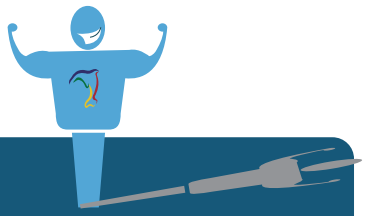
autentia

Avenida de Castilla,1 - Edificio Best Point - Oficina 21B
28830 San Fernando de Henares (Madrid)
tel./fax: +34 91 675 33 06
info@autentia.com - **www.autentia.com**

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

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

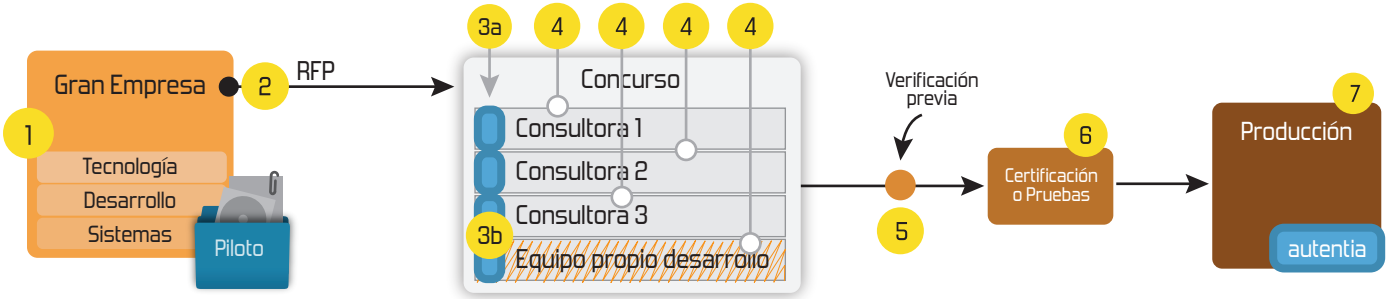
1. Desarrollo de componentes y proyectos a medida



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
- 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
- Tareas programadas (Quartz)
- Gestor documental (Alfresco)
- Inversión de control (Spring)
- BPM (jBPM o Bonita)
- Generación de informes (JasperReport)
- ESB (Open ESB)

Compartimos nuestro conocimiento en:
www.adictosaltrabajo.com

Para más información visítenos en:
www.autentia.com

juegos de pruebas sobre aplicaciones web.

Hay que indicar que desde esta herramienta **NO** se pueden **GRABAR** y **EDITAR** scripts, ya que esta tarea esta reservada a otra de las aplicaciones de SeleniumHQ : **Selenium IDE** (Esta herramienta tiene integrada Selenium Core).

Selenium Core esta basado en Javascript, de hecho, se ha realizado mediante DHTML y Javascript.Además cumple con las restricciones de seguridad impuestas a Javascript.Gracias a estas restricciones se evita en muchos casos proporcionar más información sensible de la necesaria, pero por otro lado presenta más de un quebradero de cabeza a la hora de planificar y ejecutar las pruebas.Muchos de estos problemas se irán planteando a lo largo de este tutorial.

Por otro lado, Selenium Core actua como componente principal de otra herramienta,Selenium Remote Control o Selenium RC (otra herramienta perteneciente al paquete Selenium HQ),pero hay que indicar que también se puede utilizar de forma independiente como herramienta para la realización de pruebas funcionales sobre aplicaciones web al permitir integrarse con ellas.

Características:

- Fácil instalación.
- Interfaz intuitiva.
- Facilidad en la ejecución de los test.
- Ejecución en varios navegadores (Muy importante).
- Referencia a objetos DOM
- Ejecución de scripts en modo "paso a paso".

En resumidas cuentas, esta aplicación permite integrar las pruebas dentro de la aplicación (como se verá en el apartado de Ejecución de Ejemplos) de tal forma que con una simple llamada en la URL se puede ejecutar un juego de pruebas o bien la interfaz desde donde se ejecutan los "Test Suite" con independencia del navegador utilizado.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Asus G50Vseries (Core Duo P8600 2.4GHz, 4GB RAM, 320 GB HD).
- Sistema operativo: Windows Vista Ultimate.
- Internet Explorer 7.0.6001.1800
- Mozilla Firefox 3.0.6
- Opera 9.63
- Safari 3.2.1

3. Requerimientos.

En este punto se indicará la compatibilidad de este plugin con diferentes configuraciones (navegador / sistema operativo / lenguaje de programación), describiendo las acciones permitidas o bien los problemas encontrados al ejecutarlo con esa configuración.

Navegador:

Navegador	**Funcionamiento**
Firefox 3	Reproducir test
Firefox 2	Reproducir test
IE 8b1	?
IE 7	Reproducir test
Safari 3	Reproducir test
Safari 2	Reproducir test
Opera 9	Reproducir test
Opera 8	Reproducir test
Otros	Reproducir test(*)

(*) Selenium Core esta escrito en Javascript y por lo tanto debería de funcionar en aquellos buscadores que soporten Javascript.

Importante: Algunas operaciones sólo se pueden realizar mediante Selenium RC debido a las opciones de seguridad del los navegadores.

Sistema operativo:

Sistema operativo	**Funcionamiento**
Windows	Reproducir test
OS X	Reproducir test
Linux	Reproducir test
Solaris	Reproducir test
Otros	Reproducir test (*)

(*) Selenium Core esta escrito en Javascript y por lo tanto debería de funcionar en aquellos buscadores que soporten Javascript.

Importante: Algunas operaciones sólo se pueden realizar mediante Selenium RC debido a las opciones de

(parte III)

2009-01-28
Eventos en Hibernate (parte II)

Últimas ofertas de empleo

2008-12-22
Otras - Mecánica - SEVILLA.

2008-11-27
Comercial - Ventas - ALICANTE.

2008-10-30
Comercial - Ventas - BARCELONA.

2008-10-30
T. Información - Analista / Programador - BARCELONA.

2008-10-27
T. Información - Analista / Programador - CIUDAD REAL.

Anuncios Google

seguridad del los navegadores.

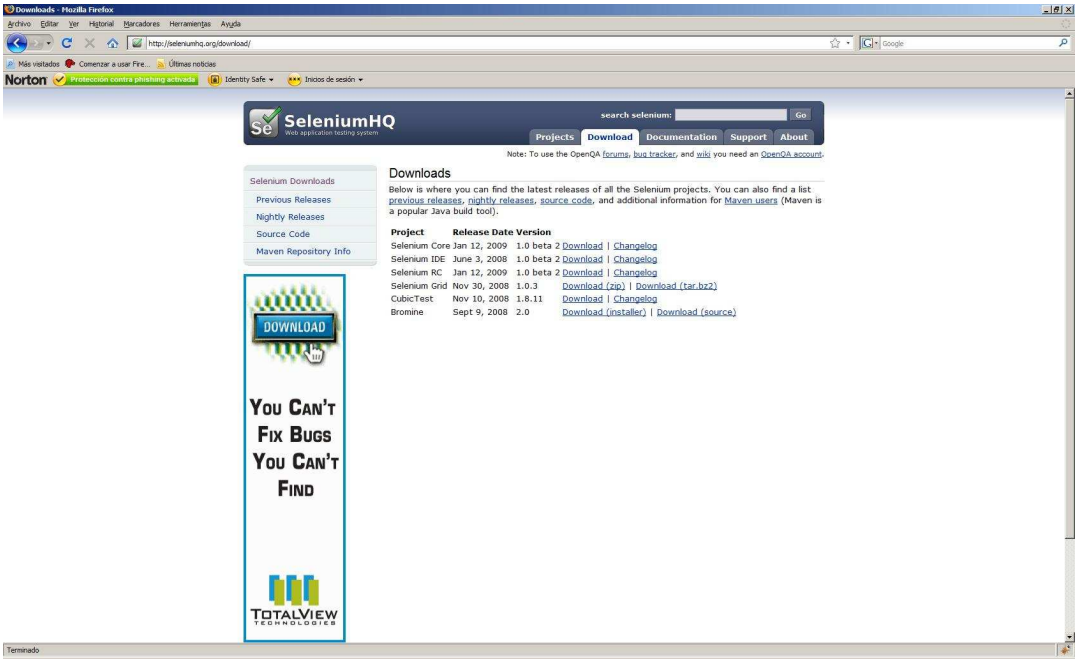
Lenguaje de Programación:

Lenguaje	Funcionamiento
C#	Genera código
Java	Genera código
Perl	Genera código
PHP	Genera código
Python	Genera código
Ruby	Genera código
Otros	Genera código personalizado

4. Instalación.

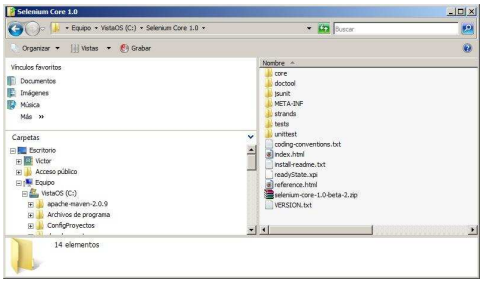
Estos son los pasos que hay que realizar para instalar Selenium Core:

Paso 1: Descargar Selenium Core desde su página web http://seleniumhq.org/download/.



