

# ¿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**

Final de Terrakas  
¡¡Ven al estreno!!  
terrakas.com



**autentia**  
Soporte a desarrollo informático

Hosting patrocinado por  
**enredados**

Entra en Adictos a través de  

E-mail

Contraseña

Entrar [Deseo registrarme](#)  
[Olvidé mi contraseña](#)



[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestros libros](#) [Más](#)

» Estás en: [Inicio](#) [Tutoriales](#) [Integración de Jira, Redmine y Bugzilla en TestLink](#)



[Rafael Macías Rodríguez](#)

Rafael es un alumno becario de prácticas en Autentia S.L. procedente del I.E.S. Rey Fernando VI

Twitter: [Seguir a @Rafa\\_g3n](#) { 114 seguidores }



[Ver todos los tutoriales del autor](#)

## Catálogo de servicios Autentia



Fecha de publicación del tutorial: 2013-05-29

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

# Integración de Jira, Redmine y Bugzilla en TestLink

## Índice de contenidos

- 1. Entorno
- 2. Introducción
- 3. Integración de Jira en TestLink
- 4. Integración de Redmine en Testlink
- 5. Integración de Bugzilla en TestLink
- 6. Conclusión

## 1. Entorno

Este tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Intel Core 2 CPU T7200 @ 2.00GHz x 2
- Sistema Operativo: Ubuntu 13.04 x32
- Sublime Text 2
- Servidor Apache con Redmine
- TestLink 1.9.5
- Bugzilla 4.4
- Jira 5
- Mysql 5.5

## 2. Introducción

Hoy en día disponemos de múltiples herramientas para gestionar los requisitos funcionales de nuestros proyectos, podemos especificarlos en herramientas como Redmine, Jira, Bugzilla, Mantis o herramientas de administración de requisitos propiamente dichas, como Pivotal Tracker.

En otro [tutorial](#) os hablé de una herramienta para comprobar estas funcionalidades, TestLink.

El problema lo encontramos a la vez que encontramos la necesidad de saber si hemos cubierto o no todas estas funcionalidades, de eso voy a hablaros.

Dadas las integraciones que nos permite TestLink podemos vincular, enlazar, los requisitos del propio producto con los test de funcionalidad (Pruebas de aceptación) que vamos a gestionar en TestLink.

Básicamente lo que buscamos es llevar una trazabilidad entre nuestras pruebas de aceptación con los propios requisitos del proyecto, así como enlazarlos con los posibles fallos que vayamos encontrando a la hora de realizar dichas pruebas.

Y de esta manera saber si cubrimos todas las funcionalidades de nuestro proyecto.

En este tutorial vamos a ver como integrar las siguientes herramientas:

- Jira
- Redmine
- Bugzilla

TestLink proporciona integración con otras herramientas como Mantis, Trac, Eventum, gForge... pero se haría un tutorial larguísimo, y lo más seguro es que se haga de una manera bastante similar a las que vamos a ver a continuación.

Si queréis conocer más sobre las herramientas que vamos a utilizar ver los siguientes enlaces:

- Documentación oficial de TestLink sobre [integración](#) de otras herramientas, versión algo anticuada.
- Página oficial [Jira](#).



Síguenos a través de:



## Últimas Noticias

» [Atención, APLAZADO](#)  
Estreno último capítulo de Terrakas

» [Vendedor: Soy inseguro, filtra o elige por mí: si quieres que te compre.](#)

» [Comentando el libro: El arte de pensar, de Rolf Dobelli](#)

» [Ya está a la venta mi segundo libro: Planifica tu éxito, de aprendizaje a empresario](#)

» [Ya esta disponible en eBook mi primer libro: Informática Profesional](#)

[Histórico de noticias](#)

## Últimos Tutoriales

» [TDD, BDD & Test de aceptación](#)

» [Comentando el Atlassian Day organizado por Deiser](#)

» [iOS Básico, Connections IBAction IBOutlet, Delegate pattern](#)

- [Tutorial Redmine](#) de Jose Manuel Sánchez en Adictos al trabajo.
- [Tutorial Bugzilla](#) de Alejandro Pérez García en Adictos al trabajo.

### 3. Integración de Jira en TestLink

Jira es un gestor de proyectos desarrollado por Atlassian que permite a los equipos planificar, construir y finalizar grandes proyectos.

