

¿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)



E-mail:
Contraseña:

Deseo registrarme
He olvidado mis datos de acceso

Estás en: Inicio Tutoriales Lanzar test de Selenium 2 en un navegador remoto

DESARROLLADO POR:

Juan Alonso Ramos
 Consultor tecnológico de desarrollo de proyectos informáticos.
 Ingeniero Técnico en Informática de Gestión e Ingeniero en Informática, especialidad en Ingeniería del Software
 Puedes encontrarme en Autentia: Ofrecemos de servicios soporte a desarrollo, factoría y formación
 Somos expertos en Java/J2EE

Anuncios Google [Java](#) [SMS API Java](#) [Video Tutorial Java](#) [Online Java Class](#)

Fecha de publicación del tutorial: 2009-02-26



Share | [Regístrate para votar](#)

Lanzar test de Selenium 2 en un navegador remoto

Índice de contenidos.

- 1. Introducción
- 2. Entorno
- 3. Instalar Selenium Server Standalone
- 4. Test de conexión a un servidor remoto
- 5. Conclusiones

1. Introducción

Vimos en el tutorial de [Introducción a Selenium 2 y WebDriver](#) cómo lanzar test de integración sobre una aplicación web. Vimos que el API de WebDriver puede hacer uso de la instancia del navegador instalado en el sistema operativo donde se lanzan los test lo que conlleva un problema para el caso de las pruebas con Internet Explorer. En entornos Unix, que además son la mayoría donde se suelen tener instalados sistemas de Integración Continua, no disponemos de este navegador por lo que la prueba fallaría.

En este tutorial veremos la forma de comunicarnos con un Internet Explorer de una máquina Windows desde un Mac para lanzar en ese navegador un test de Selenium.

2. Entorno

- MacBook Pro 15' (2.4 GHz Intel Core i5, 4GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Snow Leopard 10.6.6
- Java 1.6.0_22
- Selenium 2.0 Beta 2
- JUnit 4.8.2

3. Instalar Selenium Server Standalone

Antes de hacer nuestro test de Selenium que se conecta de forma remota a la máquina Windows, debemos instalar en esa máquina el Selenium Server Standalone que estará escuchando las peticiones de conexión por parte de los clientes encargados de realizar los test.

Nos descargamos desde la web de Selenium el jar `selenium-server-standalone-2.0b2.jar`. Una vez descargado arrancamos el servidor de Selenium mediante el siguiente comando:

```
1 | java -jar selenium-server-standalone-2.0b2.jar
```

Una vez arrancado nos muestra la URL de conexión a Selenium Server:

```
1 | RemoteWebDriver instances should connect to: http://127.0.0.1:4444/wd/hub
```

Lo importante es ver que Selenium Server está escuchando peticiones en el puerto 4444 y en la URI `/wd/hub`, aspecto a tener en cuenta en nuestro test.

4. Test de conexión a un servidor remoto

El test que vamos a hacer es muy sencillo, simplemente conectarnos a la página de Google. Lo importante es ver las clases que se utilizan para la configuración y conexión al servidor remoto.

```

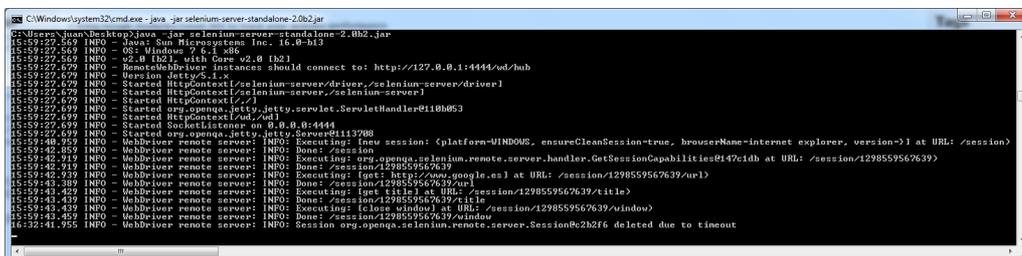
1 | @Test
2 | public void testRemoteInternetExplorer() throws MalformedURLException {
3 |     final DesiredCapabilities capabilities = DesiredCapabilities.internetExplorer();
4 |     final WebDriver driver = new RemoteWebDriver(new URL("http://192.168.1.131:4444/wd/hub"), capabilities);
5 |     driver.get("http://www.google.es");
6 |
7 |     Assert.assertEquals("Google", driver.getTitle());
8 |     driver.close();
9 | }