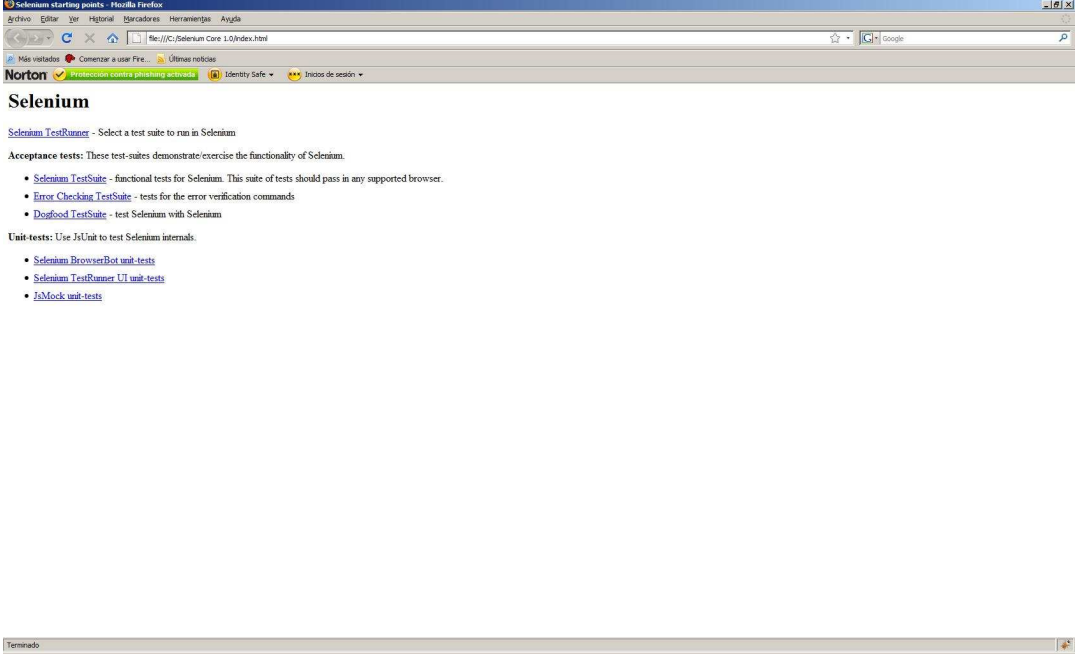
Paso 2: Pulsar sobre download.

Paso 3: Una vez se haya descargado, descomprimir el archivo en un directorio. Es aconsejable que el nombre del directorio sea Selenium core X.X (siendo el valor X.X la versión), de esta manera tendremos controlada la versión en todo momento.



Nota :Eliminar el fichero descargado del directorio (opcional).

Paso 4: Ejecutar el fichero "index.html", el cual confirmaría que Selenium Core esta "listo" para usarse, ya que todavía no se ha integrado en ningún proyecto o se ha ejecutado desde él un testuite. Desde esta página web local se podrán realizar una serie de operaciones que serán detalladas en posteriores puntos.



Incorporar Selenium Core al proyecto

Es en este apartado donde detallamos realmente la instalación (que es muy sencilla ;-)), ya que es aquí donde se explica como incorporar Selenium Core a nuestros proyecto webs.Pasos necesarios para su utilización:

Paso 1: Copiar la carpeta "core" del directorio anterior en un directorio web accesible (DocumentRoot, htdocs o webroot de su servidor web) de la aplicación que desea probar.

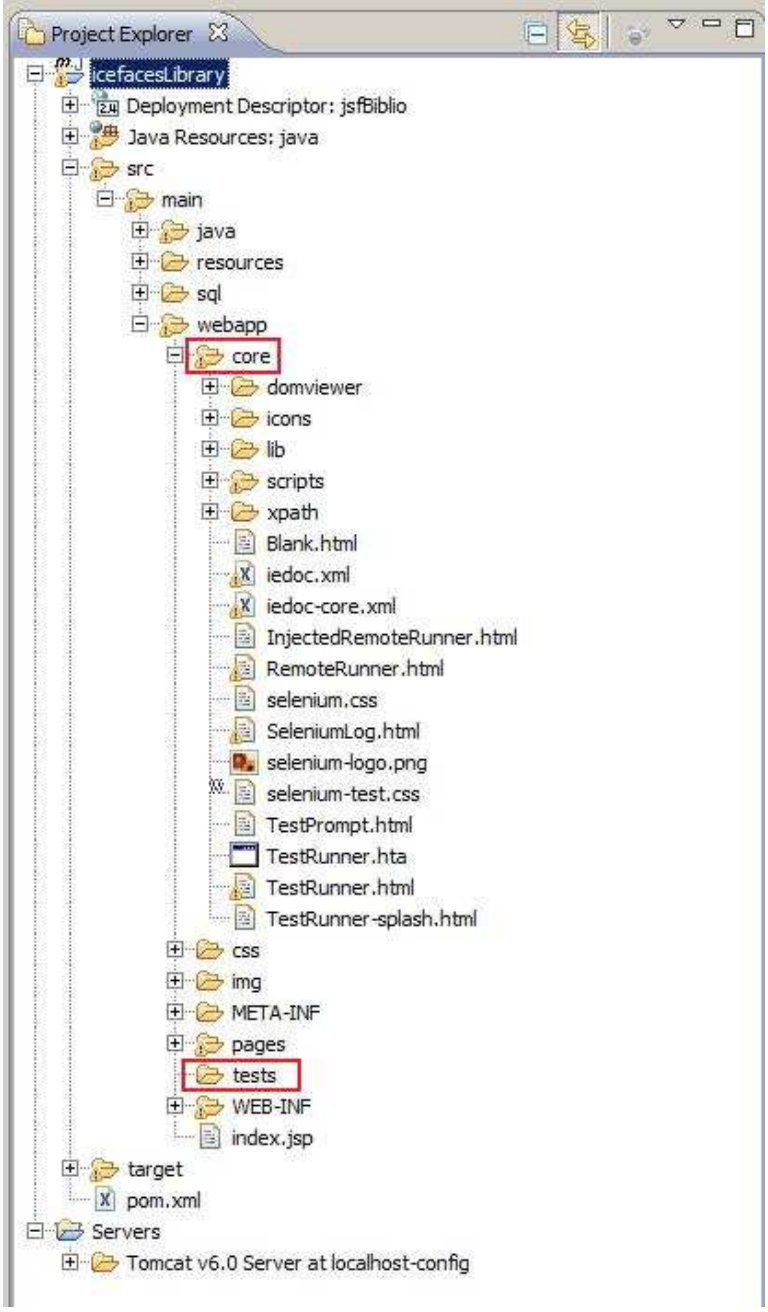
Nota : Si se esta usando Apache, este directorio será "htdocs".

Selenium Core necesita estar disponible en el mismo host y puerto de la aplicación.

¡¡¡MUY IMPORTANTE!!! (Tenedlo siempre en cuenta): Debido a la configuración de seguridad de Javascript de muchos navegadores, puede ocurrir que no funcione, por lo que quizás deberíais de utilizar Selenium IDE o Selenium Remote Control (Esta solución la plantean en SeleniumHQ).

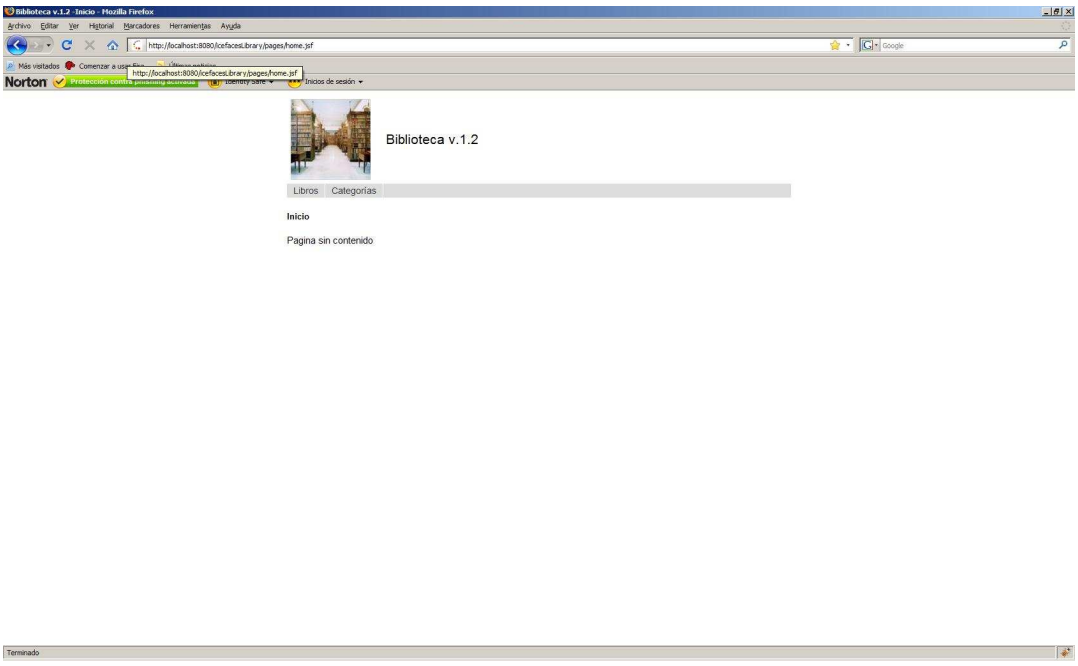
Paso 2: Crear al mismo nivel una carpeta denominada "test". Esta carpeta contendra todos los test Suite y test case que se van a utilizar.