Si quieres conocer como funciona te recomiendo visitar la [página oficial](#). No es una herramienta OpenSource por lo que requiere una [suscripción](#).

Bien ahora entramos en la chicha del asunto.

Para integrar un Issue / Bug tracker en TestLink, debemos realizar una serie de pasos.

- Vamos al directorio de TestLink y realizaremos una modificación en nuestro fichero custom\_config.inc.php, añadimos las siguientes líneas:

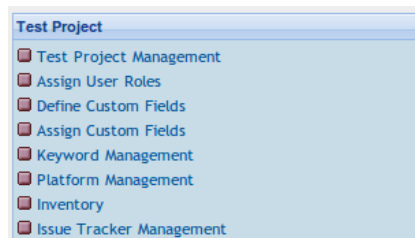
```
$g_interface_bugs = 'JIRA'; // Old value 'NO';
$g_bugInterfaceOn = true; // Old value false;
```

- Tenemos que crear un fichero tipo .cfg.php para nuestro issue tracker en el directorio Testlink/cfg, por ejemplo: "jira.cfg.php con el siguiente contenido:

```
1 <?php
2 define('BUG_TRACK_DB_HOST', 'localhost');
3
4 define('BUG_TRACK_DB_NAME', 'nombreBDJira');
5
6 define('BUG_TRACK_DB_USER', 'nombreUsuarioBDJira');
7
8 define('BUG_TRACK_DB_PASS', 'contraseñaUsuarioBDJira');
9
10 define('BUG_TRACK_HREF', "servidorJira/browse/");
11
12 define('BUG_TRACK_DB_TYPE', 'mysql'); #Tipo BD
13
14 define('BUG_TRACK_ENTER_BUG_HREF', "servidorJira/secure/CreateIssue!default.jspa"); #
15
16 ?>
```

- Reiniciamos de nuevo el servidor y vamos a TestLink.

Una vez en TestLink, en el menú principal accedemos a "Issue tracker management" o "Gestión del Gestor de Defectos":



Se nos mostrará la siguiente interfaz:

Ahí tenéis un ejemplo de como configurar una interfaz de Jira soap, en Type podéis seleccionar el tipo de Issue Tracker que váis a utilizar, y TestLink os proporciona ejemplos. Por ejemplo para Jira BD no hay implementación, aunque aparece la opción, por lo que probaremos con SOAP.

Nota: Para evitar errores a la hora de crear issues añadir la siguiente linea en configuration del Issue Tracker:

» Diseño básico para NO diseñadores.

» Iconos increíbles para nuestra web con Font Awesome Icons

### Últimos Tutoriales del Autor

» Manejo de test con TestLink

» Internacionalizar una aplicación creada con Ember

» Uso de Requirejs para modularizar una App creada con Emberjs

» Primeros pasos para conocer Emberjs

### 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

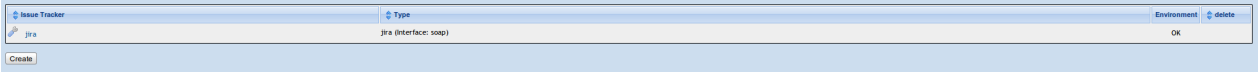
Otras Sin catalogar - MADRID.

2011-07-06

Otras Sin catalogar - LUGO.

```
1 | <projectkey>MIK</projectkey> <!-- Clave de vuestro proyecto en Jira -->
```

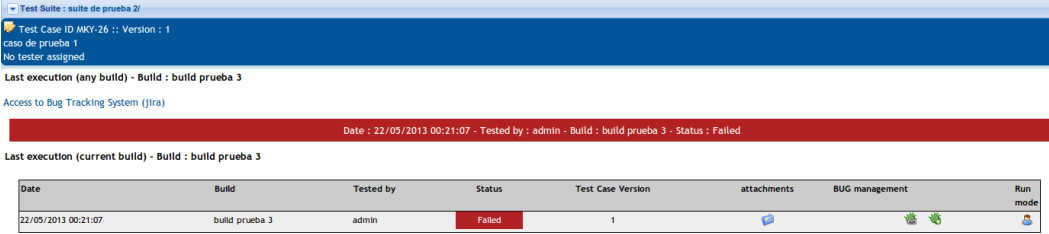
Lo comentaré más adelante.



