

Java

2014 - Beatriz

ACCIONES

Clonar

Crear branch

Crear pull request

Comparar

Derivación

NAVIGATION

Resumen

Fuente

Commits

Ramas

Pull requests

Pipelines **NEW**

Incidencias 1

Wiki

TFG /

Untitled project /

2014 - Beatriz /

Incidencias

Incidencias

+ Crear incidencia

Abrir

Flujo de trabajo

Adjuntar

Editar

Incidencia #16 **RESOLVED**

Despliegue: Ejecutable - Pruebas - calidad

Beatriz Zurera

creada una incidencia 2015-01-19

Crear el ejecutable con el ANT.

Comentarios (18)

Beatriz Zurera

PONENTE

status cambiado a [open](#)

Ya se puede utilizar el *ant* correctamente. Yo lo he probado en varias ocasiones en lugares diferentes y me ha funcionado. Si tienes cualquier problema me dices.

Sincronízate con el repositorio.

Ejecuta los siguientes comandos. (Te recomiendo que uses el *clean* para eliminar todas las carpetas que genera el ANT)

ant clean

ant [runnable]

En la carpeta *runnable* ejecuta el siguiente comando:

java -jar SistInfGenWeb.jar [nivel]

2015-01-19

Carlos López

La tarea javadoc tiene errores Añadir una taraea de test JUnit/DBUnit para probar La Fachada de acceso a datos. Incorporar la cobertura de pruebas (SonarCube o Eclemma)

Editar

Eliminar

2015-01-20

Carlos López

title cambiado a [Despliegue: Ejecutable - Pruebas - calidad](#)

2015-01-20

Carlos López

Referencia para pruebas de base de datos DBUnit <http://dbunit.sourceforge.net/intro.html> y curso de udemy. <https://www.udemy.com/curso-especialista-java-pruebas-automaticas-en-java/?dtcode=aVHqjK62fugR>

Editar

Eliminar

2015-01-20

Beatriz Zurera

PONENTE

He implementado un primer test para la clase SistInfData.java y he obtenido casi un 90% de cobertura en esa clase.

Asignado a

Beatriz Zurera

Tipo

task

Prioridad

critical

Estado

resolved

Votos

0 [Vota por esta incide...](#)

Observadores

2 [No seguir observando](#)

JIRA Software

the preferred issue tracker for Bitbucket.

Join the team!

7

1 de 6

15/11/2016 10:07

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
22. SistInfData_138.java	11.5 %	493	3,759	4,252
23. myMain	11.5 %	493	3,759	4,252
24. myMainLoaded	11.5 %	493	3,759	4,252
25. DiscMain.java	0.0 %	0	2,259	2,259
26. DiscMain.java	0.0 %	0	1,875	1,875
27. DiscMain.java	0.0 %	0	863	863
28. DiscMain.java	0.0 %	0	457	457
29. DiscMain.java	0.0 %	0	347	347
30. DiscMain.java	0.0 %	0	294	294
31. DiscMain.java	0.0 %	0	275	275
32. DiscMain.java	0.0 %	0	237	237
33. DiscMain.java	0.0 %	0	227	227
34. DiscMain.java	88.2 %	638	122	760
35. DiscMain.java	88.2 %	39	12	51

Pero he estado intentando crear una tarea en ANT con JaCoCo, pero me es imposible porque no consigo acceder al archivo de configuración que creamos, por lo que no se pueden leer las diferentes tablas para hacer las pruebas.

2015-01-23



Carlos López

No sé si entiendo el problema. Los test tienen que ser independientes de la aplicación. Por tanto la ejecución de los test tendrá sus propios recursos: bases de datos, archivos de configuración, etc Si nos vas a probar la la clase que trabaje con con los parámetros de configuración puedes hacer una clase Mock que te devuelva parámetros de configuración que necesitas.

Editar • Eliminar • 2015-01-26



Beatriz Zurera **PONENTE**

Intentaré hacer eso porque todas las clases usan el archivo de configuración, que es el que dice dónde está la carpeta de salida y dónde se encuentran los datos de entrada.

2015-01-26



Beatriz Zurera **PONENTE**

No consigo hacerlo funcionar, y creo que no he entendido bien el concepto de Test. En la clase SistInfData.java tengo esto:

```
private static final String DIRCSV = Get
```

Esa clase GetProperties es tal que así:

```
public final class GetProperties {

    private static final Logger LOGGER =
    private static final Properties prop
    static {
        try {
            File file = new File("../s
            FileInputStream fileInput =
            properties.load(fileInput);
            fileInput.close();
        } catch (IOException e) {
            LOGGER.error("El directorio
            throw new ExceptionInInitial
        }
    }

    public static String getSetting(Stri
    return properties.getProperty(ke
    }
}
```

Y aquí es donde me salta la IOException, porque no encuentra dicho fichero. Ahora mismo el test que se está realizando es sobre los datos que usa la aplicación(en la carpeta rsc/data), y por lo que me has comentado que de que tiene que ser independiente me temo que está mal realizado.