En esta imagen se muestra la inclusión de las dos carpetas anteriores en un proyecto, de manera que estas sean visibles tal y como se indico.



Paso 3: Después desplegará la aplicación en el servidor utilizado (JBoss / Tomcat / Etc.).

Comprobaremos que se ha desplegado nuestra aplicación correctamente. La siguiente imagen muestra una aplicación de uso interno como ejemplo que se va a utilizar para explicar algunos conceptos.

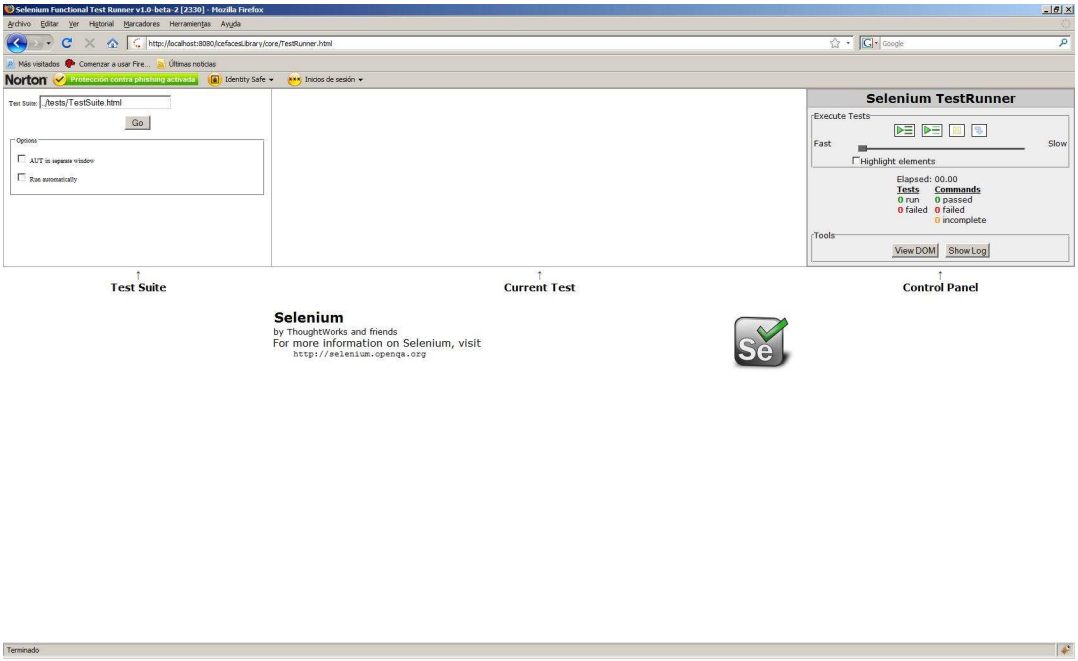


Paso 4: Para ejecutar Selenium Core desde el proyecto escriba en la URL:

http://<hostname>:<port>/core/TestRunner.html

Nota : Debido otras situaciones del carpeta "core" puede hacer cambiar un poco la URL.

La siguiente imagen muestra la ejecucción de TestRunner desde la aplicación cargada en el punto anterior.



En el apartado Ejecución de Ejemplo se indicará como continuar a partir de este punto.

5. Configuración.

Todavía queda un paso más para poder tener Selenium Core listo para funcionar.

Deberá de comprobar que la carpeta "test" esta creada, ya que en ella se almacenarán los Test Suite y Test Case que se van a utilizar para probar la aplicación. Por lo tanto, si no se ha creado la carpeta (la cual se indicó en el punto de Instalación) se debería de crear, teniendo en cuenta que para facilitar su ejecución debería de encontrarse al mismo nivel que la carpeta "core".

Si todavía no se dispone de los scripts de prueba es un buen momento para ver el tutorial de primeros pasos con Selenium IDE .

Importante: Selenium IDE es la herramienta desde la cual se generán los scripts.

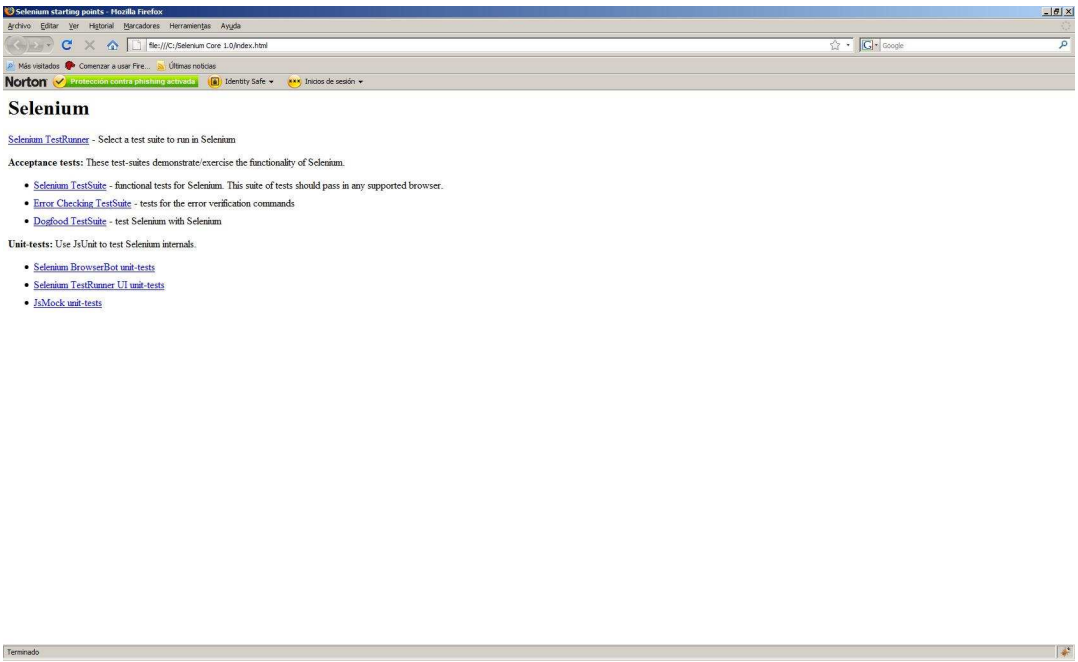
En el apartado Ejemplo se han generado dos ejemplos básicos de scripts y su posterior integración en un Test Suite de ejemplo.OJO : Estos script tiene truco que será resuelto en el apartado de Ejecución de Ejemplo.

Por otro lado y de forma opcional se puede modificar el fichero "TestPrompt.html" que se encuentra en el directorio "core" para cargar por defecto un Test Suite en particular. Para ello modificar el atributo con el nombre value por la nueva ruta o nombre del Test Suite.

<input id="test" name="test" size="30" value="../test/testuite.html"/>

6. Utilización.

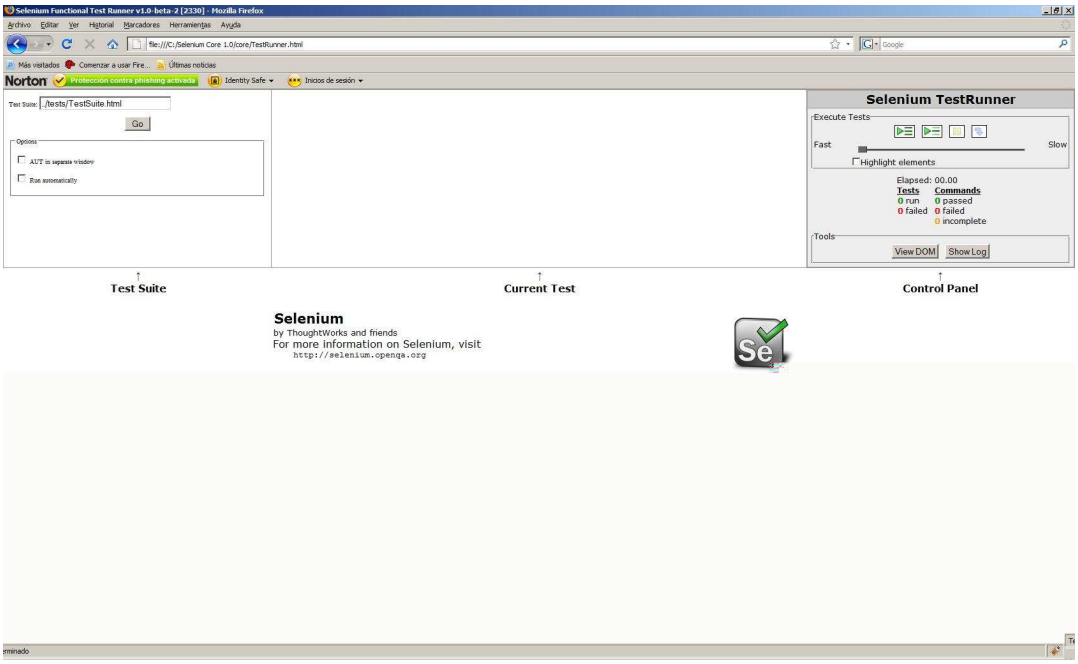
En este apartado se va a explicar en detalle las diferentes funciones que ofrece Selenium Core.Para ello es necesario ejecutar la página "index.html! que se encuentra en el directorio creado durante la instalación, es decir, la carpeta que estaba formado por el nombre y la versión (Por ejemplo : C:\Selenium Core 1.0).