Bien, tras crear todo esto es momento de asignar el Issue Tracker a nuestro proyecto. En el menú principal accedemos a "Project Management" o "Gestión de Proyectos de Pruebas".

Al crear un nuevo proyecto o editar uno ya creado, en Issue Tracker o "Integración del Gestor de Defectos" chequeamos la opción de Activo y seleccionamos el gestor.

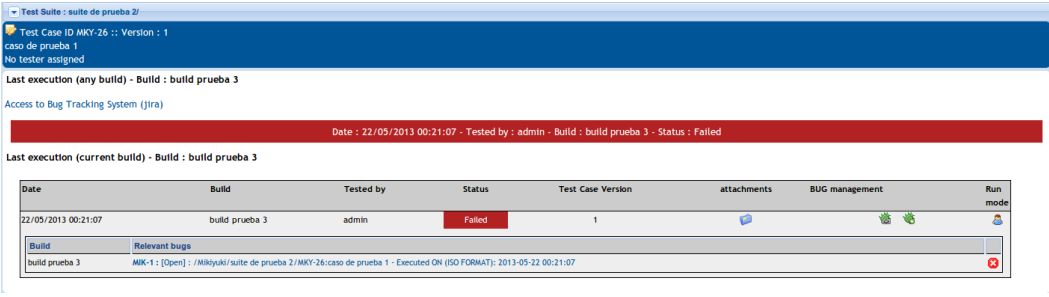
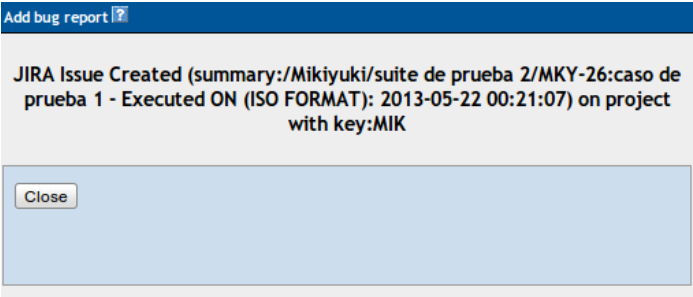
A partir de ahora, siempre que ejecutemos un test en TestLink, si no los pasamos nos aparecerá la opción de crear una Issue/Bug en nuestro tracker o vincularlo con uno ya existente:



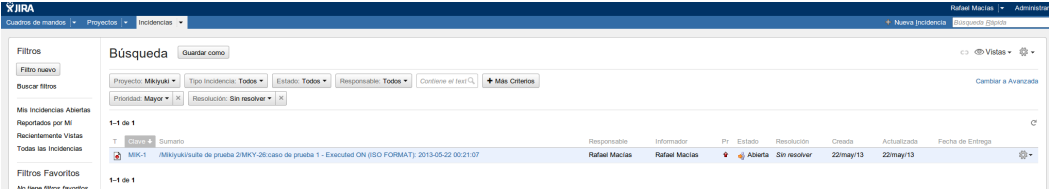
Vale, ahora paso a explicarlos el tema del error que os comenté antes. Si no añadimos en la configuración del IssueTracker la Key de nuestro proyecto en Jira, nos mostrará el siguiente error en un popup:

"Create JIRA Ticket FAILURE => com.atlassian.jira.rpc.exception.RemotePermissionException:  
The project specified does not exist or you don't have permission to create issues in it."

Sin embargo si ya lo hicimos anteriormente, la issue se creará sin ningún problema :



Comprobamos su creación en Jira:



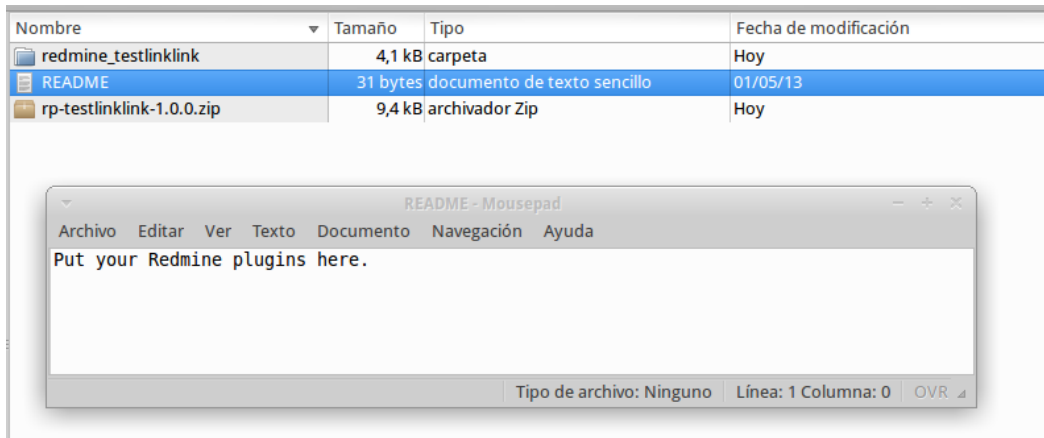
4. Integración de Redmine en Testlink

Redmine es una aplicación web muy flexible para gestionar nuestros proyectos, generar una documentación, gestionar nuestros test, crear issues... Está desarrollado sobre el framework de Ruby on Rails y es Open Source con licencia GNU. Si quieres conocer más sobre Redmine ver el tutorial que os enlacé antes o visita la página oficial.

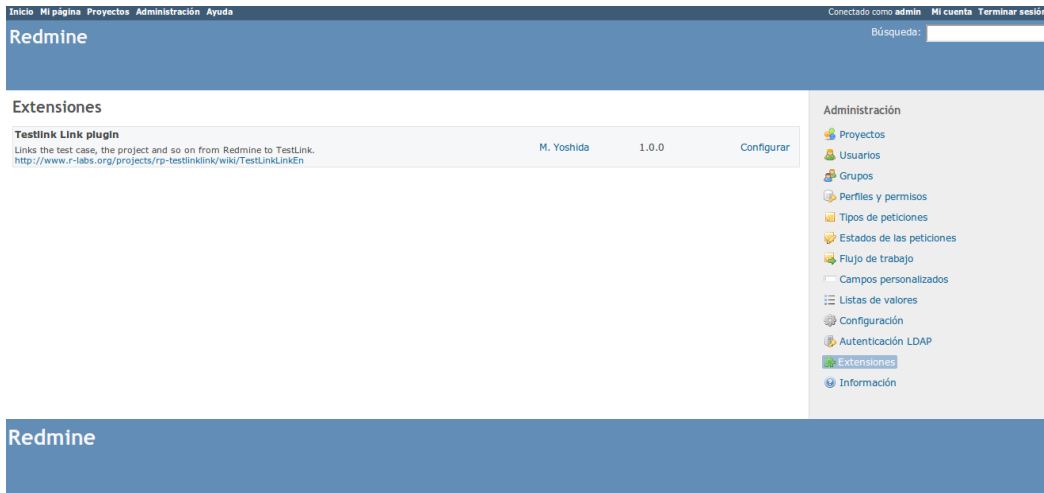
Para integrar Redmine con Testink tenemos que instalar un plugin en Redmine.

Podemos descargarlo del siguiente enlace.

Lo desempaquetamos dentro del directorio Plugins de nuestro directorio de redmine:



Una vez desempaquetado, reiniciamos nuestro servidor de Redmine, y accedemos a la lista de plugins desde Administración>Extensiones y lo configuramos a nuestro gusto.



### Configuración: Testlink Link plugin

#### TestLink Server Settings

Site Address:

Version:

#### Default Output Items Setting

**Test report (report)**

- ☐ toc : Table of Contents
- ☐ header : Test Suite description
- ☒ summary : Test Case Summary
- ☐ body : Test Case Scenario
- ☐ author : Test Case Author
- ☐ keyword : Test Case keywords
- ☐ cfields : Test Case Custom Fields
- ☒ passfail : Test results
- ☐ metrics : Metrics

**Test plan report (planreport)**

- ☐ toc : Table of Contents
- ☐ header : Test Suite description
- ☒ summary : Test Case Summary
- ☐ body : Test Case Scenario
- ☐ author : Test Case Author
- ☐ keyword : Test Case keywords
- ☐ cfields : Test Case Custom Fields
- ☐ testplan : Test Plan description

