What is SOAP?

SOAP is a protocol for accessing web services. it is XML based -SOAP provides a way to communicate between applications running on different operating systems, with different technologies and programming languages. A SOAP message is an ordinary XML document containing the following elements: An Envelope element that identifies the XML document as a SOAP message A Header element that contains header information A Body element that contains call and response information A Fault element containing errors and status information

How do you automate web services using SOAP UI?

Create project using WSDL file

-create test suite

-create test case

-add steps to case

-add Groovy script In groovy script editor.

Add comments 1st Then call the script if we have any external file

How can I parameterize the test data in soap UI?

=> We can use excel (Or) Use test case level properties

Which is the open source tool for web services testing?

-SOAP UI (JMETER)

What are the REST COMMANDS?

Create => HTTP PUT

Retrieve => HTTP GET

Update => HTTP POST

Delete => HTTP DELETE

How do you validate your response?

-added assertions to validate the data

For example date format should be mm/dd/yyyy; We can add assertions and regular expressions

Write groovy script to connect to database and retrieve data;

compare the data with the response xml

Difference between REST and SOAP

SOAP is a protocol for sending/receiving data over HTTP as XML.

REST (Representational State Transfer). It's a way of designing a web service.

How to use response coming from one service as a request for other service

By creating a \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What is end point in Web Services?

It is the IP address of the server where the web services are running

How will you do regression testing using soap UI?

By Creating Test Suite in the SOAP UI we can do regression Testing. We need to keep different assertions for the request coverage, that to confirm that all the expected parameters are coming in the response.

What is XML?

Extensible Markup Language is a uniform data representation and exchange mechanism.

What is SOAP?

Simple Object Access Protocol, SOAP is a standard way of using XML vocabulary to enable programs on separate computers to interact across any network and describing messages between applications.

What is UDDI?

Universal Description, Discovery, and Integration specification, UDDI is a mechanism to register and located WS based application.

What is WSDL?

Web Services Description Language, this is a standard Meta language to describe the services offered. Specifically,

WSDL states what a request message much contain and what the response will look like in a clear notation. WSDL also defines where the service is available and what communications protocol is used to talk to that service.

What types of operations are available in WSDL?

There are four operations available:

1. One-way, where the operation can receive a message but will not return a response.

2. Request-response, where the operation can receive a request and will return a response.

3. Solicit-response, where the operation can send a request and will wait for a response.

4. Notification, where the operation can send a message but will not wait for a response.

Define a REST web service?

REST is Representational State Transfer and it is a network of web pages where the client progresses through an application by selecting links. REST is an architectural style that uses existing standards such as HTTP.

What are web services components?

XML, SOAP, UDDI, WSDL

Can I use GET request instead of PUT to create resources?

No, you are supposed to use PUT or POST. GET operations should only have view rights.

What all tools have you used to write Restful web service?

SOAP UI

What are the different styles of Web Services used for application integration?

SOAP WS and Restful Web Service

How would you decide what style of Web Service to use?

SOAP WS or REST? In general, a REST based Web service is preferred due to its simplicity, performance, scalability, and support for multiple data formats. SOAP is favored where service requires comprehensive support for security and transactional reliability. The answer really depends on the functional and non-functional requirements. Asking the questions listed below will help you choose.

Does the service expose data or business logic? (REST is a better choice for exposing data, SOAP WS might be a better choice for logic).

Do the consumers and the service providers require a formal contract?

(SOAP has a formal contract via WSDL)

Do we need to support multiple data formats?

Do we need to make AJAX calls? (REST can use the XMLHttpRequest)

Is the call synchronous or asynchronous?

Is the call stateful or stateless? (REST is suited for stateless CRUD operations)

What level of security is required? (SOAP WS has better support for security)

What level of transaction support is required? (SOAP WS has better support for transaction management)

Do we have limited band width? (SOAP is more verbose)

What’s best for the developers who will build clients for the service? (REST is easier to implement, test, and maintain)

What tools do you use to test your Web Services?

Soap UI tool for SOAP WS

What is the difference between SOA and a Web service?