En los siguientes subapartados se explicarán cada uno de los links de que dispone esta página:

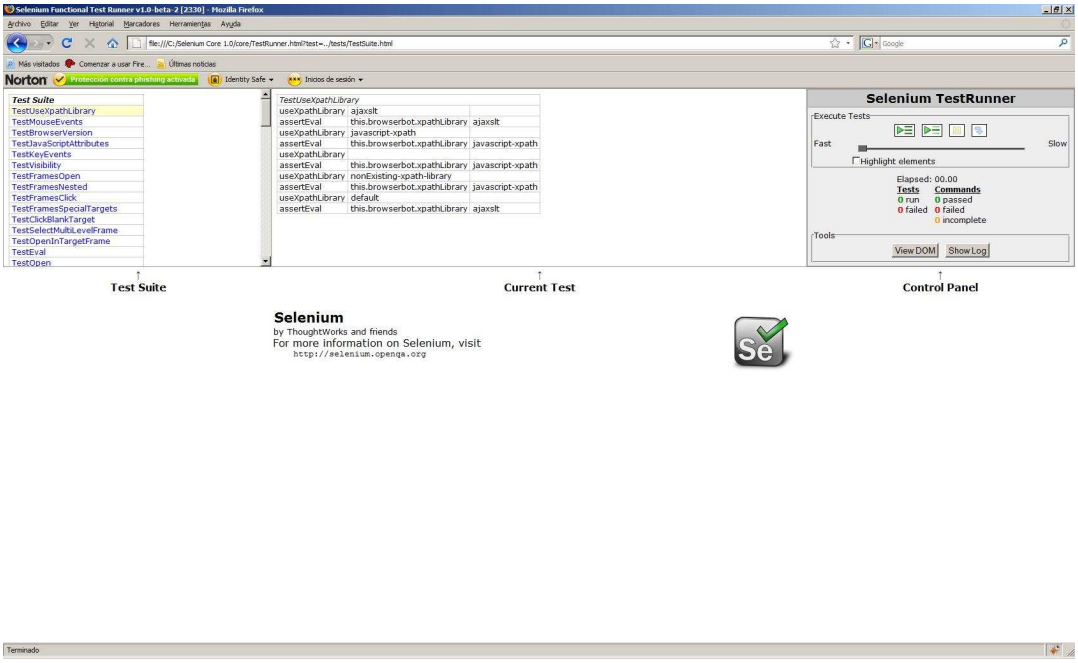
1. Selenium TestRunner

Selenium TestRunner en esta página se puede seleccionar un Test Suite para que sea ejecutado por Selenium de manera independiente,es decir,no es necesario que este integrado en un proyecto concreto.Pero si requeriría que el servidor estuviera arrancado y la aplicación desplegada si las pruebas fueran dirigidas a este sistema.



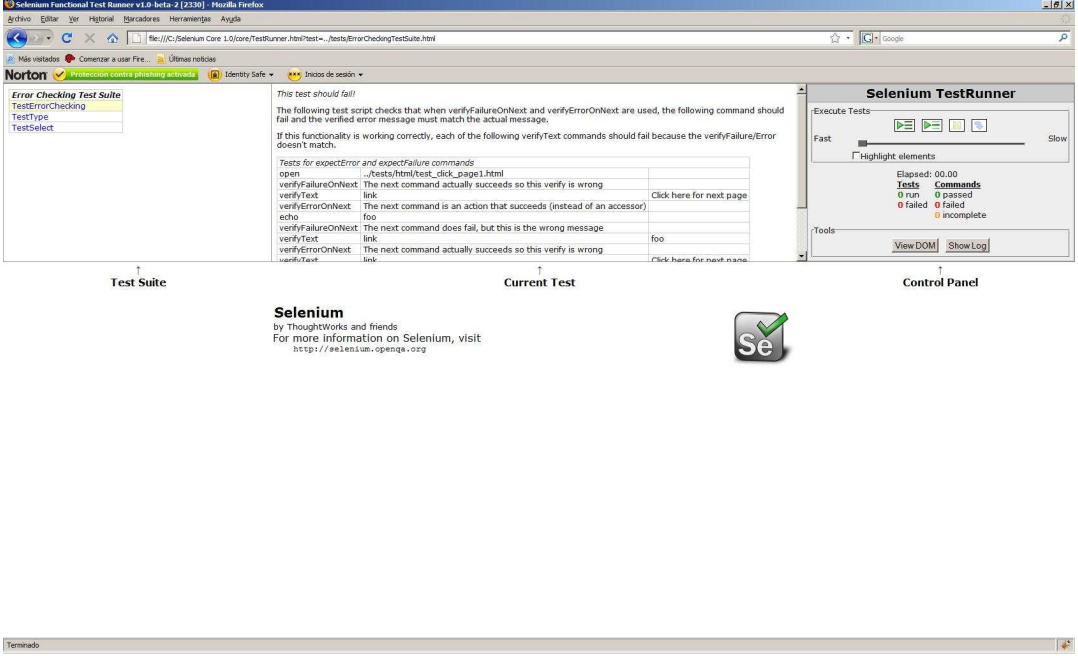
2. Test de aceptación

Selenium testuite es una página donde se pueden ejecutar una serie de test funcionales de Selenium para determinar si un navegador puede soportar los test de Selenium. Entre estos test se pueden encontrar test que comprueban que se pueden seleccionar un elemento en un desplegable, si reconoce los clic de de teclas, el tiempo de espera (retardo), etc..



3. Test de verificación de error

Error Checking testuite es un página similar a la anterior pero con la diferencia de que aquí se trata de comprobar como responde a la hora de verificar errores de comandos.



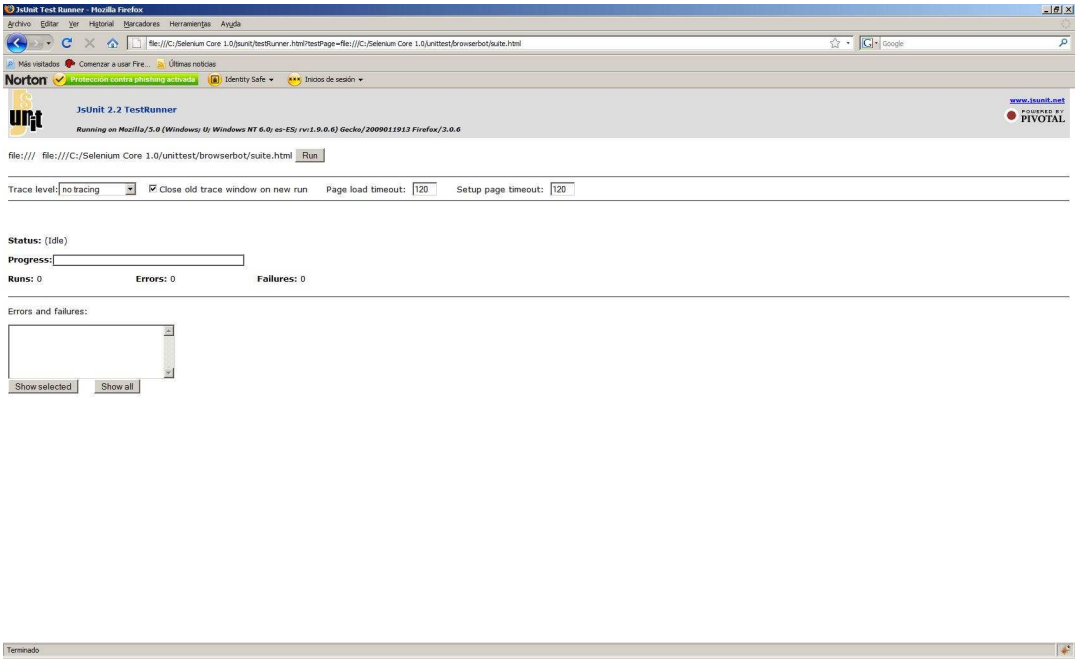
4. Test de verificación de error

Dogfood testuite Permite hacer un test a Selenium con Selenium (Este punto no es importante).