2015-01-26



Carlos López

La propiedad

```
dataIn
```

que defines en

```
SistInfData
```

debería ser diferente para las clases de test que para las clases de la aplicación. Necesitas crear un mock de la clase `GetProperties` para que cargue las propiedades de los test. Es decir sustituir

```
File file = new File("../src/config.p
```

con las nuevas propiedades definidas en

```
File file = new File("../test/config.
```

Editar • Eliminar • 2015-01-26



Beatriz Zurera PONENTE

No es que lo haya abandonado esto, pero después de 2 días sigo sin avanzar con este tema. No acabo de entender las clases `Mock` y no consigo sustituir el path del archivo de configuración.

2015-01-28



Carlos López

Perdona igual te he liado con la nomenclatura, lo que te quería indicar es que puedes hacer un `Stub` de la clase que llamas

```
GetProperties
```

para que devolviera el valor de la propiedad que necesites para probar en el test. Échate un vistazo al tema de pruebas de integración en el curso de Udemty que te pase.

Una cuestión menor de diseño, nunca nombres una clase con un verbo

```
GetProperties
```

vs

```
PropertiesApp
```

Mañana te puedo atender de 11:00 a 13:30 en mi despacho EPS A1

Editar • Eliminar • 2015-01-29



Beatriz Zurera PONENTE

He seguido paso a paso lo que va haciendo Raúl en el curso de pruebas automáticas. Y después de dar muchas vueltas y según entendí lo que me comentaste el viernes, esto es lo que me ha pasado:

```
@Test
public void testAvg() throws Excepti

    PropertiesApp prop = mock(Proper

    prop = PropertiesApp.getInstance

    when(prop.getSetting("dataIn")).
        sistInfData = new SistInfData();

    Number esperado = sistInfData.ge
    assertThat(esperado, is((Number)
    }
```

Al ejecutar esto, sale la siguiente traza de error:

```
1) testAvg(ubu.digit.pesistence.SistInfD
org.mockito.exceptions.misusing.MissingM
when() requires an argument which has to
For example:
    when(mock.getArticles()).thenReturn(
```

Also, this error might show up because:

1. you stub either of: **final/private/equ**
Those methods **cannot** be stubbed/ver
2. inside **when()** you don't call method o
3. the parent of the mocked **class is** not
It is a limitation of the mock engine

Obviamente esto se debe a que getInstance es un método static, por lo que cambié el código por este:

```
@Test
public void testAvg() throws Excepti

    mockStatic(PropertiesApp.class);

    String esp = "../src/rsc-test"

    PropertiesApp prop = PropertiesA
        .getInstance("../src/m

    when(prop.getSetting("dataIn")).
        sistInfData = new SistInfData();

    Number esperado = sistInfData.ge
    assertThat(esperado, is((Number)
    }
```

Pero con esa opción la variable prop es siempre **null**, no entiendo por qué... Pero bueno, al final he conseguido resolverlo de la siguiente manera:

```
@RunWith(PowerMockRunner.class)
@PrepareForTest({ PropertiesApp.class, S
public class SistInfDataTest {

    SistInfData sistInfData;

    PropertiesApp test = PropertiesApp
        .getInstance("../src/test/

    @Test
    public void testAvg() throws Excepti

        mockStatic(PropertiesApp.class);

        when(PropertiesApp.getInstance("
            .thenReturn(test);

        sistInfData = new SistInfData();

        Number esperado = sistInfData.ge
        assertThat(esperado, is((Number)
        }
    //....
}
```

Ahora, voy a mejorarlo con un método @ Before para no tener que repetirlo en todos los métodos... Y luego espero que no me genere muchos problemas para meterlo en ANT

2015-02-02



Beatriz Zurera **PONENTE**

Acabo de subir todo lo relacionado con las

pruebas, los ficheros xml, un properties...

He conseguido que se ejecuten los test, la cobertura y JaCoCo, pero de momento no he conseguido que me muestre los datos reales... Porque siempre obtengo un cobertura de 0% en todas las clases. Tendré que darle una vuelta a eso.

2015-02-03



Carlos López

Buena trabajo Beatriz. Efectivamente tu implementación de la prueba con el stub es la solución que te comenté. Intenta documentar la prueba (Descripción, Datos de Entrada y Salida esperada) con documentación JavaDoc.

Editar • Eliminar • 2015-02-04



Beatriz Zurera PONENTE

He estado investigando el problema con la cobertura que tenía, ya que cuando lo ejecuto sobre el SistInfDataTest la cobertura siempre es 0%.

Según este [post](#) parece que al usar PowerMock Eclemma no funciona.

Por eso, voy a usar [cobertura](#)

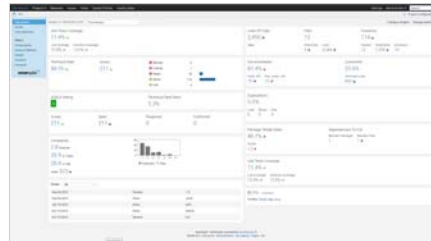


2015-02-04



Beatriz Zurera PONENTE

Acabo de conseguir que aparezca la cobertura en análisis de SonarQube:



Conseguido esto, queda descartado Eclemma por los problemas que tiene con PowerMock. En su vez estoy usando [cobertura](#), como ya he comentado.

2015-02-04



Carlos López

De nuevo un buen trabajo. Beatriz, ¿las 211 issue open se refieren a Bitbucket ? ¿Es posible integrar Sonar con BitBucket y Git?

Editar • Eliminar • 2015-02-05



Beatriz Zurera PONENTE

- status cambiado a [resolved](#)

2015-03-13



¿Qué quiere decir?

