

¿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)



Empieza el jaleo: publicamos el libro
INFORMÁTICA PROFESIONAL
LAS REGLAS NO ESCRITAS PARA TRIUNFAR EN LA EMPRESA

-¿Crees que la informática es sólo programar?
-¿Sabrías organizar eficientemente un equipo?
-¿Quién te resuelve las dudas sobre esta profesión?..... o te siembra más?

Hosting patrocinado por **ENREDADOS**



[Inicio](#) [Quiénes somos](#) [Tutoriales](#) [Formación](#) [Comparador de salarios](#) [Comentar libro](#) [Charlas](#) [Más](#)

Estas en: [Inicio](#) [Tutoriales](#) Pruebas funcionales de servicios web con soapUI

Últimas Noticias

- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » Tarta y 7º aniversario de AdictosAlTrabajo.com
- » Formación por calendario
- » Comentando el libro: Eneagrama y éxito personal de Ginger Lapid-Bogda
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Disponible la primera versión de los plugins para integrar Maven y Bugzilla.
- » Lanzamiento del nuevo Web de Autentia
- » Si se pregunta ¿Qué ofrece este Web?

+Noticias Destacadas

- » Impresiones y video de asistentes al primer coding-dojó de agilismo.es
- » "Informática profesional: Las reglas no escritas para triunfar en la empresa." Pincha para comprar..
- » Autentia patrocina el primer Agile Open Spain
- » Lanzamiento del nuevo Web de Autentia

+Comentarios Cómico

+Enlaces

Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...

Acceso de usuarios registrados:

E-mail:
Contraseña:

[Deseo registrarme](#)
[He olvidado mis datos de acceso](#)

Registra tu empresa:

Descubre las ventajas de registrar tu empresa en AdictosAlTrabajo...
[Registrar mi empresa](#)
[Listado de empresas ya registradas](#)

Tutorial desarrollado por



Iván García Puebla

Consultor tecnológico de desarrollo de proyectos informáticos.
Puedes encontrarme en [Autentia](#)
Somos expertos en Java/JEE

Catálogo de servicios de Autentia

Descargar (6,2 MB)

Descargar en versión comic (17 MB)

AdictosAlTrabajo.com es el Web de difusión de conocimiento de Autentia.



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [pruebas-funcionales-soapui.pdf](#)

Fecha de creación del tutorial: 2009-12-28

Pruebas funcionales de servicios web con soapUI

Índice

1. Pruebas funcionales de servicios web con soapUI 3.0.1
 1. Introducción
 2. Prerrequisitos
 3. Crear el proyecto en soapUI
 4. Pruebas de servicios web sobre SOAP
 5. Pruebas sobre el SOAP response
 6. Añadir nuevos casos de prueba
 7. Ejecución conjunta de los casos de prueba
 8. Ejecución de una suite completa
 9. Características avanzadas de la versión soapUI Pro
 1. Generación de informes
 2. Panel de cobertura
 10. Conclusión

Introducción

Las pruebas unitarias en cualquier paradigma de programación son, más que una buena práctica, una garantía para obtener un software robusto y (más) fácilmente mantenible. Como responsables de diseño o desarrollo de web services hemos de aplicar estas buenas prácticas, y soapUI es una herramienta fenomenal para ello, como veremos a continuación.

A propósito de lo anterior, [Leo Antolí](#) (evangelizador de las metodologías ágiles y experto en [Autentia](#)) nos recomendó el podcast #64 de [JavaHispano](#) sobre [Test de Aplicaciones](#). Os invitamos a escucharlo.

Prerrequisitos

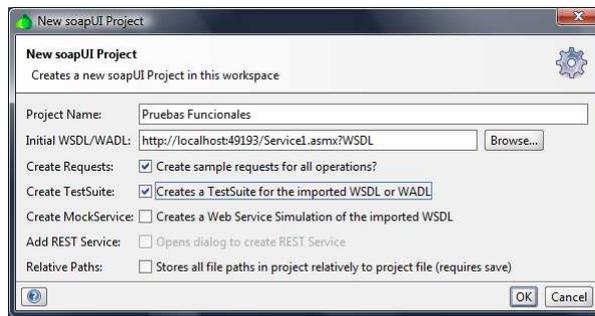
Este tutorial se ha desarrollado bajo el siguiente entorno:

- [soapUI 3.0.1](#). Podemos encontrar una introducción a la herramienta en el tutorial: [soapUI: jugando con web services](#)
- El web service desarrollado en el tutorial [Contract-First web services con Visual Studio 2008](#), y cuyo código fuente puede descargarse aquí: [wsEncryption_adictosaltrabajo.zip](#) y ser ejecutado tanto en Microsoft Visual Studio 2008 como en la versión de libre uso Visual Studio 2008 Express de [Visual Web Developer 2008 Express Edition](#)

Crear el proyecto en soapUI

Una vez tenemos el web service publicado, accedemos a soapUI y desde el menú **File | New soapUI Project**, creamos un proyecto para el tutorial con los datos:

- Nombre: Pruebas Funcionales
- WSDL: <http://localhost:49193/Service1.asmx?WSDL>
- Dejamos activado el checkbox: Create ample requests for...
- Marcamos la opción: Creates a TestSuite for the imported WSDL...



OK al diálogo siguiente:

Últimos tutoriales

2009-12-29
[Tutorial de BPEL con OpenESB \(I\)](#)

2009-12-28
[Pruebas funcionales de servicios web con soapUI](#)

2009-12-28
[SoapUI: jugando con web services](#)

2009-12-17
[¿Qué son el cloud computing y google app engine?](#)

2009-12-14
[JavaBean DataSource Ireport](#)

2009-12-11
[Contract-First web services con Visual Studio 2008](#)

2009-12-09
[Integrando Sonar con Hudson](#)

2009-12-09
[Apache + Tomcat: Balanceo de carga y alta disponibilidad](#)

2009-12-08
[MySQL: Replicación de bases de datos en MySQL](#)

2009-12-07
[Analizando la calidad del código Java con Sonar](#)

2009-12-03
[Instalar OpenESB 2.1 e Introducción](#)

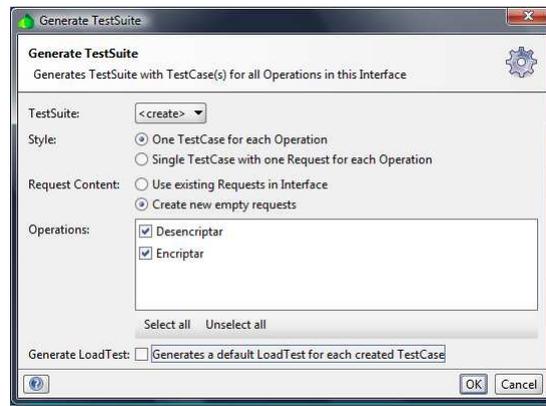
2009-11-25
[Tutorial de Google Forms](#)

2009-11-23
[Alfresco - Modificando y eliminando contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Alfresco - Añadiendo contenido desde nuestras aplicaciones Java](#)

2009-11-23
[Haciendo funcionar Google Chromium Operating System sobre Windows XP](#)

2009-11-23
[Redimensionar el tamaño de una partición de VirtualBox](#)



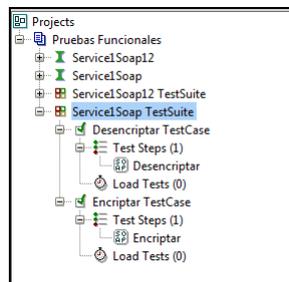
Opciones para generar pruebas de cada operación

Aceptamos el nombre propuesto por defecto:



Aceptamos el nombres propuesto

De igual manera aceptamos el resto de ventanas que aparecen. Al final tendremos la estructura del proyecto:

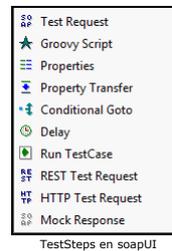


Proyecto de pruebas recién creado

En adelante vamos a trabajar únicamente con la suite de pruebas de una de las interfaces: `Service1Soap TestSuite` (hacerlo para la de SOAP 1.2 sería idéntico). Lo primero que vemos en el proyecto es que cada `TestSuite` tiene una batería de casos de pruebas para cada operación (`TestCase`) y cada batería se conforma de un conjunto de pasos o pruebas unitarias (`Test Steps`).