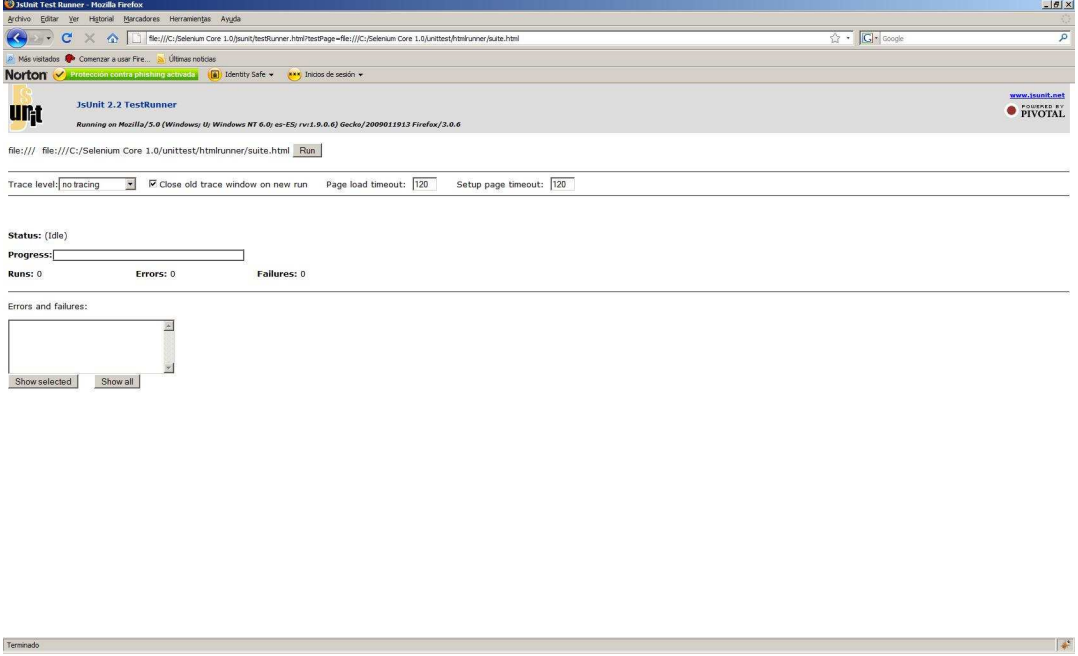
Test unitarios

Los siguientes apartados son para permitir realizar pruebas unitarias concretas, pero en este tutorial no se van a tratar puesto que no es el objetivo (Prometo hacer el tutorial ;-)).

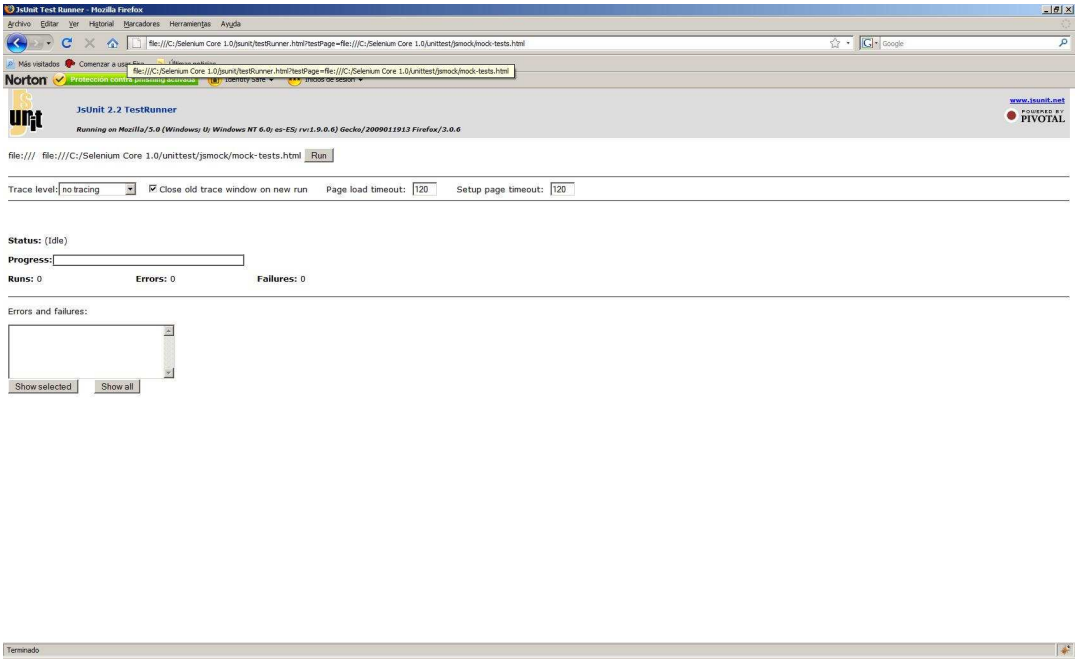
Selenium BrowserBot unit-test



Selenium TestRunner UI unit-test



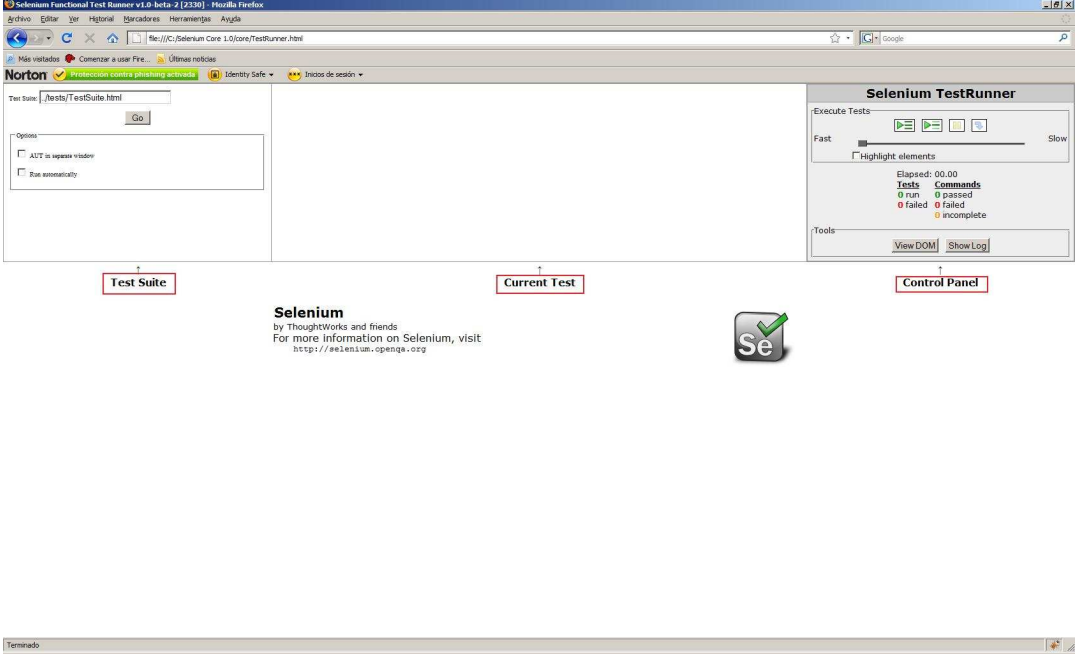
JsMock unit-test



6. Contenido.

En este apartado se explicará en detalle la pantalla de ejecución de Test Suite. Que como ya se ha visto se puede acceder a ella desde dos puntos:

- Desde el fichero **"index.html"** de la carpeta de instalación de Selenium Core
- Integrado en la aplicaciónd desplegada : **http://<hostname>:<port>/core/TestRunner.html**



Como se puede observar en la imagen la pantalla de Selenium se puede dividir en 3 partes claramente diferenciadas:

Test Suite

En esta pantalla se introduce la ruta relativa del Test Suite que será la carpeta "test" (mismo nivel que la carpeta "core") que se encuentre dentro del directorio de instalación de Selenium Core o bien la carpeta creada en el directorio accesible en la aplicación desplegada.

Las opciones permitidas en esta pantalla son:

- **AUT in separate window :** Permite ejecutar en una ventana independiente. (*)
- **Run automatically :** Ejecuta los Test Cases automáticamente una vez cargado el Test Suite.

(*) : Existe un problema cuando esta opción es ejecuta en Firefox ya que actualmente no admite la propiedad document.readyState, por lo tanto puede que Firefox se comporte de manera inestable si se selecciona esta opción. Como solución se plantea el uso de otra herramienta del paquete Selenium Remote Control o Selenium RC, que si permite la ejecución de Firefox en modo multiventana.

Nota:En el apartado Ejemplo se enseñará a generar un Test Suite a partir de unos test Cases.

Una vez cargado un Test Suite esta pantalla se actualizará y mostrára una imagen similar a esta.

Esta imagen representa el tener cargado Test Suite del apartado Ejemplo.

Test Suite
Acceder Adictos desde Google
Acceder Selenium desde Adictos

Seleccionando con el ratón un elemento de esta lista se puede ejecutar dicho Test Case de forma individual.

Current test

Esta pantalla permanece vacía hasta que se ha cargado un Test Suite, entonces se recarga con los comandos incluidos en el primero de los Test Case que se van a utilizar.

Esta imagen representa los comandos asociados al primer Test Case del Test Suite cargado en la pantalla anterior.