SOA is a software design principle and an architectural pattern for implementing loosely coupled, reusable and coarse grained services. You can implement SOA using any protocols such as HTTP, HTTPS, JMS, SMTP, RMI, IIOP (i.e. EJB uses IIOP), and RPC etc. Messages can be in XML or Data Transfer Objects (DTOs). Web service is an implementation technology and one of the ways to implement SOA. You can build SOA based applications without using Web services For example by using other traditional technologies like Java RMI, EJB, and JMS based messaging, etc. But what Web services offer is the standards based and platform-independent service via HTTP, XML, SOAP, WSDL and UDDI, thus allowing interoperability between heterogeneous technologies such as J2EE and .NET.

What types of testing ca you perform using SOAP UI?

Soap UI allows you to easily and rapidly create and execute automated functional, regression, compliance, and load tests.

What is Data-Driven testing?

How can you do this using SOAP UI?

Data-driven testing is when you store test data (input, expected output, etc) in some external storage (database, spreadsheet, xml-files, etc) and then use that data iteratively in your tests when running them. For example to test your phone-lookup service, you might have a list of names and expected phone-numbers in a database which you would use to "drive" your test, checking that each name gets the right phone-number back. It's really quite simple.

Which version of SOAP UI Pro are you using?

Ready APi 1.1.0

How to add an Assertion to the test?

1. Open the request editor.

2. in the request editor, click the Add an Assertion to Test Request button.

3. Select Response SLA from the drop down in the Select Assertion dialog box.

4. In the Configure Response SLA Assertion dialog box, write 500 and click OK. This will validate that the response of the SLA is fewer than 500.

5. Now that you've added the assertion, you are going to run the request to validate the response. If all assertions are successful, the SOAP icon should turn green in three places .

6. You can also validate the response by adding an X-Path Match assertion.

7. In Soap UI Pro you can open the Select X-Path dialog, which lets you select nodes to assert using point-and-click. As a result Soap UI creates the matching X-Path expression that refers to the selected node.

8. Soap UI automatically populates the expected result to match what's in the response.

How to access the Project name from a Groovy Script Test Step?

testRunner.testCase.testSuite.project.name (Almost all items have a name property)

How do I save the changes I make to my projects, requests, Test Cases, etc?

Soap UI automatically saves everything on exit. If you want to save your projects without exiting (for example if you want to commit your project file to CVS) use the "Save All" option in the main File menu

How to Create a Mock Service?

1. Right-click on one of the SOAP interfaces and selects Generate Mock Service.

2. In the dialog Generate Mock Service you can specify the local port/path for the service you're creating but for the moment just click OK.

3. Enter the name of your Mock Service in the Name dialog and click OK.

4. after creating the Mock Service, you should get a Mock Service with one operation and one request.

What languages are supported?

-Groovy and JavaScript

What exactly WSDL document contains?

It is a document written in XML and used to describe web services. It specifies the location of

the service and the operations (or methods) the service exposes.

Few tools used in your project if you used web services.

SOAP UI: Test the web service Sometimes some of the back end services might not available, so you can create a mock response and run the web service from the soap UI. Then your program should hit the mock response and you can check the functionality.

Restful service calls

HTTP PUT =>Create HTTP GET => Retrieve HTTP POST => Update HTTP DELETE => Delete

GET: GET is the simplest type of HTTP request method; the one that browsers use each time you click a link or type a URL into the address bar. It instructs the server to transmit the data identified by the URL to the client. Data should never be modified on the server side as a result of a GET request.

PUT : A PUT request is used when you wish to create or update the resource identified by the URL.

DELETE: DELETE should perform the contrary of PUT; it should be used when you want to delete the resource identified by the URL of the request.

POST: POST is used when the processing you wish to happen on the server should be repeated, if the POST request is repeated (that is, they are not idempotent; more on that below). In addition, POST requests should cause processing of the request body as a subordinate of the URL you are posting to.

HTTP Error messages

200 OK - This response code indicates that the request was successful. 201 Created This indicates the request was successful and a resource was created. It is used to confirm success of a PUT or POST request.

400 Bad Requests - The request was malformed. This happens especially with POST and PUT requests, when the data does not pass validation, or is in the wrong format.

404 Not Found - This response indicates that the required resource could not be found. This is generally returned to all requests which point to a URL with no corresponding resource.

401 Unauthorized - This error indicates that you need to perform authentication before accessing the resource.

405 Method Not Allowed - The HTTP method used is not supported for this resource.

409 Conflicts - This indicates a conflict. For instance, you are using a PUT request to create the same resource twice.

500 Internal Server Errors - When all else fails; generally, a 500 response is used when

processing fails due to unanticipated circumstances on the server side, which causes the server to error out.