Pruebas de servicios web sobre SOAP

Como vemos en la imagen, soapUI permite definir hasta diez tipos de pruebas unitarias (botón derecho sobre `Test Steps` | `Add Step`):

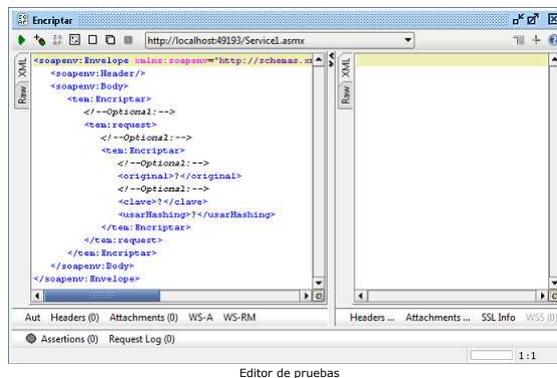


TestSteps in soapUI

En el tutorial nos vamos a centrar en pruebas de invocaciones sobre SOAP (que figura como `Test Request`) y los tipos de comprobaciones (assertions) permitidas.

Pruebas sobre el SOAP response

Pulsando dos veces sobre el nodo `Encriptar`, accedemos al editor de pruebas con un mensaje SOAP request construido según el Schema del servicio:



Editor de pruebas

Creamos un mensaje SOAP request hacia la operación `Encriptar` con valores de ejemplo sustituyendo los ?:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:tem="http://tempuri.org/">
  <soap:Header/>
  <soap:Body>
    <tem:Encriptar>
      <!--Optional-->
      <tem:request>
        <!--Optional-->
        <tem:Encriptar>
          <!--Optional-->
          <original>Lorem ipsum dolor sit</original>
          <!--Optional-->
          <clave>autentia2009</clave>
          <usarHashing>true</usarHashing>
        </tem:Encriptar>
      </tem:request>
    </tem:Encriptar>
  </soap:Body>
</soap:Envelope>
```

2009-11-20
El Arte de las Presentaciones. Siguiendo la Senda Zen.

2009-11-18
Tutorial basico de google wave bots

2009-11-13
Introducción a Escritorios Animado (Winamp y MilkWave)

2009-11-12
Maven Assembly Plugin: empaquetando aplicaciones con Maven para la ejecución de procesos batch.

2009-11-12
Tutorial básico sobre Google Application Engine

2009-11-11
Instalación de Glassfish 2.1

2009-11-26
Rational Software Architect y Rational RequisitePro

2009-10-27
Cómo conseguir que Subversion avise a Hudson para lanzar una build

2009-10-26
Cómo instalar Hudson en Apache Tomcat

2009-10-26
Estilos externos en iReport

2009-10-14
Echo en MySQL

2009-09-28
iReport : Solución al problema de los parámetros dinámicos.

2009-09-27
Mavenizar Liferay SDK

2009-09-27
Crear un plugin con Liferay SDK

2009-09-23
EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Apache Open EJB.

2009-09-21
Primeros pasos con JBoss Seam

2009-09-21
Integración con sistemas de Bug Tracking desde NetBeans 6.7

2009-09-14
EJB 3.0 y pruebas de persistencia con Maven, JUnit 4 y Embedded JBoss sobre Java 6.

2009-09-12
Instalación de Liferay en Tomcat existente

2009-09-11
Release Bugzilla Maven Plugin

2009-09-11
Enlazar Bugzilla con MavenChangesPlugin

2009-09-08
Sobre las reglas de codificación o... ¿de dónde salen esos caracteres "raros"?

2009-08-28
Cómo hacer deploy del site de Maven en SourceForge

2009-08-26
Ordenación por cantidades en informe cruzado

2009-08-20
Selenium IDE-Incorporando while en los test

2009-08-14
Blender y JMonkeyEngine. Exportación de archivos Blender y uso de los mismos en JMonkeyEngine

2009-08-14
5º tutorial TNT Concept Versión 0.16.1 Gestión de informes, vacaciones y utilidades

2009-08-14
Joomla 1.5. Instalación y configuración

2009-08-13