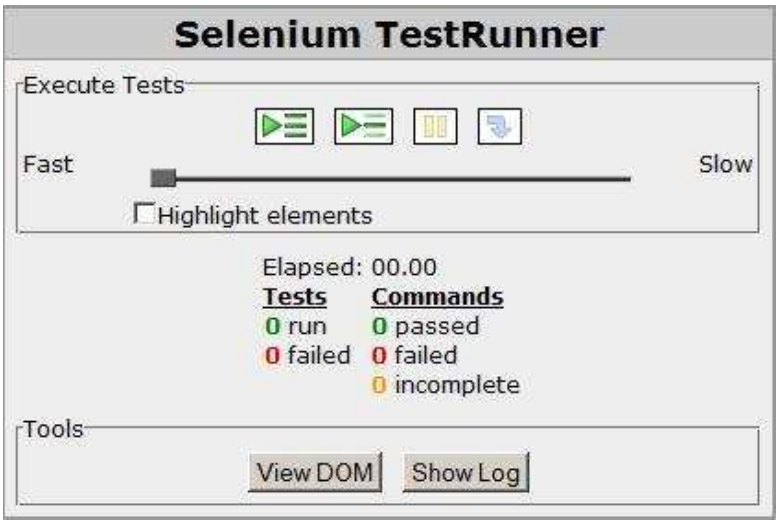
AccederAdictosDesdeGoogle		
open	http://www.google.es/	
assertTitle	Google	
type	q	Adictos al trabajo
clickAndWait	btnG	
assertTitle	Adictos al trabajo - Buscar con Google	
clickAndWait	link=Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML ...	
assertTitle	Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales sobre nuevas tecnologías.	

Seleccionando con el ratón un elemento de esta lista se puede incorporar interrupciones para permitir la ejecución en modo paso a paso.

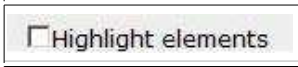
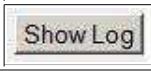
Control panel

En ese apartado se va a explicar en detalle el uso del panel de control.

La siguiente imagen es la pantalla que sirve para controlar la ejecución de los scripts



En la siguiente tabla se detallan cada una de las opciones permitidas. Hay que recordar que para que tengan funcionalidad se ha tenido que cargar un Test Suite válido.

Opción	Descripción
	Ejecuta cada test que se encuentra contenido en el test suite.
	Ejecuta el test seleccionado en la pantalla Test Suite mediante un clic.
	Para la ejecución de un test.
	Continua con la ejecución de un test después de una parada.
	Tras seleccionar un test en Test Suite y marcar una línea en Current Case, se establece una interrupción.Permite por tanto ejecutar paso a paso los siguientes comandos.
	Establece la velocidad de ejecución del script (incorpora retardos).
	Resalta los elementos que se estan ejecutando.
	Resumen del tiempo empleado y una estadística de los test.
	Visualiza el DOM del documento.
	Visualiza el log (Solo funciona mientras esta ventan este activa).

7. Ejemplo.

En este punto se va a enseñar a generar un Test Suite a partir de varios test Cases. Para ello se han generado dos script que nos van a permitir explicar este punto y una serie de problemas que plantean, pero esto se hará en el siguiente punto.

Nota: Tanto los scripts como el Test Suite han sido probados desde Selenium IDE y funcionan correctamente.

Ejemplo 1 Test Case : Acceder a la página de "Adictos al trabajo" desde Google

view plain print ?

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTI
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head profile="http://selenium-ide.openqa.org/profiles/test-case">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="selenium.base" href="" />
<title>New Test</title>
</head>
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">New Test</td></tr>
</thead><tbody>
<tr>
    <td>open</td>
    <td>http://www.google.es/</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Google</td>
    <td></td>
</tr>
<tr>
    <td>type</td>
    <td>q</td>
    <td>Adictos al trabajo</td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>btnG</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al trabajo - Buscar con Google</td>
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>link=Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML ...</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales s
    <td></td>
</tr>

</tbody></table>
</body>
</html>
```

Copia el contenido y genera un fichero denominado **AccederAdictosDesdeGoogle.html**

Ejemplo 2 :Acceder al tutorial de Selenium IDE desde la página de "Adictos al trabajo"

view plain print ?

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTI
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head profile="http://selenium-ide.openqa.org/profiles/test-case">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="selenium.base" href="" />
<title>New Test</title>
</head>
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">New Test</td></tr>
</thead><tbody>
<tr>
    <td>open</td>
    <td>http://www.adictosaltrabajo.com/</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales s
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>link=Quienes somos</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales s
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>//div[@id='who-are-we']/ul/li[20]/p[2]/a</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales s
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>link=Primeros pasos con Selenium IDE</td>
    <td></td>
</tr>
<tr>
    <td>assertTitle</td>
    <td>Adictos al Trabajo. Formación y desarrollo | JAVA, JEE, UML, XML |. Tutoriales s
    <td></td>
</tr>
<tr>
    <td>verifyTextPresent</td>
    <td>Primeros pasos con Selenium IDE</td>
    <td></td>
</tr>
</tbody></table>
</body>
</html>
```

Copia el contenido y genera un fichero denominado **AccederTutorialSeleniumDesdeAdictos.html**

Una vez llegados a este punto,con los scripts generados y listos, vamos a generar un Test Suite.Para ello disponemos del ejemplo funcional que los incorpora.

Ejemplo Test Suite

```
view plain  print  ?

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTI
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <meta content="text/html; charset=UTF-8" http-equiv="content-type" />
  <title>Testing Users</title>
</head>
<body>
    <table id="suiteTable" cellpadding="1" cellspacing="1" border="1" class="selenium">
    <tbody>
        <tr><td><b>Testing Users</b></td></tr>
        <tr><td><a href="./TestCases/AccederAdictosDesdeGoogle.html">Acceder Adictos des
        <tr><td><a href="./TestCases/AccederTutorialSeleniumDesdeAdictos.html">Acceder S
        </tbody>
    </table>
</body>
</html>
```

Copia el contenido y genera un fichero denominado **testuite.html**

Como se puede observar los anteriores scripts apararecen listados dentro de una estructura del tipo tabla, es muy importante situar los scripts en una ruta clara, como se puede observar en el ejemplo los dos scripts se han introducido en un directorio que se encontrará en la ruta donde se encuentre el Test Suite. Por otro lado, recordar los nombres utilizados y que estos deben de ser en este caso del tipo HTML (existen otros tipos pero se explicarán en Selenium RC).

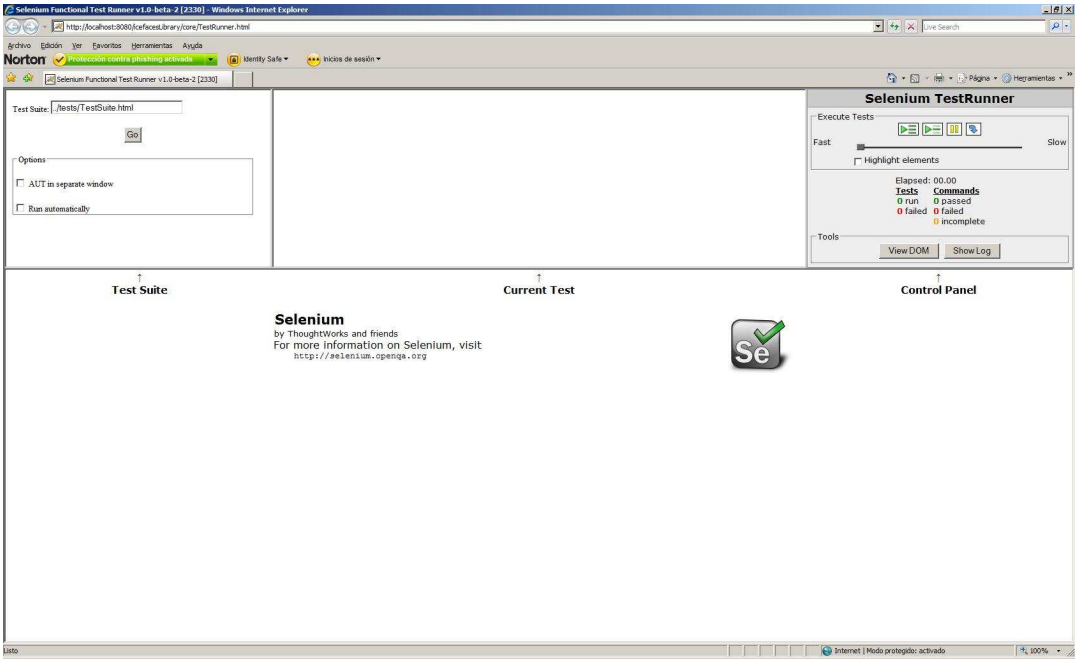
Importante : Los scripts se situan y se ejecutan en orden secuencial, por lo tanto no valdrá de nada insertar un producto en una aplicación de gestión si previamente no se ha hecho el login (si es requerido).

8. Ejecución de Ejemplo.

Tal y como se indico al principio del tutorial se ha comprobado que se carque Selenium Core con los navegadores indicados en el apartado Entorno.