```

En la línea 3 configuramos el entorno de ejecución del test, en este caso le decimos que queremos arrancar el Internet Explorer instalado en la máquina remota.

A continuación (línea 4) creamos una instancia de la clase `RemoteWebDriver` indicando la URL de conexión al servidor Selenium remoto. Una vez conectado el driver, le indicamos que se cargue la página de Google y posteriormente comprobamos que lo haya hecho bien comparando el título de la misma. Finalmente cerramos el driver para que nos cierre el navegador.

En la consola de la máquina remota podemos ver los logs de arranque del Selenium server, conexión del cliente con los parámetros de configuración del driver para levantar el Internet Explorer, la petición de la página de Google y la comprobación del título de la misma.



5. Conclusiones

Catálogo de servicios Autentia

Últimas Noticias

- XV Charla Autentia - web2py (y Google App Engine)
- XIV Charla Autentia - ZK - Vídeos y Material
- Hablando de coaching ágil, milagro nocturno y pruebas de vida
- XIII Charla Autentia - AOS y TDD - Vídeos y Material
- Las metodologías ágiles como el catalizador del cambio



Últimos Tutoriales

- CRUD con Spring MVC Portlet (IV): Realizando pruebas unitarias
- Introducción a Selenium 2 y WebDriver
- Dividir tu pantalla gigante en Mac con Divvy
- Mi experiencia en Autentia
- Patrón Intérprete

Últimos Tutoriales del Autor

- Introducción a Selenium 2 y WebDriver
- Mapeo de Procedimientos Almacenados con Hibernate
- Autoescaneo de entidades de Hibernate con Spring
- DataTable con paginación en base de datos con Primefaces
- Generación de Informes con JasperReports en PHP

Síguenos a través de:



Últimas ofertas de empleo

- 2011-03-02 T. Información - Analista / Programador - MALAGA.
- 2011-02-24 T. Información - Especialista CRM - MADRID.
- 2011-02-16

Como hemos visto no hay problema a la hora de lanzar nuestros test en cualquier entorno ya que si no disponemos de los recursos desde la máquina que lanza los test, siempre podemos conectarnos de forma remota a otra máquina que sí disponga de ellos.

Lo que no nos cansamos es de recomendar que se realicen test de las aplicaciones. Selenium es una herramienta muy recomendable para aplicaciones ya terminadas que no dispongan de test unitarios, que son muchas. Nunca es tarde para hacer test incluso en la fase de mantenimiento ya que las pruebas de regresión pueden detectar bugs que se introducen de forma accidental cuando se modifican partes del código que no se piensa que puedan estar implicadas.

Una ventaja de Selenium 2 frente a versiones anteriores es que los test se ejecutan 100% en entorno Java por lo que no es necesario configurar nada adicional como se hacía antes para el caso de Selenium Remote Control por lo que nuestro servidor de Integración Continua lanzará los test de selenium como un test de JUnit más.

Espero que te haya sido de ayuda.

Un saludo. Juan.

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

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

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

 Marketing - Experto en Marketing - CADIZ.

2011-02-08

 Comercial - Ventas - CADIZ.

2011-01-28

 Comercial - Ventas - SEVILLA.

COMENTARIOS



Esta obra está licenciada bajo licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5

Copyright 2003-2011 © All Rights Reserved | Texto legal y condiciones de uso | Banners | Powered by Autentia | Contacto

