

# ¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.  
Ese apoyo que siempre quiso tener...

## 1. Desarrollo de componentes y proyectos a medida



## 2. Auditoría de código y recomendaciones de mejora

## 3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



## 4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,  
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)  
Gestor de contenidos (Alfresco)  
Aplicaciones híbridas

Tareas programadas (Quartz)  
Gestor documental (Alfresco)  
Inversión de control (Spring)

Control de autenticación y  
acceso (Spring Security)  
UDDI  
Web Services  
Rest Services  
Social SSO  
SSO (Cas)

JPA-Hibernate, MyBatis  
Motor de búsqueda empresarial (Solr)  
ETL (Talend)

Dirección de Proyectos Informáticos.  
Metodologías ágiles  
Patrones de diseño  
TDD

BPM (jBPM o Bonita)  
Generación de informes (JasperReport)  
ESB (Open ESB)

AdictosAlTrabajo



Entra en Adictos a través de



E-mail

Contraseña

Entrar

[Deseo registrarme](#)  
[Olvidé mi contraseña](#)
[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestro libro](#) [Más](#)
» Estás en: [Inicio](#) [Tutoriales](#) [Robolectric: aplicando TDD en Android](#)

Rubén Aguilera Díaz-Heredero

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática, especialidad en Ingeniería del Software

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Ver todos los tutoriales del autor](#)

Tu éxito en la web

Controla aquí si tu nombre de dominio .eu sigue disponible



Fecha de publicación del tutorial: 2012-10-30

Tutorial visitado 1 veces [Descargar en PDF](#)

## Robolectric: aplicando TDD en Android

### 0. Índice de contenidos.

- 1. Entorno
- 2. Introducción
- 3. Vamos a probarlo
- 4. Conclusiones

### 1. Entorno

Este tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Mac Book Pro 17" (2,6 Ghz Intel Core i7, 8 GB DDR3)
- Sistema Operativo: Mac OS X Snow Leopard 10.6.4
- Robolectric 1.0

### 2. Introducción

**Robolectric** es un framework que simula la interfaz gráfica de un emulador de Android permitiéndonos ejecutar test en la propia máquina virtual de Java y ahorrándonos el tiempo de arranque del emulador cada vez que queremos ejecutar algún test. Este framework también nos ayuda a poder implementar TDD en proyectos de Android.

### 3. Vamos a probarlo

Para probar el framework nos vamos a basar en el proyecto maven de Android que ya creamos en este [tutorial](#). Para poder utilizar Robolectric lo único que tenemos que hacer es añadir la siguiente dependencia a nuestro pom.xml:

```

view plain print ?
01. <dependency>
02.   <groupId>com.pivotallabs</groupId>
03.   <artifactId>robolectric</artifactId>
04.   <version>1.0</version>
05.   <scope>test</scope>
06. </dependency>

```

Ahora creamos la clase de test con el siguiente código:

```

view plain print ?
01. package com.autentia;
02.
03.
04. import org.junit.Assert;
05. import org.junit.Test;
06. import org.junit.runner.RunWith;
07.
08. import android.widget.Button;
09. import android.widget.EditText;
10.
11. import com.xtremelabs.robolectric.RobolectricTestRunner;
12.
13. @RunWith(RobolectricTestRunner.class)
14. public class CalcActivityRobolectricTest {
15.
16.     private CalcActivity activity;
17.     private Button btnResult;
18.     private EditText strOp1;
19.     private EditText strOp2;
20.     private EditText strOperation;
21.     private EditText strResult;
22.

```

### Catálogo de servicios Autentia



### Síguenos a través de:



### Últimas Noticias

» Autentia estuvo en el Duatlón de Torrejón de Ardoz

» Participamos en la Carrera de las Empresas 2012

» ¡¡¡Terrakas 1x04 recién salido del horno!!!

» Estreno Terrakas 1x04: "Terraka por un día"

» Nuevos cursos de gestión de la configuración en IOS y Android

[Histórico de noticias](#)

### Últimos Tutoriales

» [WordPress vs. Joomla](#)» [Tu aplicación web ZK enfoque MVVM \(5-5\)](#)» [Tu aplicación web ZK enfoque MVC \(4-5\)](#)» [Añadiendo reconocimiento de gestos a nuestra aplicación desde StoryBoard](#)» [JRBeanCollectionDataSource: trabajando con arrays multidimensionales en Jasper Report.](#)

» Trabajando en Android con Maven

» Sonar Runner: Analizar proyectos sin Maven en cualquier lenguaje

» Desplegando una aplicación en Cloud Foundry con Maven

» Database MessageSource: obtener los literales de una base de datos

» Primeros pasos con Cloud Foundry

## Últimas ofertas de empleo

2011-09-08

 Comercial - Ventas - MADRID.

2011-09-03

 Comercial - Ventas - VALENCIA.

2011-08-19

 Comercial - Compras - ALICANTE.

2011-07-12

 Otras Sin catalogar - MADRID.

2011-07-06

 Otras Sin catalogar - LUGO.

```

23.     @Test
24.     public void shouldAddTwoNumbers() throws Exception{
25.         activity = new CalcActivity();
26.         activity.onCreate(null);
27.         btnResult = (Button) activity.findViewById(R.id.btnResult);
28.         strOp1 = (EditText) activity.findViewById(R.id.strOp1);
29.         strOp1.setText("3.0");
30.         strOp2 = (EditText) activity.findViewById(R.id.strOp2);
31.         strOp2.setText("4.0");
32.         strOperation = (EditText) activity.findViewById(R.id.strOperation);
33.         strOperation.setText("1");
34.         strResult = (EditText) activity.findViewById(R.id.strResult);
35.         btnResult.performClick();
36.         String result = strResult.getText().toString();
37.         Assert.assertEquals(result, "7.0");
38.     }
39.
40. }
```

Como se puede apreciar es un test normal de Junit 4, lo único que lo diferencia es la anotación `@RunWith(RobolectricTestRunner.class)`. No tenemos que aprender a utilizar un nuevo framework de pruebas porque nos permite implementar nuestro test con código propio de Android.

Ahora si ejecutamos `mvn test` en nuestro terminal veremos como el test pasa correctamente sin necesidad de levantar ningún emulador, con lo que se ahorra mucho tiempo de ejecución de los test.

Otra de las ventajas que presenta este framework es que nos permite aplicar TDD a nuestros desarrollos con Android, dado que podemos implementar casos de uso que interactuen con la interfaz gráfica sin que nuestros test presente ningún problema de ejecución.

También nos permite ejecutar los test sin necesidad de definir el `InstrumentationTestRunner` en el archivo `AndroidManifest` ni tener que utilizar la librería de test de Android que solo funciona con un dispositivo levantado.

#### 4. Conclusiones

Como hemos visto Robolectric es una alternativa muy a tener en cuenta a la hora de testear nuestros desarrollos y poder aplicar los principios de TDD a nuestros desarrollos con Android.

Cualquier duda o sugerencia en la zona de comentarios.

Saludos.

### A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

### Por favor, vota +1 o compártelo si te pareció interesante

[Share](#) |

0

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

» **Regístrate** y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

IMPULSA

Impulsores

Comunidad

¿Ayuda?

0 personas han traído clicks a esta página

sin clicks + + + + + + + +

powered by [karmacracy](#)