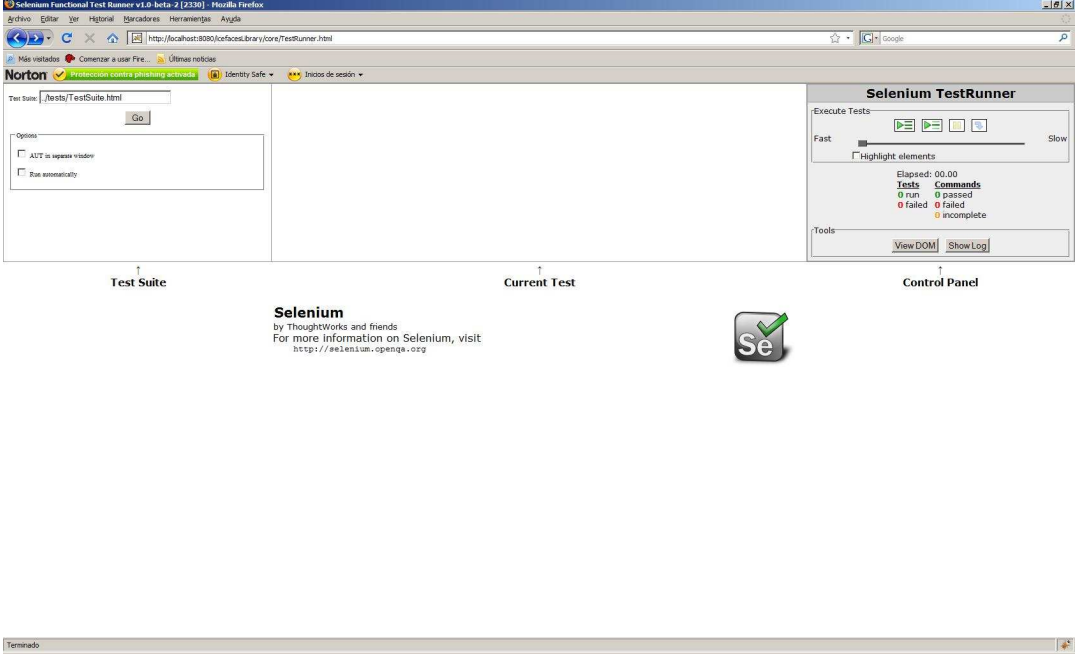
Internet Explorer 7

Versión : 7.0.6001.1800



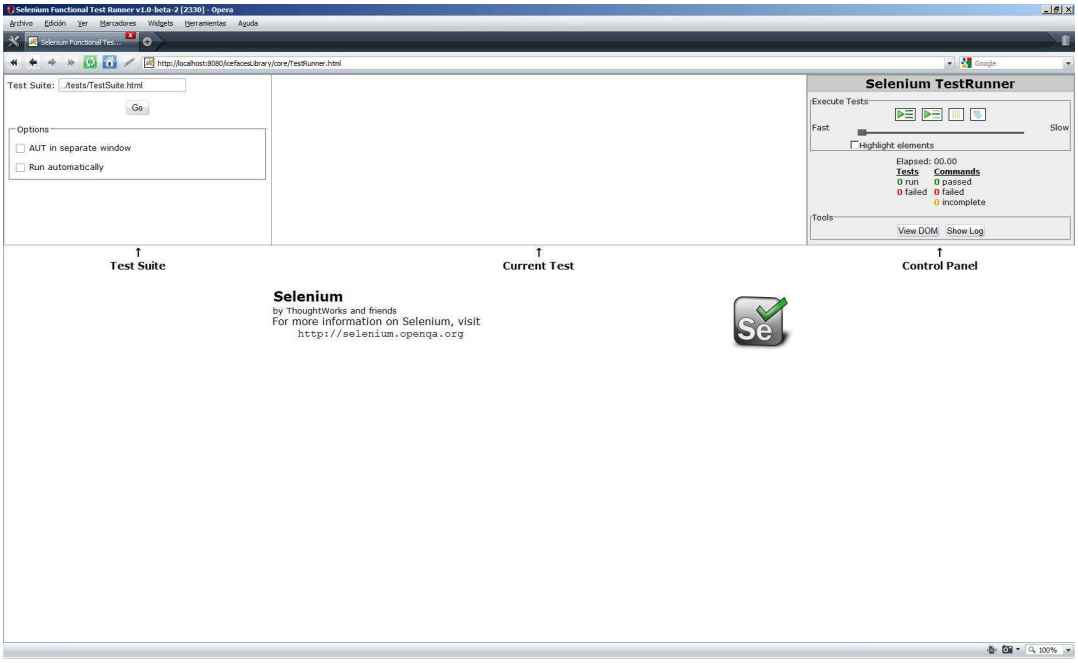
Mozilla Firefox 3

Versión : 3.0.6



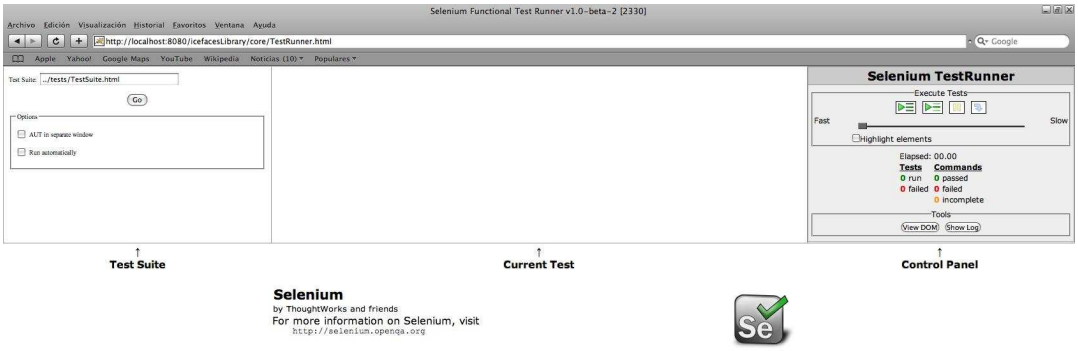
Opera 9

Versión : 9.63



Safari 3

Versión : 3.2.1



Como ya se ha dicho en el punto anterior, los ejemplos no tienen ni producen ningún tipo de error al ser ejecutados desde Selenium IDE, ahora es un buen momento de recordar que a lo largo del tutorial comente que estos scripts tenían "truco". Pues el truco es el siguiente, que cuando son ejecutados desde Selenium Core producen errores :-(.

Me imagino que llegados a este puntos plantearéis las siguientes preguntas : ¿Por qué producen errores? y ¿si Selenium IDE contiene a Selenium Core qué esta pasando? Pues el problema reside en el aspecto que he ido remarcando durante todo el tutorial y que es el siguiente:

La configuración de seguridad de Javascript de muchos navegadores no permite que se realicen muchas de las operaciones (comandos) que ejecutan estos u otros scripts, por lo que los considera ataques y los termina bloqueando o no concediendo permiso.

Todo esto, esta contemplado por SeleniumHQ y la solución que plantean es la utilización de 2 de sus herramientas que no presentan este problema o mejor dicho que lo tratan de solucionar :

- **Selenium IDE :** Funciona porque Mozilla Firefox lo considera una extensión de confianza.
- **Selenium RC :** Funciona porque actua como un servidor proxy para el navegador (engañándolo y haciendolo pensar que tiene instalado lo necesario para que se ejecute sin problemas).

Selenium Core exige estar instalado en el servidor de donde se esta ejecutando la aplicación, por lo tanto no podemos lanzar estos scripts ya que Google tendría que tener instalado Selenium Core en sus servidores.Me hago esta pregunta ¿y por qué no lo tendrán para que nos dejen probar estos scripts? ;-)

Por lo tanto, se puede decir que Selenium Core es más sensible a todo lo relacionado con la seguridad y que al ejecutarlo presentará en más de alguna ocasion algún que otro problema.

Pero no todo iba a ser malo, por ejemplo Selenium Core no presenta problemas si lo incorporamos a nuestros servidores y aplicaciones para que estos funcionen, de hecho lo hace sin ningún problema (eso es lo bueno de ser los "jefes" de nuestra aplicación). Asi que toda la funcionalidad que se ha ido contando durante el tutorial no se pierde, simplemente la adaptamos a nuestras necesidades. ;-)

En definitiva, si ejecutamos los scripts de nuestra aplicación web sobre ella, estos se ejecutan sin ningún problema.

9. Integración.

Selenium se puede integrar con constructores automáticos.Para ello permite la incorporación de parámetros en la URL que tienen funcionalidades diversas y que ahora detallare en las siguientes tablas.

Para ello, en muchos casos se requiere disponer de una aplicación web que gestione los resultados POST de las pruebas.La idea ser&iaacute;a utilizar alguno de los lenguajes sobre los que funciona Selenium o bien utilizar los recuros proporcionados por Selenium Remote Control.

Pasos necesarios para realizar la integración:

1. Crear una aplicación tipo Servlet que pueda leer los parámetros y grabarlos en un archivo.
2. Crear un script de arranque del navegador donde se le establezcan los parámetros (Generalmente llamada directa URL).
3. Tratar de que el anterior script llame a más de un navegador.
4. Finalizar comprobando la existencia del fichero de resultado.
5. Responder al finalizar (aviso pantalla, email, etc.)