```

</tem:Encriptar>
</tem:request>
</tem:Encriptar>
</soap:Body>
</soap:Envelope>

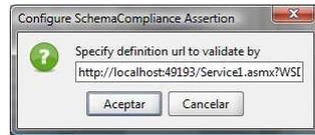
```

Sobre este mismo mensaje vamos a añadir algunas comprobaciones. Pulsamos sobre el botón  (a la derecha de la fecha verde), podemos seleccionar una aserción:

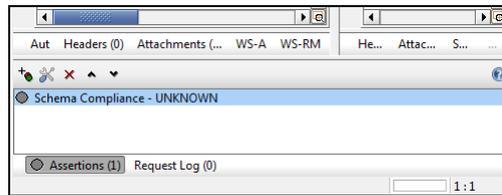


Assertions disponibles para pruebas SOAP

En primer lugar, comprobamos que el mensaje SOAP response es válido según un Schema. Seleccionamos la aserción `Schema Compliance`, pulsamos `Aceptar`, y nos preguntará en la siguiente ventana por la definición de datos Schema a aplicar. Introducimos la localización del WSDL de nuestro servicio (`http://localhost:49193/Service1.asmx?WSDL`):

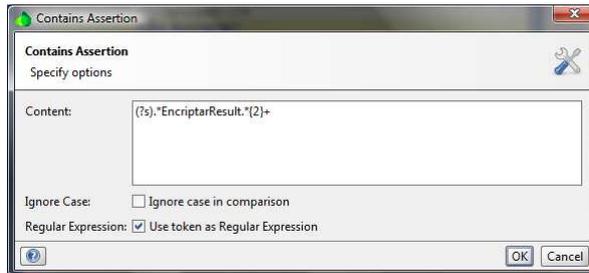


Pulsando sobre la palabra `Assertions` en la parte inferior del editor, podemos ver la lista de ellos a medida que los añadimos:



Lista de aserciones introducidas

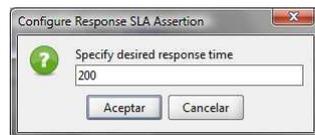
Añadimos una nueva aserción, en este caso de tipo `Contains`, y en la siguiente ventana de opciones introducimos el valor: `{?}*EncriptarResult.*{2}` y seleccionamos que sea una expresión regular, para comprobar que el token `EncriptarResult` aparece dos veces en la respuesta:



Comprobar que el SOAP response contiene determinado token

Añadimos otra aserción de tipo `NotSoapFault` para comprobar que el mensaje de respuesta no es de tipo fallo.

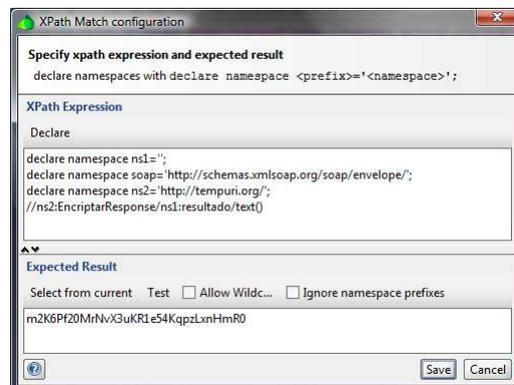
Podemos añadir otra aserción de tipo `Response SLA` para comprobar que la invocación al web service cumple con un acuerdo de nivel de servicio (*Service Level Agreement*) que determine el tiempo máximo de respuesta, que introducimos en la siguiente ventana, en milisegundos:



Acuerdo de Nivel de Servicio

Finalmente añadimos una aserción `XPath Match`, con la que vamos a comprobar, mediante xpath, la existencia del atributo resultado, y el valor esperado en la respuesta: `m2K6PF20MrNvX3uKR1e54KqzLxnHmR0`

En la pantalla de configuración debemos en primer lugar declarar el espacio de nombres. Ello puede hacerse de manera automática pulsando sobre `Declare`. A continuación introducir la expresión xpath para acceder al nodo deseado y extraer su valor, y en la ventana inferior, el valor esperado que queremos comprobar. En nuestro caso queda de la forma:



Aserción basada en XPath

XPath Expression:

```

declare namespace ns1='';
declare namespace soap='http://schemas.xmlsoap.org/soap/envelope/';
declare namespace ns2='http://tempuri.org/';
//ns2:EncriptarResponse/ns1:resultado/text()

```

Expected Result:

```
m2K6PF20MrNvX3uKR1e54KqzLxnHmR0
```

Introducción a los diagramas EPC
(Event-Driven Process Chain)

2009-08-10
Blender. Animaciones avanzadas y
renderización

2009-08-10
Gestión de Calidad, tablón y seguimiento
en TNT Concept Versión 0.16.1

2009-08-10
Cómo hacer una página web

2009-08-06
Tips And Tricks JUnit Spring

2009-08-03
Instalación de VirtualBox PUEL

Últimas ofertas de empleo

2009-07-31
T. Información - Operador (día / noche) -
BARCELONA.

2009-06-25
Atención a cliente - Call Center -
BARCELONA.

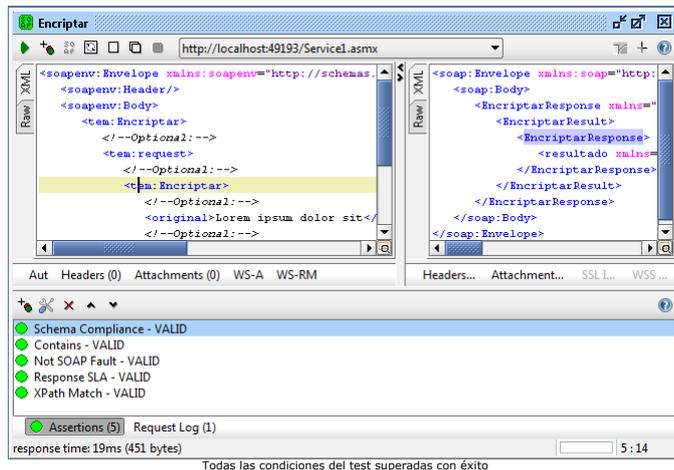
2009-06-19
Otras - Ingeniería (minas, puentes y
puertos) - VALENCIA.

2009-06-17
Comercial - Ventas - ALICANTE.

2009-06-03
Comercial - Ventas - VIZCAYA.

Anuncios Google

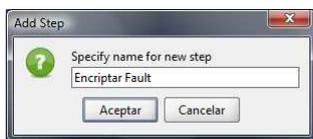
Si ejecutamos la invocación al web service pulsando en la flecha verde en el editor, obtendremos la respuesta y los test superados:



De hecho la condición del SLA se ha cumplido con mucho margen, puesto que el límite de tiempo permitido de respuesta establecimos 200 ms, y realmente ha tardado 19.

Añadir nuevos casos de prueba

Para ampliar la cobertura de nuestros test funcionales añadiremos un nuevo TestStep bajo Encryptar TestCase, de tipo Test Request, y lo denominamos Encryptar Fault, pues vamos a comprobar condiciones de fallo en la invocación al servicio:



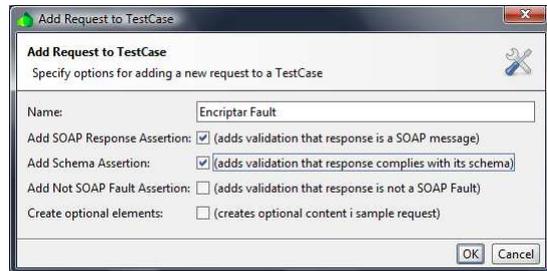
Nuevo caso de prueba

La invocación que nos interesa probar es Service1Soap -> Encryptar:



Interfaz y operación a probar

En este caso aprovechamos el diálogo del asistente para introducir dos aserciones: comprobación de que la respuesta es un mensaje SOAP y que éste cumple con el Schema:



Introducir aserciones a través del asistente

El editor del caso de prueba se abre automáticamente con un SOAP request mínimo:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:tem="http://tempuri.org/">
  <soap:Header/>
  <soap:Body>
    <tem:Encryptar/>
  </soap:Body>
</soap:Envelope>
```