Powered by Redmine © 2006-2013 Jean-Philippe Lang

Ahora vamos a configurar nuestro proyecto en Redmine, si no lo hicimos anteriormente, para obtener :

- El identificador del proyecto, en Administración>Configuración>Proyecto chequeamos la opción "Generar Identificadores de proyecto"

Configuración

General

Mostrar

Autenticación

Proyectos

Peticiones

Notificaciones por correo

Correos entrantes

Repositorios

Los proyectos nuevos son públicos por defecto

☒

Módulos activados por defecto en proyectos nuevos

☒ Peticiones

☒ Control de tiempo

☒ Noticias

☒ Documentos

☒ Ficheros

☒ Wiki

☒ Repositorio

☒ Foros

☒ Calendario

☒ Gantt

Default trackers for new projects

☐ Errores

☐ Tareas

☐ Soporte

Generar identificadores de proyecto

☒

Permiso asignado a un usuario no-administrador para crear proyectos

--- Por favor seleccione ---

Guardar

Powered by Redmine © 2006-2013 Jean-Philippe Lang

- Una clave de la Api para la configuración del Issue Tracker- En Configuración>Repositorios chequeamos "Habilitar la gestión del repositorio y damos en Generar clave. La utilizaremos después, conviene copiarla ;)

Autorellenar los commits del repositorio

☒

Habilitar SW para la gestión del repositorio

☐

Clave de la API

Generar clave

Número máximo de revisiones mostradas en el fichero de trazas

100

Activamos el uso de Servicio web Rest desde Configuración > Autenticación

Configuración

General

Mostrar

Autenticación

Proyectos

Peticiones

Notificaciones por correo

Correos entrantes

Repositorios

Se requiere identificación

☐

Inicio de sesión automático

deshabilitado

Registro permitido

activación manual de cuenta

Permitir a los usuarios borrar sus propias cuentas

☒

Longitud mínima de la contraseña

8

¿Olvidaste la contraseña?

☒

Permitir identificación y registro por OpenID

☐

Activar servicio web REST

☒

Enable JSONP support

☐

Expiración de sesiones

Tiempo de vida máximo de las sesiones

deshabilitado

Tiempo máximo de inactividad de las sesiones

deshabilitado

Advertencia: el cambio de estas opciones podría hacer expirar las sesiones activas, incluyendo la suya.


Guardar

Powered by Redmine © 2006-2013 Jean-Philippe Lang

Ahora continuaremos desde TestLink.

Creamos un Issue Tracker como hicimos con Jira, pero esta vez de tipo Redmine Rest. Igualmente podemos ver un ejemplo haciendo click en Show configuration example.

**Issue Tracker Management**

Edit 

**Issue Tracker**

**Type**  [Show configuration example](#)

**Configuration**

```
<issuetracker>
<apikey>jVcBG1kWR3eTK19V9egW</apikey>
<uribase>http://localhost:3000</uribase>
<projectidentifier>mikiyuki</projectidentifier>
</issuetracker>
```

**Configuration example**

**Used on Test Project**

Mikiyuki

Igual que lo hacíamos con Jira lo linkeamos a un Proyecto. A la hora de ejecutar un test podemos, igual que con Jira, vincular los Issues o crear uno nuevo en Redmine.

Test Suite : suite de prueba 2/

Test Case ID MKY-26 :: Version : 1

caso de prueba 1





No tester assigned

Last execution (any build) - Build : build prueba 3

Access to Bug Tracking System (redmine)

Date : 22/05/2013 00:21:07 - Tested by : admin - Build : build prueba 3 - Status : Failed

Last execution (current build) - Build : build prueba 3

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 00:21:07	build prueba 3	admin	Failed	1		 	


Build  
build prueba 3

Relevant bugs

Voy a enseñaros un ejemplo de vincular a un issue ya existente:

TestLink - Google Chrome





localhost/testlink-1.9.5/lib/execute/bugAdd.php?user\_action=link&tcversion\_id=

Add bug report 

Access to Bug Tracking System (redmine)

redmine (Interface: rest) Bug id

Una vez hecho esto dará un mensaje de aprobación y podremos ver en TestLink que se ha asociado a la Issue creada anteriormente en Redmine con ese id.