Parámetros GET

Parámetro	Tipo	Valor por defecto	Descripción
auto	booleano	falso	Ejecuta el script automáticamente y muestra el resultado.
test	URL relativa	No tiene (se ejecuta con auto=true)	Establece la URL del test suite a ejecutar.
resultsUrl	URL relativa	/postResults	Establece la URL donde se guardan los resultados.
close	booleano	falso	Cierra el navegador cuando el test ha termindo si auto=true
baseUrl	URL absoluta	URL absoluta indicada por parámetro	Modifica las rutas relativas por absolutas para ello incorpora al comienzo "http://" o "https ://", y termina con "/"
multiWindow	booleano	falso	Ejecuta los test en modo multi-ventana.
highlight	booleano	falso	Resalta los elementos que encuentra.
runInterval	Entero (milisegundos)	0	Para la ejecución entre comandos este número de milisegundos.
defaultLogLevel	cadena	falso	Establece el nivel del log ("debug", "info" es por defecto, "warn", "error" o "off"), se puede cambiar con el comando "setBrowserLogLevel".
save	booleano	falso	(Modo HTA) Si auto=true los resultados se guardan en un archivo (ruta indicada por resultsURL) en vez de ser publicados.

Parámetros POST

Parámetro	Descripción
result	Resultado del script (passed/failed) en función si se ha producido algún problema.

totalTime	El tiempo en segundos de todo el conjunto a ejecutar.
numTestPasses	Nº de test que han pasado.
numTestFailures	Nº de test que han fallado.
numCommandPasses	Nº de comandos que han pasado.
numCommandFailures	Nº de comandos que han fallado.
numCommandErrors	Nº de comandos con error.
suite	La tabla suite inclue un campo escondico con los resultados.
log	El texto de todos los log que se producen.
testTable.1	Test número 1
testTable.2	Test número 2
...	...
testTable.N	Test número N

10. Modo de ejecución HTA.

Es un modo de ejecución de Selenium Core que tiene las siguientes restricciones:

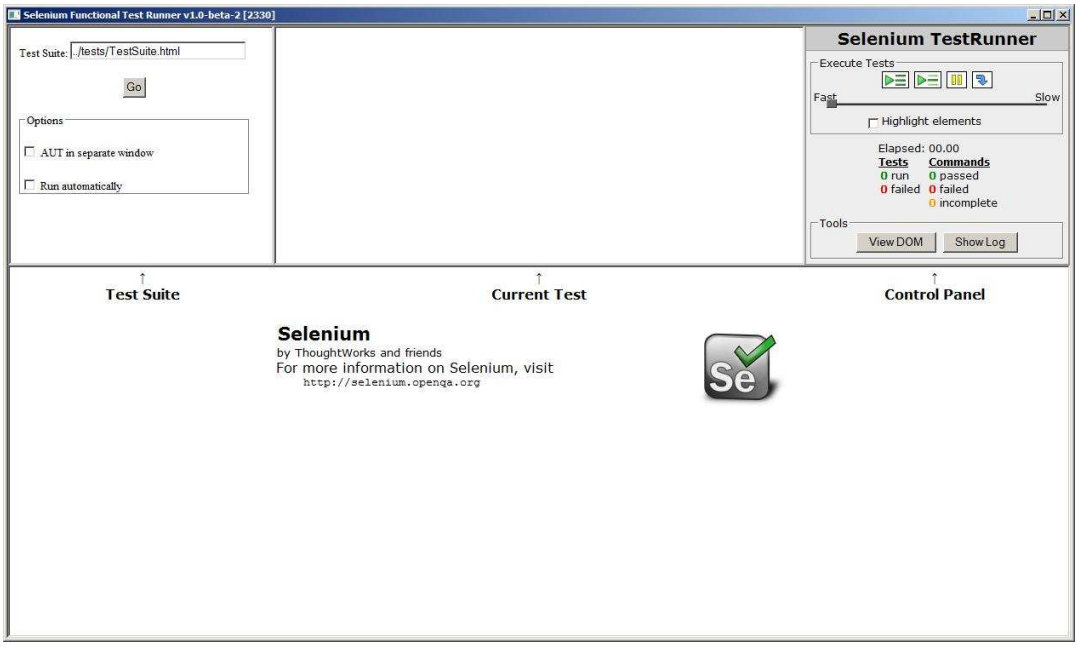
- Requiere Windows.
- Requiere Internet Explorer.

Para ello dispone de un archivo (HTA) que no es otra cosa que un tipo especial de arhivo HTML.Este archivo tiene autorización para saltarse las políticas de seguridad y escribir en archivos del disco (por ejemplo para escribir los resultados).No requiere tener instalado Selenium Core en el mismo servidor web.

Una forma de ejecutar este archivo es situarse en la carpeta de intalación de Selenium Core y dentro de la carpeta "core". En nuestro ejemplo:

C:\Selenium Core 1.0\core\

Ejecutar el archivo "TestRunner.hta".Aparecerá una ventana como la indicada en la siguiente imagen.



He probado a replicar la prueba del Test Suite y permite ejecutar el primer script y lo ha hecho sin problemas. Luego muestra un error con el segundo.

Como se puede comprobar en el mensaje, continuamos teniendo un problema de seguridad.

HTA2

Otra forma de ejecutar este archivo es realizarlo por línea de comandos. En nuestro ejemplo:

C:\Selenium Core 1.0\core>TestRunner.hta "test=..%2Ftest%2Ftestsuite.html&auto=true&close=true&resultsUrl=results.html&save=true"

Como se puede observar en el ejemplo, esta opción permite integrar los comandos de integración.

11. Conclusiones.

Selenium Core se presenta junto con Selenium IDE como el duo que nunca debe faltar a la hora de probar una aplicación web, sobre todo por que facilita mucho probar una aplicación en diferentes plataformas, ¿qué levante la mano quien no haya tenido ningún problema al realizar una prueba con dos navegadores (por ejemplo Internet Explorer / Mozilla Firefox)?...No veo demasiadas manos...jejeje. Lo dicho permite asegurarnos que el funcionamiento y sobre todo la calidad no varía en función del navegador que se este utilizando.

También hay que destacar su integración en el proyecto (sencilla y rápida) y su ejecución en cualquier momento a través de la URL.Se de muchos responsables de proyecto que esto les va a gustar mucho...jejeje.

Espero haberos ayudado un poquito con este tutorial.

Un saludo.

Víctor

mailto:vjmadrid@autentia.com

¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ¡vota!

Muy malo Malo Regular Bueno Muy bueno

Votar

- Puedes opinar sobre este tutorial haciendo clic aquí.
- Puedes firmar en nuestro libro de visitas haciendo clic aquí.
- Puedes asociarte al grupo AdictosAlTrabajo en XING haciendo clic aquí.
- Añadir a favoritos Technorati. ADD THIS BLOG TO MY Technorati FAVORITES

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

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido (Ver todos los tutoriales). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com

Servicio de notificaciones:

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales.

Formulario de subcripción a novedades:

E-mail [ ] Aceptar

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
Firmas Digitales muy Fácil con Firefox	En este tutorial vamos a ver como poder firmar documentos (certificados digitales) usando FireFox	2007-06-27	4520	Muy bueno	1	
Manual Avanzado de Firebug	Firebug es una extensión gratuita para Mozilla Firefox especialmente diseñada para todos aquellos programadores que se dedican al desarrollo web. En este tutorial os enseñaremos en profundidad su funcionamiento.	2009-02-09	355	Muy bueno	5	
Executor : Un programa para ejecutarlos a todos.	Nuestro amigo Victor nos enseña éste utilísimo programa para programar la ejecución de aplicaciones de manera sencilla y rápida	2009-01-19	671	Muy bueno	6	
Pruebas de Rendimiento y Funcionales Web	Jose María Toribio, nos enseña en este tutorial como podemos utilizar la aplicación gratuita JMeter para realizar pruebas de rendimiento y funcionales (vitales para la regresión y reingeniería) sobre nuestras aplicaciones Web	2005-04-17	36267	Muy bueno	8	
Primeros pasos con Selenium IDE	Nuestro amigo y compañero Víctor nos enseña, para acabar bien el año, el uso de una herramienta muy interesante para la realización de pruebas sobre aplicaciones web	2008-12-31	1164	Muy bueno	17	
Manual Básico de FireBug	Este tutorial nos va a enseñar una herramienta llamada FireBug, que es un plugin de Firefox que nos brinda un paquete de utilidades para el desarrollo de páginas y aplicaciones Web	2007-05-18	5857	Bueno	7	
Pruebas Software con Junit 4 y Eclipse	Junit se trata de un framework muy conocido en el mundo de los desarrolladores que con este tutorial intentamos acercar a quienes empiezan en este mundillo	2009-01-20	847	Bueno	14	
Pruebas unitarias Web para aplicaciones JSF	En este tutorial se puede encontrar una introducción y un análisis de los diferentes frameworks disponibles para realizar pruebas unitarias web de aplicaciones JSF	2006-11-13	6794	-	-	
Desarrollando portales para móviles con FireFox	En este tutorial, se va a presentar User Agent Switcher una extensión para el navegador Web FireFox que nos permite de una forma sencilla emular y probar aplicaciones Web sobre cualquier teléfono móvil a través del propio navegador.	2007-03-19	2723	-	-	
Navegador Mozilla FireFox	En esta ocasión probamos el estado de evolución del navegador gratuito Mozilla FireFox, una verdadera alternativa en el mercado.	2005-03-01	5590	-	-	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.