Pulsamos en el icono  para recrear un mensaje de petición que cumpla con el Schema, al que daremos a continuación los valores siguientes:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:tem="http://tempuri.org/">
  <soap:Header/>
  <soap:Body>
    <tem:Encryptar>
      <!--Optional:-->
      <tem:request>
        <!--Optional:-->
        <tem:Encryptar>
          <!--Optional:-->
          <original></original>
          <!--Optional:-->
          <clave></clave>
          <usarHashing>true</usarHashing>
        </tem:Encryptar>
      </tem:request>
    </tem:Encryptar>
  </soap:Body>
</soap:Envelope>
```

La respuesta a la invocación será un mensaje de fallo del tipo:

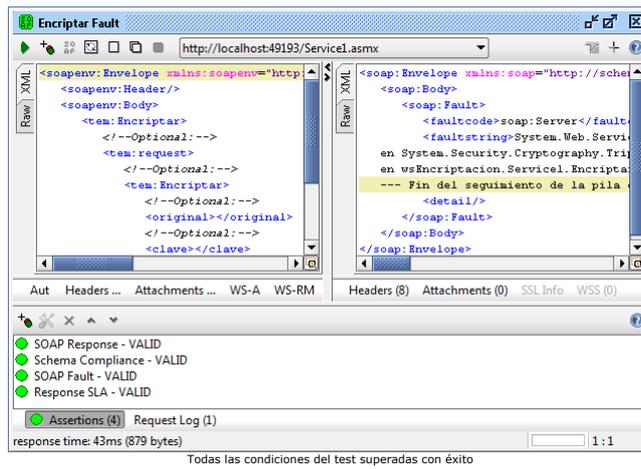
```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>System.Web.Services.Protocols.SoapException:
El servidor no puede procesar la solicitud. ---> System.Security.Cryptography.CryptographicException:
La clave especificada no tiene el tamaño válido para este algoritmo.
en System.Security.Cryptography.TripleDES.set_Key(Byte[] value)
en wsEncryption.Service1.Encryptar(InputEncryptar request) en C:\Users\igpuebla\Documents
\Visual Studio 2008\Projects\wsEncryption\wsEncryption\Service1.asmx.cs:línea 45
--- Fin del seguimiento de la pila de la excepción interna ---</faultstring>
      <detail/>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

```

</soap:Fault>
</soap:Body>
</soap:Envelope>

```

Por tanto vamos a añadir una aserción del tipo Soap Fault. Asimismo la respuesta debe recibirse en un plazo menos a 200 ms, por lo que creamos otra comprobación de tipoResponse SLA. Lanzando la invocación, superaremos las aserciones del test:



Encriptar Fault

http://localhost:49193/Service1.asmx

Raw XML

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <tea:Encriptar>
      <!--Optional:-->
      <!--Optional:-->
      <tea:request>
      </tea:Encriptar>
    </!--Optional:-->
    <original></original>
    <!--Optional:-->
    <clave></clave>
  </soapenv:Body>
</soapenv:Envelope>

```

Raw XML

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>System.Web.Services.Protocols.SoapFaultException: System.Security.Cryptography.CryptographicException: vsEncriptacion.Service1.Encriptar
      --- Fin del seguimiento de la pila de la excepción ---
      <detail/>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

Aut Headers ... Attachments ... WS-A WS-RM Headers (8) Attachments (0) SSL Info WSS (0)

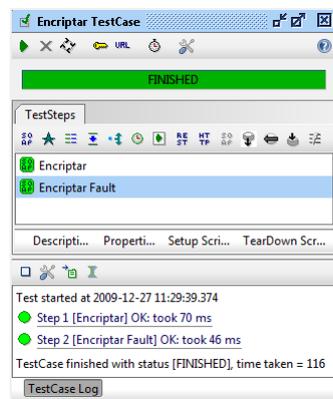
Assertions (4) Request Log (1)

response time: 43ms (879 bytes)

Todas las condiciones del test superadas con éxito

Ejecución conjunta de los casos de prueba

Los casos de prueba anteriores son ejemplos para este tutorial, y en un entorno real deberán ser complementados. En caso de alcanzar un número elevado, o para probar de manera conjunta todos los TestSteps, abrimos el nodo Encriptar TestCase y accedemos a su editor. Ejecutamos el caso de prueba pulsando sobre la flecha verde y vemos que los test se ejecutan secuencialmente:



Encriptar TestCase

FINISHED

TestSteps

- Encriptar
- Encriptar Fault

Descripci... Properti... Setup Scri... TearDown Scr...

Test started at 2009-12-27 11:29:39.374

- Step 1 [Encriptar] OK: took 70 ms
- Step 2 [Encriptar Fault] OK: took 46 ms

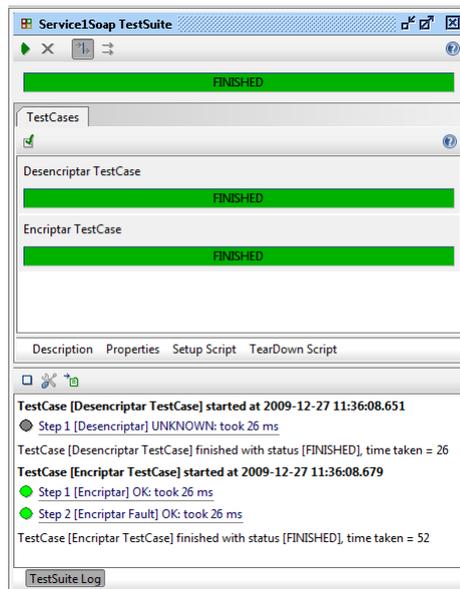
TestCase finished with status: [FINISHED], time taken = 116

TestCase Log

Ejecución de un TestCase completo

Ejecución de una suite completa

Asimismo podemos ejecutar todas las operaciones a nivel de interfaz del servicio web, abriendo el editor de pruebas a nivel de nodoService1Soap TestSuite. Las operaciones sin casos de prueba definidos no interrumpirán la secuencia; podemos observarlo en el log tras una ejecución:



Service1Soap TestSuite

FINISHED

TestCases

- Desencriptar TestCase
- Encriptar TestCase
- Encriptar Fault

Description Properties Setup Script TearDown Script

TestSuite Log

TestCase [Desencriptar TestCase] started at 2009-12-27 11:36:08.651

- Step 1 [Desencriptar] UNKNOWN: took 26 ms

TestCase [Desencriptar TestCase] finished with status: [FINISHED], time taken = 26

TestCase [Encriptar TestCase] started at 2009-12-27 11:36:08.679

- Step 1 [Encriptar] OK: took 26 ms
- Step 2 [Encriptar Fault] OK: took 26 ms

TestCase [Encriptar TestCase] finished with status: [FINISHED], time taken = 52

Ejecución de un TestSuite completo

Características avanzadas de la versión soapUI Pro

Antes de finalizar este tutorial, y por si nos estamos planteando el uso de soapUI a nivel corporativo, quiero comentar un par de características documentadas que me parecen muy interesantes. Quiero dejar claro que no lo hago por publicidad, sino por un criterio personal de utilidad de este software.

Generación de informes

Un jefe de proyecto debe estar informado de la evolución del software que desarrolla su equipo. Si no se dispone de un sistema de integración continua donde pueda observar los paneles de métricas, habrá que proporcionárselas de otra manera. Para ello soapUI Pro dispone de la funcionalidad de generación de informes en HTML con los resultados de las pruebas. La imagen que muestro a continuación está obtenida de la propia documentación:

| soapUI Test Results | | | | | | |
|--|-----------|--------|----------|----------|------------|------|
| TestSuite Amazon.AmazonEC2PortType TestSuite | | | | | | |
| Name | TestCases | Errors | Failures | Time(s) | Time Stamp | Host |
| AmazonEC2PortType TestSuite | 17 | 0 | 1 | 2140.000 | | |

| TestCases | | |
|--|---------|------|
| Name | Status | Type |
| AuthorizeSecurityGroupIngress TestCase | Success | |
| CreateKeyPair TestCase | Success | |
| CreateSecurityGroup TestCase | Failure | N/A |

[CreateSecurityGroup] Failed
 [Response for a SOAP Fault]
 Status: FAILED
 Time Taken: 126
 Size: 533
 Timestamp: Wed Jul 25 23:20:24 CEST 2007
 TestStep: CreateSecurityGroup
 Response is a SOAP Fault

soapUI Pro JUnit-Style Report - <http://www.soapui.org/userguide/projects/reporting/junit.html>

Panel de cobertura

La calidad de una batería de pruebas se mide principalmente por la cobertura que consigue sobre el objeto de prueba. soapUI Pro ofrece un panel de cobertura realmente interesante:

| Element | Contract Coverage | Count |
|--|-------------------|-------|
