we want tests that can be developed and executed efficiently.

.*Bloque mañana* *(11:00 - 12:30)*

- *Mock objects challenging*. Wikipedia provides great a definition of mock objects.
- *Emulation is slow and impractical for a large number of tests.* An additional option is the development of tests that run within the Android emulator. The Android SDK provides library components for achieving this, but the test execution is slow.

.*Bloque tarde* *(15:45 - 17:00)*

- Caso práctico: Ataraxia Archive Android PeopleApp: Cierre módulo autentificación. Generar informe de tests usando el framework Robolectric. An alternate approach to Robolectric is to use mock frameworks such as Mockito or to mock out the Android SDK.

Subtareas

#4: Verificación secuencia de arranque.
#5: Verificación activación de servicios
#6: Verificación desconexión