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 00:21:07	build prueba 3	admin	Failed	1		 	
Build		Relevant bugs					
build prueba 3		1: [Nueva] : BLA					

## 5. Integración de Bugzilla en TestLink

Bugzilla es una herramienta para el manejo de fallos y bug puramente dicha. Nos permite manejar el proceso del desarrollo de nuestro software mejorando la comunicación y aumentando la calidad de nuestro software. Si te interesa saber más de esta herramienta visita el tutorial que os enlacé [antes](#) o visita la [página oficial](#)

Para la integración de Bugzilla en TestLink seguiremos, casi a rajatabla, los mismos pasos que hicimos con Jira. De todas formas os haré un pequeño seguimiento.

Creemos nuestro fichero /cfg/bugzilla.cfg.php con la siguiente configuración:

```
1 define('BUG_TRACK_DB_HOST', 'localhost');
2 define('BUG_TRACK_DB_NAME', 'bugzilla');
3 define('BUG_TRACK_DB_USER', 'bugzilla');
4 define('BUG_TRACK_DB_PASS', 'bugzilla');
5 /** link to the bugtracking system, for viewing bugs */
6 define('BUG_TRACK_HREF', "http://localhost/bugzilla-4.4/show_bug.cgi?id=");
7 /** link to the bugtracking system, for entering new bugs */
8 define('BUG_TRACK_ENTER_BUG_HREF', "http://localhost/bugzilla-4.4/");
```

Hecho esto vamos al fichero custom\_config.inc.php y establecemos lo siguiente:

```
$g_interface_bugs='BUGZILLA';
```

Lo siguiente sera crear un Issue tracker en TestLink:

Lo asignamos a nuestro proyecto.

Y a partir de ahora podremos vincular nuestros test fallidos a un Bug en Bugzilla.

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 00:21:07	build prueba 3	admin	Failed	1			

Vemos como se vinculan correctamente:

Last execution (any build) - Build : build prueba 3

Access to Bug Tracking System (bugzilla)

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 00:21:07	build prueba 3	admin	Failed	1			

Podemos vincular varios Issues al mismo caso de prueba:



Test Suite : suite de prueba 2/

Test Case ID MKY-26 :: Version : 1

caso de prueba 1

No tester assigned

Last execution (any build) - Build : build prueba 3

Access to Bug Tracking System (bugzilla)

Date : 22/05/2013 00:21:07 - Tested by : admin - Build : build prueba 3 - Status : Failed

Last execution (current build) - Build : build prueba 3

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 00:21:07	build prueba 3	admin	Failed	1			
Relevant bugs							
1 : IN_PROGRESS : Prueba 01							
3 : IN_PROGRESS : Prueba 02							

Y el mismo Issue a varios casos de prueba:

Test Suite : suite de prueba despues de implementar tracker/

Test Case ID MKY-27 :: Version : 1

test case

No tester assigned

Last execution (any build) - Build : build prueba 3

Access to Bug Tracking System (bugzilla)

Date : 22/05/2013 09:52:28 - Tested by : admin - Build : build prueba 3 - Status : Failed

Last execution (current build) - Build : build prueba 3

Date	Build	Tested by	Status	Test Case Version	attachments	BUG management	Run mode
22/05/2013 09:52:28	build prueba 3	admin	Failed	1			
Relevant bugs							
3 : IN_PROGRESS : Prueba 02							

6. Conclusión

De esta manera vinculamos nuestras especificaciones del software con los problemas que hemos ido encontrando, de forma que tanto el cliente como el encargado de realizar las pruebas puede seguir una trazabilidad de los requisitos, de las pruebas y los posibles fallos. Mejorando así la calidad de nuestro software y de nuestro proyecto.

TestLink nos proporciona gran compatibilidad con nuestros Gestores de Issues, por lo que podemos trazar más nuestros proyectos, generar una documentación y un catálogo sólido de nuestras pruebas, ya sean unitarias como de aceptación.

Os recomiendo encarecidamente leer este [tutorial](#) sobre TDD, BDD y Pruebas de Aceptación realizado por [Daniel Díaz](#) para enlazarlo con este tutorial, saber mejor de que estamos hablando, y tratar de combinar estas técnicas con las opciones que nos proporciona TestLink.

Alguna sugerencia o duda podéis comentar.

Un saludo , [Rafael](#).

[Volver al índice](#)

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](#)