| Sample Project | 69% (19%) | 25/36 |
| SampleServiceSoapBinding | 69% (19%) | 25/36 |
| login | 83% (0%) | 3/3 |
| logout | 80% (0%) | 4/5 |
| search | 89% (22%) | 9/10 |
| buy | 89% (0%) | 0/0 |
| return | 0% (0%) | 0/0 |
| Sample Simple TestSuite | 0% (0%) | 0/0 |
| Sample TestSuite fails if we don't get fault | 0% (0%) | 0/0 |
| Sample expanded TestSuite | 69% (19%) | 25/36 |
| Search and buy TestCase | 69% (17%) | 25/36 |
| URL DataSource | 69% (0%) | 25/36 |
| Groovy DataSource | 69% (0%) | 25/36 |
| More Than Less Than TestCase | 47% (0%) | 17/36 |
| Test Request: login | 83% (0%) | 3/3 |
| Test Request: search | 89% (22%) | 9/10 |
| Message | 89% (22%) | 9/10 |
| Request | 100% (0%) | 3/3 |
| Response | 83% (33%) | 3/6 |
| Test Request: logout | 80% (0%) | 4/5 |

soapUI Pro Web Service Coverage - <http://www.soapui.org/userguide/coverage.html>

Conclusión

Este tutorial trata de aportar su granito de arena para concienciar de la importancia que tienen las pruebas unitarias dentro del ciclo de desarrollo de software, sea éste mediante metodologías tradicionales, ágiles u otras.

Si además el entorno tecnológico fomenta las buenas prácticas de implementación (alta encapsulación, bajo acoplamiento, interfaces bien definidas, patrones), como es SOA con respecto a sus servicios web básicos, el uso de pruebas unitarias se convierten en una piedra angular en el desarrollo, versionado y mantenimiento no regresivo de los mismos.

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

Muy malo Malo Regular Bueno Muy bueno



Votar

(Sólo para usuarios registrados)

>> [Regístrate y accede a esta y otras ventajas](#) <<

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](#) <<

Autor

Mensaje de usuario registrado

- Puedes inscribirte en nuestro servicio de notificaciones 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.  **Technorati FAVORITES**



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

Creatividad Internet

Tutoriales recomendados

| Nombre | Resumen | Fecha | Visitas | Valoración | Votos | Pdf |
|--|---|------------|---------|------------|-------|---|
| Tutorial de BPEL con OpenESB (1) | En este tutorial vamos a aprender a crear procesos BPEL practicando con un ejemplo: un proceso de negocio de venta online de libros. | 2009-12-29 | 12 | - | - |  |
| SoapUI: jugando con web services | SoapUI es una aplicación muy versátil que nos permite probar, simular y generar código de servicios web de forma ágil. En este tutorial introducimos esta herramienta, mostrando su instalación y un primer ejemplo de uso con un web service público | 2009-12-28 | 106 | - | - |  |
| Pruebas funcionales de servicios web con soapUI | Las pruebas unitarias en cualquier paradigma de programación son, más que una buena práctica, una garantía para obtener un software robusto y (más) fácilmente mantenible. Como responsables de diseño o desarrollo de web services hemos de aplicar estas buen | 2009-12-28 | 90 | - | - |  |
| Contract-First web services con Visual Studio 2008 | Utilizaremos Visual Studio 2008 con el addin WSCF.blue para generar contract-first web services (servicios web dirigidos por contrato) | 2009-12-11 | 394 | - | - |  |
| Integrando Sonar con Hudson | En este tutorial vamos a ver como a partir de un build satisfactorio de Hudson se puede analizar automáticamente el código Java mostrando el resultado en la herramienta Sonar. | 2009-12-09 | 346 | - | - |  |
| Analizando la calidad del código Java con Sonar | En este tutorial vamos a dar a conocer la herramienta Sonar para el control de la calidad del código de nuestros proyectos | 2009-12-07 | 704 | - | - |  |
| Instalar OpenESB 2.1 e Introducción | En este tutorial veremos como descargar e instalar OpenESB y explicaremos sus funcionalidades | 2009-12-03 | 733 | - | - |  |
| Tutorial de Google Forms | En este breve tutorial vamos a ver cómo podemos usar la herramienta Google Forms, para crear rápida y cómodamente un formulario de recogida de datos online (y de paso opinar sobre el Real Madrid). | 2009-11-25 | 990 | - | - |  |
| Haciendo funcionar Google Chromium Operating System sobre Windows XP | Cómo instalar el nuevo SO de Google corriendo sobre XP | 2009-11-23 | 1135 | Muy bueno | 2 |  |
| Alfresco - Añadiendo contenido desde nuestras aplicaciones Java | Tutorial sobre Alfresco, alternativa de código abierto para la gestión de contenido empresarial (ECM), que proporciona gestión documental, colaboración, gestión de registros, gestión de información, gestión del contenido web e imágenes. | 2009-11-23 | 937 | - | - |  |

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.